



uOttawa

L'Université canadienne
Canada's university

**FACULTÉ DES ÉTUDES SUPÉRIEURES
ET POSTDOCTORALES**



**FACULTY OF GRADUATE AND
POSTDOCTORAL STUDIES**

Valentin Borsu

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.A.Sc. (Electrical and Computer Engineering)

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

**Pose and Motion Estimation of Parts Exhibiting Few Visual Features for Robotic Marking of
Deformations**

TITRE DE LA THÈSE / TITLE OF THESIS

Pierre Payeur

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

Robert Laganière

Mark Lanthier

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Pose and Motion Estimation of Parts Exhibiting Few Visual Features for Robotic Marking of Deformations

By:

Valentin Borsu

A thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements for the degree of

**Master of Applied Science
In Electrical and Computer Engineering**

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Information Technology and Engineering
University of Ottawa

December 2010
© Valentin Borsu, Ottawa, Canada, 2010



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-79687-0
Our file *Notre référence*
ISBN: 978-0-494-79687-0

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Abstract

This thesis examines the complex problem of robotic interaction with moving objects exhibiting few distinctive visual features in the context of marking surface deformation defects for quality control in the automotive industry. The designed pose and motion estimator, which is the central component of the proposed robotic tracking and marking station, embeds a feature-based tracking approach, which builds upon the selection of a limited, but consistent set of features and their tracking on a frame-by-frame basis.

While the visual acquisition system relies on low resolution cameras, the proposed algorithm provides sub-pixel accuracy on the pose estimation of an automotive panel, and its associated motion. The pose and motion estimator embeds classical computer vision algorithms for feature extraction, matching and tracking. Their limitations, in the case of tracking industrial objects with few contrasting features, are solved from a software perspective, without complicating their mathematical foundations or the hardware architecture of the visual acquisition system. In order to reliably solve the limitations imposed by the general appearance of the objects, coupled with the complex factory environments in which they exhibit their motion, the pose and motion estimator incorporates a supervisory layer, whose goal is to provide time-efficient, accurate and fault-tolerant visual servoing data to the robotic station. The only knowledge provided to the supervisory layer is related to a limited number of macro-features, which are pre-selected over the structure of the automotive panels, when configuring the robotic tracking and marking station.

The knowledge provided to the system by the macro-features is successfully integrated into the inter-calibration procedure between the defects detection stage, whose description remains beyond the scope of this thesis, and the autonomous robotic tracking and marking station. As a result, only a limited number of macro-features are sufficient to inter-calibrate two sensing devices, located in two different stations along an assembly line. Additionally, this inter-calibration procedure is performed on-line and does not require a target object.

Also, with the integration of the supervisory layer, the experimental validation demonstrates the robustness of the proposed pose and motion estimator to a series of realistic situations, such as occlusions from the robot, slight changes in the illumination or the reflectance properties of the panels' surfaces, as well as the sporadic appearance of factory associates in the view of the acquisition system.

Different defects marking procedures are tested with an actual robot arm, including a stamping operation on a static object. An experimental validation of the robotic marking operation on a moving panel, using an LED-pointer to mimick a spray-gun end-effector, is also performed. The accuracy achieved in the two marking validation phases demonstrates the suitability of the proposed robotic solution to become a viable alternative to perform fully automated region marking of deformations over large surfaces and for substantial volumes of production.

Acknowledgements

I would like to thank my supervisor, Dr. Pierre Payeur for his guidance, support and understanding during my research. This thesis would have not been possible without him.

I would also like to acknowledge the financial support from Precarn Inc., and the collaboration of Neptec Design Group Ltd. and Honda Canada to this research.

Also, I am grateful to many colleagues at SMART Research Group, such as Phillip Curtis, Ana-Maria Cretu, Alain Boyer, Fouad Khalil, Sylvain Bériault and Dejan Duvnjak for their occasional assistance and productive discussions.

Table of Contents

Abstract	ii
Acknowledgements	iv
Table of Contents.....	v
List of Figures	viii
List of Tables.....	xi
List of Abbreviations	xii
Chapter 1. Introduction.....	1
1.1. Motivation.....	1
1.2. Objectives	2
1.3. Thesis Organization	4
Chapter 2. Literature Review.....	5
2.1. Introduction	5
2.2. Pose and Motion Estimation.....	5
2.2.1. Feature Extraction	5
2.2.2. Feature Tracking	10
2.2.3. Feature Matching.....	16
2.2.4. Structure and Motion Estimation	21
2.2.4.1. Stereoscopic Vision	22
2.2.4.2. Structure from Motion	23
2.2.4.3. Motion Estimation	24
2.3. Robotic Interaction with Moving Objects.....	27
2.3.1. Inter-calibration of Vision Sensors and Robotic Arms	27
2.3.2. Visual Servo Control for Robotic Applications	28
2.4. Chapter Summary	33
Chapter 3. Design of the Robotic Tracking and Marking Solution	34
3.1. Introduction	34
3.2. Rigid Body Models	34
3.3. Initial Vision Configurations with Multiple Cameras.....	36
3.4. Proposed Vision Configurations with Only Two Cameras	39
3.5. Description of the Complete Robotic Work Cell	43

3.6. General Framework for Pose and Motion Estimation.....	46
3.6.1. Pose and Motion Estimation Approach Selection.....	47
3.6.1.1. Challenges and Constraints Analysis	47
3.6.1.2. Background Subtraction Approach.....	49
3.6.1.3. Feature-based Pose and Motion Estimation Approach	50
3.6.2. High-Level Description of the Pose and Motion Estimation Solution.....	52
3.7. Chapter Summary	53
Chapter 4. Feature Extraction, Matching and Tracking Processes	55
4.1. Introduction	55
4.2. Macro-Features Selection and Object Detection	56
4.2.1. Selection and Refinement of Macro-Features	56
4.2.2. Re-initialization of the Macro-Features Set.....	61
4.2.3. Topological Structure Buffers.....	63
4.2.4. Object Detection.....	64
4.3. Feature Extraction Analysis.....	70
4.3.1. SIFT Keypoint Detector	71
4.3.2. A Correlated Stability-Robustness Empirical Measure	75
4.4. Feature Tracking Analysis	78
4.4.1. Preliminary Optical Flow Results	79
4.4.2. Limitations of the Pyramidal LK Tracker	80
4.5. Feature Matching Analysis	84
4.5.1. “Torr-Tool” for Feature Matching.....	85
4.5.2. Feature Matching Sub-system	88
4.7. Chapter Summary	91
Chapter 5. Supervised Pose and Motion Estimation	93
5.1. Introduction	93
5.2. Sparse 3D Structure Estimation	93
5.3. Motion Estimation.....	96
5.4. Proposed Supervisory Level.....	99
5.4.1. Coarse-Level Supervision of the Feature Tracking and Matching Processes.....	100
5.4.2. Fine-Level Supervision of the Feature Tracking and Matching Processes	105
5.4.2.1. Sampson’s First-order Correction.....	106
5.4.2.2. Proposed Fine-level Drift Correction.....	108

5.4.3. Impact of the Supervisory Layer in MFs Detection	119
5.4.3.1. Impact of the Supervisory Layer in the Approximate MFs Selection	119
5.4.3.2. Impact of the Supervisory Layer in the MFs Re-initialization Process	121
5.5. A Metrology System for Validating the Pose and Motion Estimations	123
5.6. Chapter Summary	128
Chapter 6. Robotic Integration and Experimental Validation.....	130
6.1. Introduction	130
6.2. Integrated Autonomous System for Marking of Surface Deformation Defects.....	131
6.2.1. Inter-calibration between the Stereo-vision Sensor and the Robot Base.....	132
6.2.2. Inter-calibration between the Stereoscopic Sensor and the 3D Surface Imager.....	135
6.2.3. Robotic Pointing of Surface Deformation Defects.....	141
6.3. Robotic Stamping of Deformations in a Static Environment.....	143
6.4. Robotic Marking of Deformations on Moving Panels	149
6.4.1. On-line Robotic Stamping of Surface Deformation Defects.....	149
6.4.2. Robotic Interaction with Motion Prediction using a Spray Gun.....	156
6.5. Chapter Summary	164
Chapter 7. Conclusions	166
7.1. Summary.....	166
7.2. Contributions	168
7.3. Future Work	170
References	172

List of Figures

Fig. 3.1. Rigid body models used in the experimentation	35
Fig. 3.2. Linear robotic system used for motion generation	36
Fig. 3.3. Initial vision configurations with three cameras	37
Fig. 3.4. Vision configuration with three short baseline stereo-vision sensors	38
Fig. 3.5. Vision configuration with two orthogonal cameras.....	39
Fig. 3.6. Segments from the frames grabbed by CamL and CamU during the tracking	40
Fig. 3.7. Ceiling mounted stereo sensor.....	41
Fig. 3.8. Final stereo-vision sensor	42
Fig. 3.9. Complete deformations detection and marking framework.....	44
Fig. 3.10. Complete experimental setup for deformation detection and marking	45
Fig. 3.11. Manipulator robot with different tool prototypes.....	46
Fig. 3.12. Scaled frames grabbed by CamL, during the tracking sequence.....	49
Fig. 3.13. High-level block diagram of the pose and motion estimation solution.....	52
Fig. 4.1. Initially selected MFs over the surface of the car door and fender model	57
Fig. 4.2. MFs refinement for the car door model	58
Fig. 4.3. MFs disparity vectors computed with the pyramidal LK tracker	60
Fig. 4.4. Refined MFs in the initialization frame grabbed by CamR	60
Fig. 4.5. Image segment with reference frame attached to the rigid body	61
Fig. 4.6. Motion vectors computed during the re-initialization process	62
Fig. 4.7. Topological structure buffers for the car door model	64
Fig. 4.8. Scaled binary images resulting from the background subtraction methodology....	65
Fig. 4.9. Flow diagram of the object detection procedure.....	66
Fig. 4.10. MF ₁ , MF ₇ and MF ₉ binary patches during the object detection process.....	68
Fig. 4.11. Object detection results in CamL's frames	70
Fig. 4.12. SIFT keypoints in scenes with different complexity levels	72
Fig. 4.13. SIFT matching results on frame segment versus full-size frame	73
Fig. 4.14. Extracted SIFT keypoints in image segments during the tracking sequence	75
Fig. 4.15. Feature extraction results in MFs 5x5 pixels patches	77
Fig. 4.16. Preliminary optical flow results computed with the pyramidal LK tracker	80
Fig. 4.17. Optical flow results in scenes with increased complexity.....	81
Fig. 4.18. 1 st and 18 th frame segments with highlighted MFs affected by drift.....	82

Fig. 4.19. Harris corners detected with Torr's toolkit	86
Fig. 4.20. Feature matches computed with Torr's toolkit	87
Fig. 4.21. Epipolar lines associated to the MFs in three segments extracted by CamR.....	91
Fig. 5.1. Results from the evaluation of the linear versus mid-point triangulation	95
Fig. 5.2. Real/estimated 3D sparse structure of the car door's MFs	97
Fig. 5.3. Validation gate for finding the correctly tracked MFs pair	102
Fig. 5.4. Optical flow results after applying the coarse-level validation gate	103
Fig. 5.5. Epipolar residuals before and after applying Sampson's correction	107
Fig. 5.6. Segments from the 1 st and 20 th frames showing the effect of drift on the MFs....	109
Fig. 5.7. Proposed fine-level supervision gate and normalized MFs' structures	112
Fig. 5.8. Results obtained after the integration of the fine-level supervision gate	115
Fig. 5.9. Color/Binary patches of MF ₁ in the 1 st and 17 th extracted frames	117
Fig. 5.10. Results obtained after the intergration of the two supervisory gates.....	118
Fig. 5.11. Disparity vectors associated to fenders' MFs in CamR's initialization frame.....	120
Fig. 5.12. Disparity vectors after the coarse-level supervision.....	121
Fig. 5.13. Metrology system for validating the pose and motion estimations	123
Fig. 5.14. MFs for the mock-up car door and measurements used in the first validation ..	125
Fig. 5.15. Real/estimated 3D trajectory comparison throughout the motion sequence	127
Fig. 5.16. Real/estimated velocities throughout the motion cycle	128
Fig. 6.1. Textured point set surface map of the scanned car door with attached dings.....	131
Fig. 6.2. Results of the 3D surface deformation defects detection system	132
Fig. 6.3. Images taken by CamR during the inter-calibration procedure	133
Fig. 6.4. Inter-calibration of CamR and robot's base procedure	133
Fig. 6.5. Image segments with identified chessboard corners in both views.....	134
Fig. 6.6. Refined MFs in the initialization frames acquired by the SLS	137
Fig. 6.7. Recovered/estimated 3D MFs' area at re-initialization	139
Fig. 6.8. Back-projected MFs in CamL _{SL} 's frame after the two refinements methods	140
Fig. 6.9. 3D deformation meshes defined with respect to CamR for the three dings	142
Fig. 6.10. 3D mesh of Ding ₂ along with its interpolated plane	143
Fig. 6.11. Top/lateral view of stamping tool and scenarios for performance evaluation	144
Fig. 6.12. State-flow diagram of the robot during the stamping procedure	145
Fig. 6.13. Deformation defects stamping results in the three scenarios	146
Fig. 6.14. Results related to MF ₆ (relative to CamR), during the first scenario	150

Fig. 6.15. Results related to MF_6 (relative to O_B) during the first scenario	152
Fig. 6.16. Results related to pre-Ding ₂ with the two inter-calibration procedures	154
Fig. 6.17. Manipulator robot with LED-pointing tool for mimicking a spraying operation ...	157
Fig. 6.18. Estimated/predicted trajectory of MF_7 relative to CamR	159
Fig. 6.19. Estimated/predicted trajectory of the “pre-contact” point of Ding ₂	159
Fig. 6.20. Flowchart for the on-line defects spraying operation	160
Fig. 6.21. Images acquired during the robotic spraying operation on the car door	162
Fig. 6.22. Images acquired during the robotic spraying operation on the fender	163

List of Tables

Table 4.1. Pose changes of the rigid body model at re-initialization	61
Table 4.2. “Success Percentage” for four feature extractors	78
Table 5.1. Error analysis for the motion estimators	99
Table 5.2. Quality measure I – ground truth data comparison.....	125
Table 6.1. Error analysis of the SLS/SSPME inter-calibration	139
Table 6.2. Accuracy of the defects stamping operation.....	147

List of Abbreviations

CAD	Computer-Aided Design
CSS	Curvature Scale Space
DLT	Direct Linear Transformation
DOF	Degree Of Freedom
GT	Ground Truth
LK	Lucas and Kanade
MAD	Median Absolute Deviation
MAPSAC	Maximum <i>A Posteriori</i> Sampling Consensus
MF	Macro-Feature
MLP	Maximum Likelihood Principle
NIST	National Institute of Standards and Technology
PC	Personal Computer
PVT	Projective Vision Toolkit
RANSAC	Random Sampling Consensus
RCS	Radial Cumulative Similarity
RMSE	Root Mean Squared Error
ROI	Region Of Interest
SCR	Size of the Convergence Region
SIFT	Scale-Invariant Feature Transform
SLS	Structured Light Sensor
SSD	Sum of Squared Differences
SSPME	Stereoscopic Sensor used for Pose and Motion Estimation
SURF	Speeded Up Robust Features
SVD	Singular Value Decomposition
TCP	Tool Center Point
USAN	Univalue Segment Assimilating Nucleus
VNC	Variance Normalized Correlation
VRML	Virtual Reality Modelling Language

Chapter 1. Introduction

1.1. Motivation

Quality control in the manufacturing industry has traditionally been performed manually by factory associates. However, due to the advancements in computers, robotics and sensor technologies, automated quality assessment is quickly entering this area of operation, resulting in more efficient, accurate, safe and cost effective solutions. In the automotive industry, quality control is essential in order to ensure that the automotive body parts meet predefined standards. Identifying deformations, such as undesired dings and dents over the surface of the automotive panels, and marking them, such that they are repaired while still on the assembly line is crucial. Therefore, it is very important to fix these defects prior to the subsequent assembly procedures, which will result in more complex structures that will make the repairing process harder. Under current industrial settings, the procedure for identifying surface defects over panels often requires a manual surface rubbing operation, in order to spot tiny but important defects which need to be fixed. The identified deformations are then manually marked with washable ink over the inspected part.

The proposed quality control application, which was defined and developed in collaboration with our industrial partners, combines an autonomous deformation defects detection system with a robotic station that handles the marking of deformations over the automotive panel, with no human supervision beyond system configuration. Therefore, encouraged by the autonomy embedded in a varied range of automotive assembly processes such as welding, painting, drilling or mounting of fixtures, this research work focuses on the integration of a defects detection system and a robotic stage within an autonomous quality control application. While the automated detection of surface deformations remains beyond the scope of this thesis, this document details the second component of the research work, that is linked to the robotic interaction with moving objects under visual guidance, as well as the integration between the two major sub-systems.

Autonomous marking of the locations where deformations appear over an automotive body panel, with a robotic system and on an assembly line, requires that the pose and motion of the panel on the assembly line is accurately estimated. Under the general constraints of car manufacturing, the panel is either translating or rotating, that is, describing a smooth and continuous motion. In most cases, the automotive body panels are unfinished at the stage of inspection. Therefore, the color and texture properties of the

surface are not strongly contrasting or even easily detectable to help in solving the pose and motion estimation problems. Additionally, the pose estimator needs to run without an exact 3D CAD model of the panel, in order to maximize flexibility and computational efficiency, for real-time operation.

The main difficulties met in the cases of tracking and interacting with industrial bodies, which often suffer from a lack of prominent features, have already been highlighted in the literature. For pose and motion estimation, the literature also provides a number of alternatives to the feature extraction, matching, and tracking problems. However, the reliability of these techniques on unfinished automotive body panels is highly affected by the lack of sharp and unique features visible over their surface. As a result, the pose and motion estimator proposed in this thesis represents an original attempt to adapt the existing computer vision technologies such that the solution ensures reliable performance on objects exhibiting few contrasting features.

On the other hand, the literature remains very limited about the problem of robotic interaction with moving bodies in an industrial setting. Some approaches have applied visual servoing for performing the classical “peg-in-hole” experiment on moving industrial objects. However, complex vision systems are often required to accomplish such a robotic task in a controlled environment, and models of the objects are assumed to be available. The current work proposes an innovative solution, from a software perspective, to address the difficulties encountered when applying conventional computer vision algorithms, within a pose and motion estimation methodology, on industrial objects, that is on objects in the process of being manufactured. It results in a solution that maintains a low complexity of the hardware architecture, without introducing any shortcomings to the on-line robotic marking of deformations procedure.

Finally, the research work also addresses the complex issue of accurately integrating several sensing systems along with a robotic marking work cell within a unified solution. As a result, a methodology for inter-calibration between two different 3D vision sensing devices mounted in different stations along the assembly line, and with a manipulator robot, is also proposed.

1.2. Objectives

The primary objective of this research is to develop an autonomous robotic system for marking of surface deformation defects on moving panels, for quality control in the automotive industry. Since the solution must be cost effective and operate in real-time on an

assembly line, the option to mount proximity sensors on the robot arm to assist with close interaction between the robot and the panel is not considered here. The final robotic marking prototype should operate only with visual servoing provided by passive vision sensors. However, because deformation defects also require to be fixed by factory associates, the robotic marking station does not need to be extremely accurate. The goal is rather to efficiently spot the regions in which the deformations appear. As a consequence, a few (3-4 cm) centimeters accuracy is allowed for the marking operation, as long as all regions that contain deformations get marked by the end of the tracking and marking procedure. The proposed solution does not aim at solving problems in metrology but rather at optimizing manufacturing processes. Also, the actual marking step can be performed in a variety of ways, including stamping or paint spraying, as will be discussed in this work.

In the considered application, the general appearance of automotive panels to be inspected imposes severe operational constraints, along with the necessity to operate in real-time. One of the specific objectives is therefore to properly integrate and adapt classical computer vision algorithms for feature extraction, tracking and matching, in the challenging situation where the target object does not exhibit a significant amount of contrasting features. From the perspective of the pose and motion estimation, a methodology must then be developed for the selection of the most relevant and reliable features from what is available on the objects. The constraints of the application push well-established pose and motion estimators up to their limit. Efforts will then be invested at developing innovative hierarchical processing structures to override these limitations without complicating the hardware architecture, and while relying only on passive vision sensing. The proposed approach should also ensure a balance between accuracy and computational complexity. The generality of the solution must also be preserved for the proposed pose and motion estimator and the robotic marking station to properly operate on different types of automotive panels. The solution must also deal with the reality of a robotic manufacturing environment where assembly lines can adopt multiple configurations, and where associates and robots can appear sporadically in the field of view of the cameras.

Finally, the last specific objective imposes a complete integration of the pose and motion estimator with a 3D scanner that is used to collect surface shape measurements from which deformations are identified, as well as with the robotic marking system. The inter-calibration between the deformation defects detection module and the autonomous robotic tracking and marking stage must be achieved accurately since the two systems will be incorporated into two different stations along the assembly line, for improved process

pipeline scheduling. In order to embed the necessary flexibility imposed by this physical separation, an adapted inter-calibration technique must be developed to support efficient system configuration in a factory environment.

1.3. Thesis Organization

This thesis is divided into seven chapters, the first of which is this introduction. The second chapter proposes a review of some of the most important techniques and research results related to the two major components involved in the problem of robotic interaction with moving objects, that is pose and motion estimation along with the actual robotic integration that relies on visual servoing data.

Chapter 3 describes the design process for the complete quality control application for robotic marking of surface defects on automotive body panels, under passive visual guidance. This chapter details the hardware components of the proposed prototype for quality control applications, together with the high-level software framework.

Chapter 4 proposes an overview of the feature extraction, tracking and matching processes, which are the central components of the proposed pose and motion estimation solution. In addition, original strategies for the pre-selection and refinement of macro-features, as well for object detection procedures, are discussed.

Chapter 5 details and experimentally evaluates methodologies for 3D reconstruction and motion estimation, which play a fundamental role in the accuracy of the robotic navigation. Subsequently, an innovative supervisory layer which continuously monitors and validates the feature extraction, tracking and matching processes, which have the strongest impact on the precision of the pose and motion estimations, is presented to complete the proposed pose and motion estimation solution.

Chapter 6 completes the description of the proposed autonomous robotic system by examining various ways for marking the undesired deformations over static and moving automotive body panels. The integration of the robotic marking system within the autonomous defects detection and marking station is also discussed. The problem of robotic stamping of deformation defects in static environments is introduced and its performance is evaluated. Subsequently, the solution is extended to allow robotic interaction on moving panels.

Finally, Chapter 7 provides the conclusions as well as future work to be performed.

Chapter 2. Literature Review

2.1. Introduction

The general problem of robotic interaction with moving objects represents a complex task and an active research subject. The pose and motion estimation of the moving rigid body, together with the robotic integration are the major components of the considered application. The first module builds upon the extraction of a set of reliable features in the structure of the rigid body, followed by their tracking on a frame-by-frame basis, the 3D reconstruction of the sparse structure of the object, and finally the object's motion estimation throughout the tracking sequence. The second module relies on the inter-calibration between the robot and the pose and motion estimator and is responsible for the actual robotic interaction with the mobile target.

In this chapter, important research work on all the components of these two main sub-systems for the application considered is reviewed.

2.2. Pose and Motion Estimation

The pose and motion estimation sub-system offers information about the location of the object in the 3D scene and its associated motion. The extraction of reliable features, the tracking and matching processes and the formal estimation of structure and motion are examined in this section.

2.2.1. Feature Extraction

By taking into account the characteristics of the tracked objects, discussed in Chapter 1, the robust feature extraction is a vital step for the pose and motion estimation of the dynamic body. Simply selecting a point on a patch that has uniform color and no variations nor texture would likely make the tracking of that point in subsequent frames unsuccessful. Therefore, attention must be given to choosing keypoints, parameterizable in such a way, that they can be compared to other points in subsequent images.

Harris and Stephens [1] have introduced the “combined corner and edge detector” by addressing the problem of consistency in image edge filtering. Their corner extractor is based on the local auto-correlation function in order to deal with image regions that contain texture or isolated features. After characterizing the corners as “discrete, reliable and meaningful” the authors state that explicit feature tracking can successfully be accomplished

by concentrating on keypoints that are not included in a continuum-like texture. The general suggestion is related to the fact that the points which exhibit significant amount of change, involve a strong derivative. Moreover, if these strong derivatives are observed in two orthogonal directions, then chances are for the point to be a corner. As a result, a 2x2 symmetric matrix, related to the auto-correlation function, is introduced:

$$M(x, y) = S * (\nabla I(x, y) \cdot \nabla I^T(x, y)) = S * \begin{pmatrix} I_x^2(x, y) & I_x I_y(x, y) \\ I_x I_y(x, y) & I_y^2(x, y) \end{pmatrix} \quad (2.1)$$

where S is a smoothing operator and $I_x(x, y)$, $I_y(x, y)$ are the image gradients in the x and y directions. By considering the matrix above, the eigenvalues (α, β) will be proportional to the principal curvatures of the local auto-correlation function. Under these settings, a corner is characterized by both eigenvalues being large and the suggested corner strength measure (that does not require explicit eigenvalue decomposition), is given by:

$$\text{CornerStrength}^{\text{Harris}} = \det(M(x, y)) - k \cdot \text{Trace}^2(M(x, y)) \quad (2.2)$$

where the parameter k is usually equal to 0.04. The corner response is large when both of the eigenvalues are large. However, as interestingly pointed out in [2], there is a certain level of difficulty in determining what “large” means, especially when the terms included in the corner response (eq. (2.2)) have units of intensity gradient to the fourth power. That is why, the Harris operator is sensitive to image patches containing contrast variations, resulting in a considerable amount of difficulty when setting the threshold for the corner strength. For this reason, and the concern caused by the proper selection of the parameter k , Noble [3] has introduced a new strength measure for the corners, based on the same matrix as in (eq. (2.1)), given by:

$$\text{CornerStrength}^{\text{Noble}} = \det(M(x, y)) / \text{Trace}(M(x, y)) \quad (2.3)$$

which attempts to solve the above mentioned drawbacks. With this new measure [2, 3], similar eigenvalues that are small in magnitude will produce a smaller response than in (eq. (2.2)), simplifying the problem of robust threshold setting.

Inspired by the work presented in [1, 3], Brady and Wang [4] proposed a new corner detector that relies on the edge tangential curvature which is estimated by means of the linear interpolation and non-maxima suppression used in Canny’s edge detector [5]. Their corner strength measure is given by the ratio between the tangential curvature and the norm of the image gradient.

Similar to Harris and Stephens [1] and Noble [3], Shi and Tomasi [6] are interested in identifying corners or regions characterized by a noticeable high mix of second derivatives.

Taking into account the noise sensitivity of these techniques, which rely on derivatives of image intensities, Shi and Tomasi [6] introduce a validation gate related to the fact that the two eigenvalues of the matrix given by eq. (2.1) cannot differ by several orders of magnitude. As expected, two large eigenvalues can represent corners, salt-and-pepper textures or any other formation that can be tracked in a consistent manner.

The feature extraction process was formulated as a signal parameter estimation problem by Tan *et al.* [7]. By considering the simplistic approach in which arbitrary 2D image intensity patterns are affected by the same displacement, the authors are interested in the selection of image patches which have a large feature response. The proposed feature strength measure is based on the sum of the reciprocal eigenvalues of a Fisher information matrix similar to the one in eq. (2.1). However, their technique tends to be more general than the ones presented in [1, 6] since the considered block features can have different sizes and shapes and their corner strength measure represents a quantitative measure of the quality of the features, and not a heuristically related descriptor. The limitation of the proposed framework is related to the simplicity of the motion model associated with the feature which is highly sensitive to perspective distortion that can affect the images during the tracking.

Smith and Brady [8] have proposed the well-known “SUSAN” corner detector which relies on the calculation of the area of a mask, inside a circular window, that includes pixels which share the same intensity with the central point of the circular patch. The authors [8] are interested in the minimization of this local image region called the univalue segment assimilating nucleus (USAN). In order to detect edges and corners, thresholds are imposed for the ratio between the USAN's area and the entire area of the considered circular window.

A new direction in the field of feature extraction has been initiated by Lowe [9] with the introduction of SIFT (“Scale-Invariant Feature Transform”) features that are invariant to both scaling and rotations. The suggested procedure is started by a search over all scales and locations of the images, guided by a difference-of-Gaussian function, for the discovery of possible keypoints. Subsequently, the location and scale of the features are identified and the most stable keypoints are extracted. Finally, an orientation and a descriptor are assigned for each feature, for consistent matching and robustness to perspective distortions and illumination changes.

In order to solve the variance to scale changes of the Harris detector [1] and to obtain less computational complexity than SIFT extraction [9], Wei and Bao-long [10] have

proposed a hybrid Harris-SIFT feature detector. Firstly, the authors use a scale-adapted M matrix (eq. (2.1)) for localizing Harris corners in the scale-space. Then, based on the SIFT algorithm [9], the extracted features are assigned an orientation and a descriptor in a way in which orientation invariance is achieved. The experimental results on real images demonstrate the value of the proposed algorithm in extracting better distributed features than Harris detector [1], within an execution time that is almost a quarter of the SIFT extraction time [9].

Since the SIFT Detector achieves “near real-time performance” [9], Bay *et al.* [11] have proposed the SURF (“Speeded Up Robust Features”) extractor, which is able to detect features invariant to scale and rotations, within less computational time. Their proposed detector builds upon an approximation of the Hessian matrix using box filters and integral images. Additionally, the authors addressed the problem of “repeatability” for the corner detection, which is related to the ability of the feature extractor to detect the same keypoints, even in the cases in which the images are affected by perspective distortion, scale and zoom [11, 12].

Interested in the general problem of stereo matching, Vincent [12] has introduced three feature-point detectors based on the baseline of the stereo-vision system used for scene reconstruction, and the availability of the stereo-calibration information. A first solution related to the case of a narrow baseline calibrated stereoscopic system is to use the epipolar gradient feature-point detector. Features are selected if the image intensity gradient is collinear with the epipolar lines, and this intensity gradient, in the direction of the epipolar lines, is called epipolar gradient. The experimental results demonstrate the superiority of this feature extractor, when compared to the Harris detector [1], from the perspective of the feature distribution which supports a better structure reconstruction from a sparse set of points. However, the generality of this approach is highly affected by the way in which the stereoscopic sensor is located with respect to the imaged scene. To account for this limitation, the author proposed a configuration consisting of a three-view L-shaped vision system.

Inspired by the process of assigning invariant descriptors to the matched features [9, 11], Vincent [12] proposed a new feature detector useful in the cases in which the uncalibrated stereoscopic sensor encompasses a large baseline. The “wedge-based” corner detector builds upon the segmentation for extracting univalue segments performed by the SUSAN corner detector [8]. However, the extracted features are explicitly compared to a corner model, characterized by an angular position and an angular width, which is also

useful in the process of estimating the local affine transformation that the corner regions have undergone between the two widely separated views. Finally, for the case in which the calibration information of the widely separated stereoscopic sensor is available, Vincent [12] suggested the possibility of detecting junction points which are defined at the intersection of two non-collinear image line segments. The addition of this useful information to the extracted feature points is very important in the process of estimating the perspective distortion present between the two views of the stereo-vision system.

He and Yung [13] introduced a Curvature Scale Space (CSS) corner extractor for multi-scale feature detection. The input to their algorithm is represented by a binary edge-map computed with Canny's edge detector [5] from which the edge contours are extracted. In order to detect both fine and coarse features simultaneously, all of the curvature local maxima are used in the extraction process that relies on adaptive local thresholds and dynamic regions of support (for computing corner angles) with the purpose of removing false corners.

Following the general framework of [7], Kaneko and Hori [14] proposed a feature extractor for consistent tracking by means of a template matching approach. In order to obtain a higher level of generality than in [7], an affine motion model is considered in the template matching problem. While the algorithm outperforms Harris [1] and SUSAN [8] detectors, for the cases in which the real images are affected by Gaussian noise, the principal limitation is related to the high computational complexity for the feature extraction.

An interesting extension to the SIFT detector [9] has been introduced by Park *et al.* [15] where, apart from the invariance to scale and rotations, the color information is also used in order to provide stability to photometric variations including highlights, illumination changes and shadows. By making use of the dichromatic reflection model and the opponent color space and hue, the authors are able to identify photometric quasi-invariant features which are then grouped into scale spaces in order to also achieve geometric invariance [9]. The validation shows that the proposed " π -SIFT" extractor [15] improves the robustness to illumination changes by increasing the proportion of correctly matched points. However, no information is given regarding the additional computational complexity for adding the invariance to photometric variations to the SIFT detector [9].

After highlighting the limitations of the Harris corner detector [1], such as variance to scale changes [12], sensitivity to noise [2, 3] and non-symmetrical distribution of corners [10, 12], Liu *et al.* [17] make the transition to a multi-scale Harris extractor based on wavelets. Therefore, the authors propose a parallel architecture for extracting features by

using corner detection pyramids. The first corner pyramid is related to the approximate coefficients of the image wavelet transform which are associated to the low frequency content of the image, and as a result, corners belonging to the extracted contours are detected. The second corner pyramid is coupled to the detail coefficients of the image wavelet transform which are linked to the specific gradients of the objects. By combining the sets of detected features from the two corner pyramids, the proposed corner extractor is able to solve the identified limitations [2, 3, 10, 12, 17] of the Harris corner detector [1]. The low computational complexity of the proposed algorithm and the quality of the experimental results highlight the superiority of this approach with regards to the feature extractors and trackers that rely on template matching [7, 14].

Guided by the same objectives of identifying “good” features for tracking [6, 7, 14], Živković and van der Heijden [18] introduce a new “goodness” measure related to the “size of the convergence region” (SCR) of image patches, with respect to the tracking process. Under this framework, a smaller estimation error is obtained in the calculation of the displacement of an image patch if the latter is associated with a high convergence region. Unlike the heuristic method of characterizing the corners, introduced by Shi and Tomasi [6], or the quantitative measure proposed in [7], the authors use an ingenious validation technique for computing the region of convergence, which has its origins in the methodologies for non-linear system analysis. While the computational complexity for computing the SCR for a specific feature is similar to the computational time for tracking that feature, the experimental results demonstrate the success of this measure from the tracking perspective and highlight its improvements when compared to the Harris [1] and SUSAN corner detectors [8].

2.2.2. Feature Tracking

The characteristics of the objects considered for tracking in the current research work, introduced in Chapter 1, guide the tracking methodology towards a feature-based approach which relies on the estimation of motion vectors, on a frame-by-frame basis, for a set of visually significant keypoints *a priori* selected. The motion analysis of these feature points forms the basis for the final estimation of the structure and motion experienced by the moving body. At the core of the general technique for assessing motion over a sequence of frames, without any *a priori* knowledge about their content, the computation of a sparse optical flow is well established and often based on Lucas and Kanade's technique [19].

The approach introduced by Lucas and Kanade [19] was related to stereo-vision and the problem of stereo matching for image registration. However, their solution can also be employed over successive frames in time rather than over a spatial distribution, such that the feature correspondence problem becomes a tracking issue. The generalization of this transition was initiated by Tomasi and Kanade [20] in 1991, and will be briefly explained in this section. By considering an image region, W , characterized by “rich” texture information, the proposed solution searches for the optimal motion vectors by minimizing a criterion which is given by the L_2 norm of the difference between the image intensities around the selected region [17-20]:

$$\mathcal{E} = \iint_W ([I_{t+1}(x - d_x, y - d_y) - I_t(x, y)]^2 w) dx dy \quad (2.4)$$

where $I_t(x, y)$ represents the intensity of the grayscale image at pixel location (x, y) and at time t , $D = [d_x \ d_y]^T$ is the searched image displacement at the location (x, y) during a time period and w is a weighting function, that can be chosen according to the image intensity pattern in the window of interest. At the basis of the criterion selection stood three important assumptions [21]:

- *Brightness constancy*, according to which the brightness of a pixel is not changing during tracking on a frame-by-frame basis.
- *Temporal persistence*, which assumes that the frame rate of the sequence is high enough with respect to the velocity of the object.
- *Spatial coherence*, based on which the neighboring pixels that belong to the same rigid body exhibit similar motions.

As performed in [19], the first-order Taylor expansion of the first term in the minimization criterion, $\mathcal{E}$, is used under this framework as well [20, 21]:

$$I_{t+1}(x - d_x, y - d_y) \cong I_{t+1}(x, y) - [I_x \ I_y] \cdot D \quad (2.5)$$

where the vector $[I_x \ I_y]$ represents the image intensity gradient in the x and the y direction respectively. By substituting eq. (2.5) into eq. (2.4) and setting the first derivative of the criterion, $\mathcal{E}$, to zero, the optimal displacement is obtained [17-21]:

$$D_{\text{opt}} = G^{-1} \cdot b \quad (2.6)$$

where $G = \iint_W \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} w dx dy$ is the 2x2 spatial gradient matrix (similar to the matrix

M described by eq. (2.1), ([6]), and $b = \iint_W (I_{t+1} - I_t) [I_x \ I_y]^T dx dy$ represents the two-

dimensional image mismatch vector at location (x,y) . Eq. (2.6) also gives more insight into the technique suggested by Shi and Tomasi [6] for the extraction of reliable features for tracking, since their corner strength response is related to the condition number of the matrix G in eq. (2.6).

The idea of applying a “coarse-to-fine” strategy in the feature matching process for image registration was initially suggested by Lucas and Kanade [19] with the purpose of improving the accuracy of the computed correspondences. As it was also noticed by Liu *et al.* [17] and Bouguet [22], the first-order Taylor approximation in eq. (2.5) can only be used when the displacement vector, D , is small enough. Therefore the tracking process becomes unsuccessful when the speed of the inspected objects is too high. As a result, the “coarse-to-fine” search proposed in [19] is transposed into the pyramidal implementation of the Lucas and Kanade (LK) feature tracker [22].

According to Bouguet [22] the possibility of detecting large feature displacements is highly correlated with the size of the search window, W . However, the selection of the proper size for the search window needs to properly balance the trade-off between the accuracy and the robustness of the feature tracker. On one hand, the feature tracker should detect the feature displacements with sub-pixel accuracy, which can be achieved with a small integration window, in order not to smooth out the details. On the other hand, the use of small windows preempts the identification of large motions or the needed insensitivity to photometric variations, highly affecting the robustness. As a result, there is always a “natural trade-off” [22] between local accuracy and robustness when selecting the dimensions of the integration window.

In order to keep the size of the search window fairly small, while still being able to identify large pixel motions, the pyramidal implementation of the LK tracker [19, 20] starts with setting the zero-th level of the pyramid, corresponding to the original image having the highest resolution. Then, the pyramid of images is built recursively, from the bottom layer to the top, where the image has larger spatial scales. After computing the optical flow at the deepest pyramid level, the result of the computation is propagated to the next level in the form of a guess or estimation for the pixel displacement at that level. Given this initial guess, the refined optical flow is computed, and the new result is transmitted as a new guess to the next level and so on, until the base of the pyramid is reached. At each of the stages, the iterative LK tracker [20] is employed.

The methodology proposed by Bouguet [22] is applied on a wavelet pyramid by Liu *et al.* [17] which also introduced the transition to multi-scale Harris feature detection.

Although the experimental validation [17] showed reliable results in the tracking of a vehicle, for traffic monitoring purposes, the authors only relied on the comparison with the iterative LK tracker implementation [19, 20], without referencing the existing pyramidal implementation [22].

As it can be noticed from eq. (2.4), the minimization criterion selected by Tomasi and Kanade [20] relies on a purely translational model, and does not consider the situation in which the two analyzed views are affected by perspective distortion. Therefore, an extension is suggested by Shi and Tomasi [6] in which the optimization problem involves a full-affine motion field model characterized by a 2×2 deformation matrix and a two-dimensional translation vector. Furthermore, Jeon *et al.* [23] notice that the distortion of the local regions within the two analyzed frames can be characterized by only four parameters (2-D translation, scaling and rotation), leading to a weakly-affine motion model that lowers the computation complexity of the approach presented by Shi and Tomasi [6].

The possible sources of errors that affect the optical flow calculation, based on the above mentioned assumptions, have been divided by Weber and Malik [24] into four categories. The first class of errors, called “stochastic”, are caused by the sensor noise. The second category is represented by errors due to the failure of the brightness constancy assumption, which happens in case of occlusions, changes in illumination or variations in the light reflectance properties of the objects’ surface. Thirdly, the class of “systematic” errors is linked to the cases in which a large displacement is present in-between the processed frames. The pyramidal implementation of the LK tracker [19, 20, 22], together with the multi-scale processing approaches [24, 25], exhibit robustness to this third class of errors. Finally, the cumulative errors in the motion prediction process are also to be taken in consideration [24], as they produce a drift of the features throughout the tracking history. Under these settings, the literature provides a considerable number of solutions for robust feature tracking, some of which will be shortly presented in the next paragraphs. Nevertheless, it should be noted that the procedures of validating and improving optical flow estimations or adding additional knowledge to the tracking process have a direct impact on the trade-off between accuracy and computational costs.

Wei and Bao-long [10] use the pyramidal implementation of the LK tracker [20, 22] to compute the motion vectors for *a priori* extracted Harris-SIFT features. Then, with the purpose of monitoring the tracking, two additional measures are added to the system. The first one is related to the computation of the dominant motion vectors in the scene, and is used to target the regions in which the objects of interest are situated. The second one is

constantly monitoring the tracking error, which quantizes the amount by which the tracked patches have changed over time. By using a full-affine transformation model, the target objects are identified by feature matching with an available image database.

Guided by the goal of overcoming the thresholding problem [10, 24] needed for improved tracking, Fusiello *et al.* [26] proposed an automatic method for the selection of a robust threshold for rejecting erroneous motion vectors. By assuming that the intensity distributions in the features' regions belonging to the first and the current frames differ by Gaussian noise, the authors are able to obtain an expected modified Gaussian distribution for the tracking residual (computed as a SSD ("Sum of Squared Differences") between the image patches). Therefore, the recommended model-free "X84" rejection rule discards features whose associated tracking residuals are farther than 5.2 Median Absolute Deviations (MAD) from the median element of the distribution.

Uenohara and Kanade [27] introduced a validation gate for the feature tracking, relying on geometric and affine moment invariants [28] from five coplanar points. Additionally, they are addressing the problem of sensitivity of the geometric invariants, which highly depend on the configuration of feature points and their motion, and make the thresholding process difficult. As a result, the authors assume that the tracking error of each of the features has a Gaussian distribution and propose thresholds based on the standard deviation of each computed invariant. A first identified limitation of the evoked technique is linked to the fact that it is difficult to extract the set of ground-truth invariants even at the beginning of the processing sequence, since the locations of the detected features are also noise-prone. Then, occlusions are a very frequent problem in object tracking which might result in the loss of a part of the five selected points, preempting the use of the suggested method.

A more interesting framework in the field of feature tracking using geometric invariants was recommended by Tsui *et al.* [29]. Their method builds upon the tracking of eight general points in space, which are used to estimate the fundamental matrix [45, 49] between subsequent frames. Based on the requirement of tracking stability for the selected set of points, the authors make use of the equations of a "simulated" trifocal-tensor [12] for tracking all the other extracted points. Similar to [27], the accuracy of the introduced method relies on the tracking correctness of the eight chosen points that will be used to estimate the fundamental matrix.

Yoon *et al.* [30] identified the challenges met in the cases of tracking industrial objects, which often suffer from a lack of prominent features for tracking or matching.

Moreover, since their general goal is the automation of industrial processes, the authors point out the necessity of accurate real-time object tracking. Based on this requirement and given the structure of the tracked body, the authors select three circular features for tracking on a frame-by-frame basis. Based on the *a priori* knowledge of the appearance of the objects being tracked, and their motion patterns, a 2D reference frame is attached to the initially grabbed frame that contains a full view of the object. This reference frame gives reliable information about the location of the features in the image space and helps in the process of assigning regions of interest. Then, the authors apply blob tracking to identify the centroid of the circular features during the motion sequence. The locations of the regions of interest (ROIs) in a new extracted frame, during tracking, are updated based on the positions of the target features in the previous frame, by considering that the displacement of the target features is not exceeding the size of the ROIs.

Shokurov *et al.* [31] proposed a feature tracking technique that does not rely on image pyramids and multi-scale optical flow computation [17, 22], but rather on a cross-correlation approach comparable to the feature matching technique mentioned by Whitehead and Roth [48]. The recognized drawback of the algorithm is related to the fact that only pixel-precise accuracy can be obtained. The feature filtering approach resembles the tracking method indicated by Tsui *et al.* [29], with the estimation of trifocal tensor geometry between pairs of three frames, but within a RANSAC (Random Sampling Consensus)-like procedure [32]. A similar RANSAC-like approach for discarding the erroneous motion vectors while refining the 3D motion that a mobile robot has undergone between two extracted frames, was proposed by Milella *et al.* [33]. The authors are using the dual number quaternion method introduced by Zhang [34] to estimate the robot's 3D ego-motion.

Olson *et al.* [35] have identified the problems of feature tracking based on correlation criteria, for robotic ego-motion estimation. On the one hand, they notice that the selection of a large search area results in increasing the density of the erroneous matches, since the appearance of a feature region in between the extracted frames may change more than the difference in appearance of the feature region with respect to other image locations, especially in an outdoor environment. On the other hand, the search region cannot be too small either, since it may not contain the searched feature point. To solve this trade-off, also identified by Bouguet [22], the authors are predicting the locations of the searching regions based on the estimations from robotic odometry.

Jin *et al.* [36] proposed an innovative solution to the tracking problem in order to improve its robustness to changes in illumination and image regions undergoing affine deformations. The authors identify the limitations of the Lucas-Kanade feature tracker [19, 20] to long sequences caused by the accumulating tracking error, and suggest the use of the full-affine transformation model suggested by Shi and Tomasi [6]. In order to overcome the sensitivity of the LK tracker [19, 20] to illumination changes, a hybrid model based on photometry and geometry is used to characterize the change in the appearance of the image patches during tracking. Since the measure proposed by Tomasi and Kanade [20] for evaluating the tracking error is not invariant to intensity variations among the patches of interest, Jin *et al.* [36] recommended the use of the normalized cross-correlation as a proper discrepancy function. The proposed algorithm is successfully tested on real images, that undergo changes in contrast and brightness, and its superiority is demonstrated to the LK tracker [19, 20] using the full-affine motion model [6].

An interesting method based on motion prediction, to aid in the feature tracking process, was introduced by Yao and Chellappa [37], which addressed the problem of ego-motion estimation for mobile robots. Thus, the authors assign to each of the *a priori* extracted features a “local” 2-D translational and rotational model that makes use of the temporal information in the sequence. Moreover, their system relies on the estimation of inter-frame motion between patches in which the detected features are to be expected. By making use of probabilistic data association filtering, image warping, correlation matching and the Extended Kalman Filter, the authors propose and validate an efficient framework for pose and ego-motion estimation.

2.2.3. Feature Matching

One of the fundamental problems in computer vision is to find corresponding points between different views. A few examples of applications that strongly rely on feature correspondences include the reconstruction of the 3D structure of an inspected scene for robotic navigation, the calibration of multiple-view systems or the content indexing of an image database. Vincent [12] has analyzed the problem of feature matching between different configurations of calibrated and un-calibrated multi-camera systems. In order to be able to find feature correspondences between multiple views, two conditions must be met. The first one is related to the fact that the feature points must be visible in all inspected views. The second condition is linked to the difference in appearance between the

corresponding image patches, which should be completely characterized by a restricted transformation space.

This section introduces the state-of-the art in feature matching, by examining the three fundamental choices [12] that this process requires. The selection of the feature type, the chosen similarity measure for comparing features, and the general scheme in which the similarity measure is applied, will be analyzed. Moreover, the literature proposes different strategies for the feature matching process, involving attribute-based methods [12, 25, 38-41], guided matching [12, 31, 42-46] or a combination of these techniques [12, 31, 45, 47, 48].

Vincent [12] proposed an empirical study for the feature matching strategies using attribute-based methods. For the selection of the feature points, the author used the Harris corner detector [1], and the variance normalized correlation (VNC) as a similarity measure for comparing the feature points. The VNC for a candidate match (x,y) in the correspondent images I_x, I_y is given by [12]:

$$\text{VNC}(x,y) = \frac{1}{N\sqrt{\sigma_{I_x}^2(x) \cdot \sigma_{I_y}^2(y)}} \sum_{k \in W} [I_x(x+k) - \overline{I_x(x)}] \cdot [I_y(y+k) - \overline{I_y(y)}] \quad (2.7)$$

where $\overline{I_x(x)}, \sigma_{I_x}^2(x)$ are the sample mean and variance of pixel intensities over an image patch W around x , having a cardinal given by N . However, the amount of noise in the processed images, and the appearance of the selected image regions have a considerable impact on the success of the VNC in the feature correspondence problem. Apart from this selection of similarity measure, that removes a good proportion of outlier matches, there is a group of other attributes that can be used in the feature matching procedure, with the goal of improving the density of good correspondences.

The unicity property [12, 42] of the feature matching is related to the fact that for a particular feature-point only the corresponding match with the highest VNC score should be kept. Since Fua [38] noticed that imposing unicity results in introducing asymmetries in the similarity measure process, he proposed the additional use of the symmetry constraint [12, 42]. Therefore, (x,y) is selected as a match if x is the strongest match for y , whereas y is the strongest match for x . The experimental validations [12, 42] confirm the improvement of the proportion of good matches when applying the symmetry attribute, together with VNC and unicity constraints.

The feature point properties can also be used in improving the feature matching process [12, 38, 42]. A first measure is related to the two eigenvalues of the matrix given by

eq. (2.1) which is computed over an image region centered on the feature-point, and helps define the “cornerness” at that particular point. The corner orientation constraint which can be computed by using the same matrix as in eq. (2.1) results in imposing a threshold for the bisector of the angle defined by the two principal gradient directions in the corresponding views [12]. However, Vincent [12] has shown that these constraints do not improve the proportion of good matches, and he concluded that these attributes are intrinsically tested by the VNC similarity measure.

Deriche *et al.* [39] indicated a set of three constraints based on the attributes of the features, to be used with the correlation test. Unlike [12], Deriche suggested the use of the cosine of the angle between the dominant gradient directions of the matched corners. Moreover, by analyzing both gradient directions for a feature point, a constraint of curvature is also proposed and a threshold is imposed on the magnitude of the curvature difference between the matches. Finally, the third constraint is related to the neighborhood of the feature points and the need for *a priori* knowledge about the disparity of the matching points.

An interesting technique suggested in [12] is the corner shape similarity test that builds upon the ideas embedded in the SUSAN corner detector [8], and the block-truncation coding proposed by Delp and Mitchell [16]. Therefore, the correlation window used in computing the VNC is divided into two regions, the background and the foreground, and a threshold is imposed on the Hamming distance between these segmented binary patches. Based on the same principles, Darrell [40] is interested in assigning a signature to the matched features based on a radial cumulative similarity (RCS) transform, for a better characterization of the local image homogeneity and the surrounded local similarity structure. When compared to [16], the RCS transform is able to perform a better segmentation by highly attenuating the influence of exterior pixels and providing invariance to background contrast.

Vincent [12], Li *et al.* [41] and Laganière *et al.* [42] have proposed an automatic method for enforcing displacement consistency based on the disparity gradient, without any initial knowledge on disparities, as in the case of Deriche *et al.* [39]. The considered non-iterative approaches [12, 41, 42] take advantage of a local constraint which imposes that a certain displacement for a feature needs to be similar to the ones of its closest neighbors. However, the proposed techniques are highly dependent on the density of the extracted features and cannot be successfully applied in scenes that contain objects with few details.

In the following paragraphs the strategy of guided matching with estimation of the epipolar geometry is discussed. In these cases, it is supposed that an *a priori* set of feature

matches is available and the goal is to use it in estimating the epipolar geometry of the vision system. Since the projections of the same 3D point in different views are characterized by the epipolar geometry, its robust estimation is very important in the matching process. Therefore the general goal is to filter the set of correspondences such that all the matches are consistent with the geometric constraint, and then replenish this set with the use of the estimated fundamental matrix. The base of the 8-point algorithm for computing the fundamental matrix was set by Longuet-Higgins [43], who was interested in finding the relative orientation between two viewpoints by using a set of already computed correspondences. However, the proposed methodology was extrapolated to the case in which the epipolar geometry is estimated in order to correct erroneous correspondences and guide the matching procedure.

Since the 8-point algorithm [43] is very sensitive to noise present in the feature matches, Hartley [44] proposed an efficient normalization procedure, as a pre-processing step in the estimation of the fundamental matrix, which results in comparable performances with the iterative procedures using minimizations [46] and RANSAC paradigms [12, 32] for improved robustness. This normalization, which made the transition to the “normalized 8-point algorithm” [44], consists of a translation such that the centroid of the set of 8-points is at the origin, and an isotropic scaling such that the average distance from the new origin to any of the translated points is equal to $\sqrt{2}$. In this way, the condition number of the measurement matrix in the linear 8-point method [43, 44] is increased and the matrix inversion process becomes more stable. From a different perspective, Kanatani and Sugaya [46] presented a “very compact” and robust algorithm for estimating the fundamental matrix by using the maximum likelihood principle (MLP) and considering that the noise in the feature matches follows a Gaussian distribution. While Hartley [44] verifies the accuracy of the epipolar geometry estimation on real images, Kanatani and Sugaya’s validation tests [46] rely only on simulated images and controlled noise distribution.

Shokurov *et al.* [31] used a cross-correlation approach for the feature similarity measure and combined the 7-point algorithm [47, 49] with a RANSAC perspective [32] for the refinement of the fundamental matrix with continuous update of the support set. A similar approach is presented in [12], with the purpose of removing a higher proportion of outliers, while refining the epipolar geometry estimation which is vital in reconstructing the 3D structure. This methodology [31] can be integrated in the class of hybrid strategies for feature matching, that rely on a combination of attribute-based techniques and guided-matching through the use of the known or estimated epipolar geometry. In the same

category, Beardsley *et al.* [47] developed a framework for the acquisition of 3D geometric models with the use of a triple-view vision system. In order to accurately reconstruct the 3D structure of the scene, Beardsley *et al.* [47] rely on corners and lines correspondences between the three views followed by a final matching between image primitives and the 3D structure. In all of these cases a RANSAC technique [32] is applied to robustly estimate the trifocal tensor equations and the projection matrices for correcting the erroneous correspondences and replenishing the set of matches.

The Projective Vision Toolkit (PVT) developed by Whitehead and Roth [48] is another robust tool included in the hybrid strategy for robust matching. The features extracted by the authors are SUSAN [8] corners and the initial matches are computed using cross-correlation followed by imposing the unicity [12] and symmetry [12, 38, 42] constraints to the detected matches. Additionally, another attribute-based measure is used to prune the outliers by using a relaxation-like process which is similar to the disparity gradient tool used by Vincent [12], Li *et al.* [41] and Laganière *et al.* [42]. The fundamental matrix is estimated [48] using a non-linear approach whose goal is to minimize the distance of the matches to their associated epipolar line, for obtaining consistent epipolar geometry. The interesting minimization criterion is operating on both images of the stereo-vision system, and relies on the sums of the mentioned Euclidean distances to the epipolar lines computed with the continuously estimated and refined fundamental matrix. In addition to this, the terms of the minimization criterion are weighted based on the variance of the correspondences, and the final mathematical form of the criterion coincides with Sampson's first-order correction which has been described by Hartley and Zisserman [45]. As stated by Torr [49], this correction is very useful even in the cases in which the correspondences are affected by more than one pixel noise [45] and enforces the validation of the epipolar geometry.

Finally, two different perspectives to the feature matching problem will be reviewed. Le *et al.* [25] developed a matching technique that builds upon cross-correlation, curvature and displacement as matching tools. Assuming that the sets of rigid points are related by a rigid 2-D motion, with the use of the proposed multi-scale methodology, the authors are able to extract pertinent calibration information for the acquisition stereoscopic sensor, which will guide the matching. Then the proposed curvature measure and the optical flow data computed on a frame-by-frame basis are used to form a criterion for further refinement and replenishment of correspondences. The ideas of Le *et al.* [25] to exploit optical flow information in refining the feature matches were used from a different perspective by

Mulligan [50] which suggested the use of an optical flow computation strategy [91] in guiding the feature correspondences that were further refined with an edge-based correlation.

2.2.4. Structure and Motion Estimation

The structure and motion estimation represents an important research topic in computer vision with a broad range of applications including 3D object tracking and robotic navigation in an unknown environment. From the motion perspective, there are three possible situations that are investigated in the literature. The first class is related to the case in which the vision system is static and the inspected objects are moving. In the second case the objects are static and the vision system is mounted on a mobile platform. Finally, the third class deals with the cases in which both the cameras and the objects in the scene are moving.

Based on the problem statement formulated in Chapter 1, the proposed application corresponds to the first of the classes defined above. By considering a calibrated multiple-view vision system and a scene containing a moving rigid body, the extracted feature points, together with their correspondences in the space and time domain are the principal components in the process of estimating the 3D structure and motion of the inspected object, throughout the image sequence. Therefore, in this case the problem of 3D pose estimation involves the calculation of the 3D sparse structure of the object with respect to an *a priori* selected world reference frame. In addition, the task of motion estimation is responsible with the computation of the rigid transformation (R, T) that the object has undergone between every pair of processed frames, given the corresponding 3D reconstructed point clouds. Hence, by considering a point $D(X, Y, Z)$ at time t_a which moves to the location $D'(X', Y', Z')$ at time t_b as an effect of motion of the rigid body, then:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = R \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + T = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (2.8)$$

where R is the rotation matrix and T represents the translation vector.

This sub-section reviews the most popular techniques for pose and motion estimation in the first two classes mentioned previously, since the techniques related to the second class can be extrapolated to the first class, by assigning the inverse motion of the camera to the static objects in the scene.

In the following paragraphs, research results for the problem of estimating the structure of a scene are analyzed, by considering two different scenarios. In the first setting

[43, 45], the vision system is represented by a stereoscopic sensor which is static, and the object is moving in the monitored scene. The second scenario is related to the structure from motion perspective [51-52] in which only one vision sensor is used for reconstructing the 3D sparse structure of the object in the scene.

2.2.4.1. Stereoscopic Vision

Under the first scenario, the linear triangulation procedure, described by Hartley and Zisserman [45], is used for calculating the 3D sparse structure of the scene, given the camera matrices and the feature correspondences between the stereo frames. In the cases in which the camera matrices are not available, they can be consistently constructed if the fundamental matrix can be estimated from feature correspondences between the frames [44, 45, 47, 48]. In this case the scene can be reconstructed up to a projective ambiguity [45]. The base of the linear triangulation procedure is represented by the DLT (Direct Linear Transformation) algorithm [45] which is used in estimating a homography between image points belonging to planar regions. Moreover, the two equations that are responsible with the back-projection of the 3D point $D(X,Y,Z)$ in the stereo-views, $d=PD$, $d'=P'D$, in which $d(x,y)$, $d'(x',y')$ represent the image coordinates and P , P' are the projective matrices, are combined into a compact form $A \cdot D = 0$. This compact representation is linear in the components of D and results in four equations with four homogenous unknowns that are necessary and sufficient to recover D . The proposed technique has the advantage of giving a closed-form solution by using a computationally inexpensive linear methodology. However, as pointed out in [45] the non-optimal estimated point is not exactly validating the projective equations. In order to solve these drawbacks, the authors [45] have introduced a minimization criterion for correcting the feature correspondences such that they accurately satisfy the epipolar constraint $d'^T \cdot F \cdot d = 0$, where F is the fundamental matrix. The correction scheme which is more computationally expensive than Sampson's first-order correction [45, 49], but more accurate, is applied to the initial set of correspondences, prior to the linear triangulation. As a result, an optimal 3D reconstructed point is obtained.

Inspired by the perspective of optimal 3D structure reconstruction [45], Torr and Zisserman [53] have introduced a different minimization criterion given by the sum of the squared differences between the extracted feature points in the image and their back-projections from the 3D world, through the use of the estimated camera matrices. Different from the optimization function in [45], the suggested approach involves both 2D and 3D data in the minimization process. However, as mentioned by Torr [49], this optimization approach

for structure recovery is computationally expensive. As a result, Torr preferred to use the linear triangulation procedure [49].

The mid-point triangulation, discussed by Laganière *et al.* [42], is a different approach for solving the drawbacks of the linear triangulation. One of the well-known issues in triangulation is related to the fact that the back-projected rays from the corresponding features, in the image plane, to the 3D world do not necessarily intersect, introducing uncertainty in the process of accurately identifying the 3D position of the point of interest. That is why the mid-point triangulation searches for the point which is at the middle of the segment perpendicular to both back-projected lines. This constraint, together with the set of correspondences and the camera calibration information is translated into a linear system of three equations in three unknowns which provides a closed-form solution for the point of interest [42]. As a result, this method is less computationally expensive than the optimal triangulation [45, 53]. Interested in the problem of structure estimation, Torr [49] mentioned that Sampson's first-order correction coupled with the linear triangulation [45] gives similar results to the mid-point triangulation.

A mixed structure recovery and estimation of extrinsic calibration for a stereoscopic sensor was proposed by Longuet-Higgins [43]. With the use of just eight correspondences in the stereo-frames, the author set up a linear system of equations that gives a scaled closed-form solution to both extrinsic calibration and structure estimation. The algorithm proposed by Longuet-Higgins is built upon the assumption that the corresponding rays from the two centers of projections, passing through the matched features, intersect in space, whereas in reality this condition may not always hold. Moreover, the author identified a considerable number of degenerated cases for the proposed solution and concluded that the most accurate results are obtained in the setting in which the baseline of the stereo-vision system is not very small in comparison to the depth of the objects in the scene.

2.2.4.2. Structure from Motion

Estimating 3D structure from motion represents the second scenario of the structure estimation problem that is applied for the cases in which only one static camera is used for monitoring the dynamic scene. Inspired by the framework developed by Longuet-Higgins [43], Weng *et al.* [51] introduced an algorithm for estimating the motion and structure of the scene from point correspondences between two perspective views. The proposed technique builds upon the estimation of a fundamental matrix between the two extracted views in the time domain. In this way, the second frame extracted during the tracking is considered as an

additional view in the space domain, grabbed with a virtual camera which forms a stereoscopic sensor with the single physical camera. Therefore, a methodology similar to that of Longuet-Higgins [43] can be used, with the difference that in this case, the extrinsic calibration data represents the actual rigid motion exhibited by the object in the scene, whereas the reconstructed structure can be extrapolated to both views. Similar to [43] the actual depth of the 3D structure and the translation component of the rigid motion transformation are estimated up to a scale factor. Although the proposed closed-form solution for the complete estimation problem is computationally efficient, the suggested error correction algorithm for the motion estimations is very complex. Moreover, the experimental validation proves that in order to get accurate results the camera should have a large field-of-view and the motion of the object in the scene should resemble a translation orthogonal to the image plane, similar to a far focus of expansion.

Pollefeys [52] proposed an innovative technique for recovering the 3D structure of the scene and the motion of the camera, in a setup consisting of a moving camera and a static scene. The proposed algorithm is different from the methods introduced by Weng *et al.* [51] in that the authors process a complete sequence of images, continuously updating the 3D recovered structure which is initialized through the processing of the first two extracted frames. Moreover, the process of selecting a pair of frames for processing is also supervised, such that each new image is associated with the previous frames with which it shares a high density of matches.

2.2.4.3. Motion Estimation

In the previous paragraphs the problems of 3D structure recovery and structure from motion were analyzed. The final part of this section analyzes experimental results reported on the problem of motion estimation.

Huang and Netravali [54] introduced a review of different algorithms for computing the 3D motion and structure of rigid objects, by using correspondence information of different primitives such as points, straight lines, curved lines and corners. The discussed matches are either 2D-to-3D, 2D-to-2D or 3D-to-3D feature correspondences. In the latter case, $N=4$ correspondences provide twelve linear equations (by using eq. (2.8)), which are sufficient to solve for all the unknown coefficients. Conversely, since the used data is given by “real-world” sensors, the given correspondences might be subject to errors, therefore the authors suggest using more than four correspondences for the (R, T) estimation, and reformulate the problem in a least-squares fashion (using the same notation as in eq. (2.8)):

$$\min_{\text{w.r.t. } R, T} \left\{ \sum_{i=1}^N \|D'_i - (R \cdot D_i + T)\|^2 \right\} \quad (2.9)$$

where N is the number of available 3D correspondences and $\|\cdot\|$ represents the Euclidean norm.

Holt and Netravali [55], Shariat and Price [56] have noticed that more useful information and enhancement of the motion estimation and understanding can be obtained by analyzing a sequence containing more than just two frames. In the former approach, the authors build a motion model that depends on only a few parameters which can be presumed to remain constant over a short period of time. In the latter case, Shariat and Price [56] set up a technique that builds upon the decomposition of the motion into its translation and rotation components. In this way, by subtracting the translation component from the 3D features of the rigid body, the new points will only be affected by the rotation, thus they will trace a circle in space. However, the proposed methods are constraining the freedom of motion of the inspected object, since they consider that the translation is not variable between successive pairs of uniform time intervals.

One of the most popular techniques for 3D motion estimation between two sets of 3D points was proposed by Arun *et al.* [57]. The suggested “rotation-first” algorithm is searching the least-squares solution for (R, T) , which is based on the SVD (“Singular Value Decomposition”) of a well-conditioned 3x3 matrix. In order to build this matrix, a change in coordinates is performed over the 3D point clouds, such that the translation vector is removed from the least-squares problem. The only identified weakness of the mentioned technique is related to the case in which the amount of noise in the 3D points is very large. In this situation, the authors suggest to replace the least-squares perspective with a RANSAC-like technique [32, 42] that is more robust to noise in the data.

Umeyama [58] introduced an algorithm for motion estimation with the goal of correcting in a linear fashion, the sensitivity to severely corrupted data of the method proposed by Arun *et al.* [57]. The proposed algorithm provides a more elaborated least-squares method for computing the rotation matrix, in a way in which the technique in [57] becomes a special case. The experimental validation on a numerical example shows the improved robustness of this method, when compared to Arun’s approach [57].

Kuang and Liu [59] developed a methodology for rigid transformation estimation by using Euclidean geometry. Their mathematical formulation is presented as an alternative for the cases in which the rotation estimation problem is conditioned by the orthogonality constraint [54]. As a result, the authors are interested in finding a closed-form solution for

the axis and angle of rotation. The two proposed algorithms, based on the availability or lack of the translation vector, rely on the assumption that the increments on the points on which rotation has been applied are orthogonal to the direction of rotation. The proposed algorithms were experimentally validated on a numerical example.

Yoon *et al.* [30] proposed a simple and efficient method for estimating the pose of an industrial object with respect to the reference frame of a robotic tool, on which a stereo-vision sensor was mounted. Based on the distribution of three circular macro-features being tracked, the authors assign a 3D reference frame to the industrial body based on a group of three vectors computed in the tool's reference frame. In this way, the rigid transformation between the object's reference frame and the robotic tool, and therefore the motion estimation data during tracking, can be easily computed. In order to validate the pose estimations, the object is kept stationary and the stereoscopic sensor is moved to a set of pre-defined positions in the robot's workspace. Then, based on the known inter-calibration between the stereo-vision sensor and the end-effector, the algorithm for pose estimation yields accurate results.

Interested in the problem of motion estimation, Laganière *et al.* [42] have introduced a RANSAC-like technique [32] to remove the corrupted pairs of 3-D points before giving a closed-form solution to the motion estimation problem. Random sampling is used to select triplets of 3D points that are not in a collinear configuration, which will be used in estimating unambiguous rigid transformations. Then for each of these estimated transformations, the one exhibiting the highest support set is kept and its associated pairs of 3D points are used as inputs for the algorithm proposed by Arun *et al.* [57] to obtain a final rigid transformation. Since the technique uses a stereo-vision system and a static object, the new world reference frame of the system (given by one of the cameras of the stereoscopic sensor) is updated each time, based on the computed rigid transformation. As a result, any new position is computed from the previous one, causing error accumulation. To overcome this aspect, the authors use an approach similar to the one discussed in [52] relying on the use of "loop-back" points of view that are being "re-visited".

Milella *et al.* [33, 60] minimize a mean-squared objective function (similar to the one in eq. (2.9), scaled by the number of 3D correspondences) for computing the rigid transformation that a mobile robot (having an attached stereoscopic sensor) has undergone between two processed frames. They employ an iterative scheme similar to the registration stage in the Iterative Closest Point algorithm [34]. Specifically, their outlier removal scheme relies on the dual number quaternion method introduced by Zhang [34], which is an efficient

tool for registering two sets of 3-D curves. Under these settings, the optimization process is guided by dynamical thresholding which builds upon the statistics of the registration error.

Interested in estimating the ego-motion of a mobile robot navigating in an unknown environment, Olson *et al.* [35] conclude that the stereo correspondence error has a considerably higher effect on the estimation of the 3D position of the landmarks, than the error in the actual feature tracking. For the ego-motion recovery the authors use maximum-likelihood estimation which takes directly into account the effects of the errors in the observed positions of the extracted features in both views.

2.3. Robotic Interaction with Moving Objects

The robotic interaction with a moving object under passive visual guidance represents a challenging research topic which is reviewed in this section. Since the previous section provided an insight into the research state regarding the pose and motion estimation of a moving object, the beginning of this section is dedicated to the problem of inter-calibration between vision sensors and robotic arms. The second part of this section introduces the problem of visual servo control and presents some robotic applications with moving objects in which the difficulties met by the researchers are highlighted.

2.3.1. Inter-calibration of Vision Sensors and Robotic Arms

Mulligan [50] described an integrated framework for robotic manipulation of static objects, under passive stereo-vision guidance. After identifying the requirements for the vision system to provide reliable, accurate and “timely-mannered” information regarding the pose and motion of the object with which the manipulator will have to interact, the author presents the procedures used for localizing the object in 3D space by using camera calibration data and feature matching. In order to calibrate the manipulator robot with the stereo-vision system, a 2.5cm diameter white dot is added to the robotic tool. Then, the robot is driven in over 160 positions, and based on the computed transformation between the target and the tool, together with the forward kinematic chain of the robot, the 3D positions of the target in the robot’s base reference frame are calculated. Moreover, the image correspondences are determined by blob detection. Tsai’s camera calibration algorithm [61] is used to compute the intrinsic and extrinsic calibration for the vision sensors, as well as the inter-calibration between the robot’s base and the stereoscopic system. The recommended technique has the advantage of calibrating at once the stereoscopic sensor with the robot, without requiring a separate, initial calibration for the vision system. Another

interesting aspect mentioned in the paper is the fact that the inter-calibration is used in the image processing module to limit the search range of stereo disparity according to the robotic workspace and task requirements.

Investigating the problem of calibration for many sensors in a large system, Le and Ng [62] proposed a unified framework which calibrates the entire system at once. The reasons for such an approach are linked to the fact that in the case in which the calibration is done individually, different algorithms need to be used for each sensor or pair of sensors, making the calibration difficult, time consuming and often inaccurate. Since the proposed system is composed of a stereoscopic sensor, a laser projector and a robotic arm, the authors form four sub-systems that are able to provide 3D data. Then a maximum likelihood estimation problem is being set for a joint intrinsic and extrinsic calibration for all the sub-systems at once. With only 10 checkerboard images and 5 images of a planar surface, the proposed technique is able to recover the entire set of intrinsic and extrinsic parameters by using redundant 3D-data generating sub-systems. The experimental validation shows the superiority in accuracy and computational time of this method over other approaches in which separate calibration procedures are applied to each individual sub-system.

2.3.2. Visual Servo Control for Robotic Applications

Hutchinson *et al.* [63] presented a taxonomy of visual servo control systems in which computer vision is capable of providing a closed-loop position control for a robotic end effector. After introducing two possible camera configurations, related to an “eye-in-hand” setup or a setting with the camera being fixed in the working environment, the authors classify the control systems using visual-sensory data in: position-based look-and-move, position-based visual servoing, image-based look-and-move, image-based visual servoing and hybrid combinations of these approaches. In the position-based case, the 3D pose of the target with respect to the camera is known and the feedback law is computed by minimizing the errors in the estimated pose space. That is why the position-based approach has the advantage of separating the control problem from the estimation task responsible for computing the structure and pose of the target with respect to the vision system. However, the principal drawback of this technique resides in the fact that the success of the feedback control law depends on the accuracy of the calibration data. In the latter case, the control law relies on image features. Some of the advantages of the image-based approach are given by the reduction of the computational delay or the elimination of image interpretation, sensor modeling and camera calibration. However, the controller design is

very challenging since the plant model is nonlinear and highly coupled. After developing the complex mathematical formulations for both of these general procedures, the authors concluded that the associated computational costs of both approaches are comparable. However for the case of an object moving on an assembly line as considered in the present work, its motion is simpler to express in a Cartesian reference frame, which is why the position-based visual servoing is preferred in the literature [64-67].

As already mentioned in the previous section, Yoon *et al.* [30] have proposed a robust system for pose and motion estimation of industrial objects, to work in integration with a robotic arm for the automation of industrial processes. Similar to Mulligan's conclusions [50] on the problem of robotic interaction with stationary/moving objects, the authors point out the necessity for a pose and motion estimation system whose features should be the accuracy, robustness and speed. The discussion on the accuracy of positioning the end effector with respect to the tracked object highlights the principal sources of error involved in this complex task. These are related to the various calibration and inter-calibration techniques applied to the complete system, followed by the accuracy of the pose estimation system relying on reconstructed 3D data and the capability of the system to detect and track the moving object.

The pose and motion estimator described by Yoon *et al.* [30] was integrated in a "subsumptive, hierarchical and distributed" vision-based design for smart robotics, by DeSouza and Kak [64]. The first two characteristics of the vision-based robotic architecture are linked to the possibility of one of the multiple control loops to enhance the competence level of another loop positioned in a lower hierarchic level, similar to a "coarse-to-fine" vision sensing design. The distributive characteristic of the system is related to the fact that the processing afferent to each of the control modules is totally independent of the other control loops. Under these conditions, an "arbitrator" is responsible for selecting the proper control loop (according to the criteria imposed by the specific application), which has assigned a confidence level derived exclusively from sensory information. Therefore, the total delay of the system is given by the computational time of one of the modules and not the summation of all the sub-systems' delays. As a result, apart from the enhanced fault tolerance of the scheme, the overall performance of the system is not affected by the slowest control loop, since a high confidence level is not necessarily linked to the "slowest link" in the system. These interesting features of the proposed architecture make it suitable for use with industrial automation processes. From the perspective of the robotic control law, the authors are interested in positioning the robot's end-effector in alignment with the tracked object's

coordinate frame and they describe both the position-based look-and-move and image-based look-and-move approaches [63].

Similar to the conclusions in [63], the authors [64] mention that an accurate calibration procedure for the entire system (robot, stereoscopic sensor, inter-calibrations) results in very good accuracy for the position-based visual servoing, whereas in the case of the image-based approach the control law might not always converge. However both approaches are accommodated within the proposed architecture. The distributed visual-servo application contains one coarse control loop and two fine control loops. The former makes use of a single camera for roughly estimating the object's position in the workspace, and moving the robot to an approximate location in front of the object. The two fine control loops use an eye-in-hand stereoscopic sensor and build upon the results in [30] employed with the two above mentioned approaches for visual servoing. Some of the interesting ideas behind the design of the control arbitrator share intrinsic similarities with the control task decomposition in sub-goals introduced by Davis *et al.* [67]. As a result, in the first functional mode, the coarse control provides a rough trajectory to the robot, whereas the fine controls generate the incremental displacements needed for the accurate positioning of the end-effector with respect to the object. However, a different functional mode is also presented, in which the control law is only given by a single independent control block. The two proposed experiments, one with the projection of a laser dot onto a 1cm color sticker attached to the object, and the second one linked to a "peg-in-hole" operation demonstrate the robustness and high accuracy of the proposed integrated system.

The research work presented in [64] was continued by Yoon *et al.* [65] who introduced a new tracking loop building upon model-based line matchings for a more robust pose and motion estimation of the industrial object. For that purpose, the authors use an extra camera positioned on the end-effector, between the cameras forming the stereoscopic sensor. When compared to the approach in [30, 64], the fine control loop that relied on the blob-based pose and motion estimator and image-based visual servoing was complemented with a position-based servoing approach based on a rigorous model-to-scene matching strategy. Since this new model is slower than the blob analysis technique [30, 64], the control arbitrator always selects the control law of the fastest fine control block if its confidence level is adequate. In the case in which the fine control block based on blob analysis enters a faulty mode due to occlusions, shades or changes in illumination, the arbitrator switches to the model-based line-tracking module if its confidence level is satisfactory. Furthermore, if both of these modules are faulty the coarse control mode is

selected, which positions the robot in an approximate position in the vicinity of the object. The re-iterated “peg-in-hole” experiment demonstrates the success of the proposed integrated system to a set of experiments producing a failing mode for the first fine control loop relying on blob analysis. The only identified limitation is related to the fact that a white panel is positioned behind the moving space of the object for aiding the segmentation process which is useful in both techniques for pose and motion estimation.

Building on the same system described by Yoon *et al.* [30, 65], Chang *et al.* [66] introduced a dynamic 6DOF metrology for the evaluation of a position-based visual servoing system with the help of a NIST (“National Institute of Standards and Technology”) laser. After pointing out the problem of robotic interaction with moving objects for industrial automation purposes, the authors identify the limitations which kept these approaches [64, 65, 67] from the transition to commercial products. These limitations are related to the lack of robust and objective methods for evaluating empirical performance. According to the authors, the three “must-have” features of a metrology system for visual servoing are related to the synchronization of 3D data streams, the inter-calibration between multiple sensors and the establishment of comparison metrics for associating system data with ground truth datasets. The selected robotic experiment involved the “peg-in-hole” operation described in [64, 65] by using the system architecture introduced in [64]. Two important limitations were encountered in designing the metrology system for validating the pose and motion estimations. On the one hand, the difficulty of the calibration between the NIST laser system and the vision system preempted the establishment of a full set of coordinate transformations between the two systems. Therefore, the differential motion during tracking was measured by the two systems and evaluated for consistency. On the other hand, as the stereoscopic sensor is positioned on the robot’s end-effector, the acquired data might have a certain level of uncertainty because of the physical motion of the robot, apart from the movement of the rigid object. Finally, the different test configurations illustrate the robustness of the vision system and validate its high accuracy.

Under the same framework of robotic acting on moving objects [64-66], Davis *et al.* [67] have considered a robotic assembly process in space. Under the absence of gravity, and the fair assumption that during the teleoperated assembly some of the building elements might break loose from the gripper’s jaws and tumble in space, the robot will have to catch an element, which will exert a translating tumbling motion. While the vision system, composed of a camera mounted on the robot, uses a known 3D model of the building elements to compute the pose and motion estimations during tracking, the proposed motion

prediction for robotic interaction is very interesting. Moreover, the task and trajectory planning module is built on the assumption that a complex goal can be reliably decomposed into a chain of simple subgoals which can be executed like in a static environment. More specifically, a “tagging joint” is moving along with the target, while the remaining “distal joints” are used to perform the finer details of the complex goal. For computing the trajectory of the tagging joint to a goal point in the structure of the target, the authors use a scalar piecewise quadratic interpolation function which depends on two parameters, the initialization time and the duration of the task.

An integrated system for robotic catching a moving ball was introduced by Namiki and Ishikawa [68]. When compared to the approaches described in [64-67] in which the on-line visual information is principally used in the servo control only, the proposed technique aims at employing the visual data in the higher level processing stage represented by trajectory generation. As a result, a vision-based real-time trajectory generator is advantageous in overcoming the situations in which the target motion prediction is affected by disturbance or uncertainty. The described trajectory generator relies on a nonlinear mapping of visual data to the objective trajectory, which builds upon learning based on geometric, kinematic and dynamics constraints.

Luo *et al.* [69] proposed an adaptive robotic tracking system able to intercept an object circulating with an unknown speed on the assembly line. In the design of the vision system the authors have analyzed two popular techniques, namely an eye-in-hand solution and the setting with static camera. Based on the characteristics of their general application, the authors find disadvantageous the latter setup since the vision system is obstructed when the robot manipulator is intercepting the moving object. Moreover, the use of such a vision system does not allow zooming in on an area of interest that needs to be inspected at a higher resolution. In the former case, two main benefits were identified. Firstly, the camera system is always stationary with respect to the robot's gripper. Secondly, in contrast with the latter approach, the object is always in the camera's field of view. Therefore, the authors selected a fiber-optic eye-in-hand vision configuration with remote camera location, for a suitable physical integration with the robot's gripper. Since the robot vision system only provides a 2-D description of the scene, an ultrasonic sensor is also attached to one of the gripper's fingers for the measurement of accurate range data. Based on a proper alignment of the visual/acoustic system on the gripper, the authors point out a simple homogenous matrix which links the image coordinate system with the robotic hand coordinate system. In this way, the task of intercepting the objects on the assembly line is simplified.

Vázquez *et al.* [70] and Adán *et al.* [71] have introduced a robotic system for interaction with selected points on moving objects, by using structured light sensing. Thus, the SLS (“Structured Light Sensor”) which is composed of a camera and a projector is responsible for producing range image sequences by projecting a single disordered pattern onto the moving scene. Then a group of feature points is extracted from regions of high curvatures, and tracked in the reconstructed range images. Based on the time complexity of the 3D modeling process, the object’s motion experiment has to be performed twice. In the first run, the range images are computed and the proposed algorithm generates the trajectories of the selected keypoints in the robot’s base reference frame, based on an *a priori* SLS/robot’s base inter-calibration. Then, during the second run, the object is exhibiting the same motion and the 6DOF manipulator robot is pointing the key-features. During the second run, the mean pointing error in the robot’s reference frame is computed and the results report millimeter accuracy for the pointing operation.

2.4. Chapter Summary

This chapter presented a review of some of the most important techniques and research results for the variety of tasks involved in the complex problem of robotic interaction with moving objects that are freely translating and rotating in 3D space.

The first section of this chapter was dedicated to the analysis of the constitutive blocks involved in the pose and motion estimation of a rigid object. A considerable amount of attention was given to feature extraction, feature tracking and feature matching. A sub-section investigating different structure and motion estimation approaches complemented the first section of this chapter.

The second section addressed the robotic aspect of the application of interest in this research. Interesting approaches for the inter-calibration between multiple sensors present in a robotic cell were reviewed. This inter-calibration represents a prerequisite for the success of the robotic interaction with moving targets. Finally, the problem of visual servoing control was presented along with the evaluation of some robotic platforms executing tasks in dynamic scenes.

Chapter 3. Design of the Robotic Tracking and Marking Solution

3.1. Introduction

The purpose of this chapter is to provide a description of the complete process of designing the industrial application of robotic interaction with industrial parts, under passive visual guidance, for marking of surface deformation defects. Section 3.2 introduces the rigid body models used in the experiments, which were selected with the objective of getting as close as possible to the appearance of regular automotive body panels, from the perspective of the few visual features that they exhibit.

The need of getting a complete understanding of the motion patterns exhibited by the automotive panel resulted in the design of the vision configurations requiring more than two cameras, which are discussed in Section 3.3. The challenges identified in Section 3.3, together with the prerequisite of getting closer to the particular aspects of the industrial application, from the perspective of both the appearance and dynamics of an industrial objects on the assembly line, resulted in the analysis performed in Section 3.4 regarding two-view vision configurations. Section 3.5 proposes a description of the final experimental setup for the integrated framework relying on automated surface deformation defects detection, developed by Yogeswaran [80], and robotic tracking and marking of deformations. Moreover, Section 3.5 also introduces the high-level description of the software component of the proposed solution, whereas Section 3.6 describes the central component of the autonomous robotic marking system, which is represented by the pose and motion estimator of the rigid body.

3.2. Rigid Body Models

This section describes the different rigid body models used in the laboratory experiments. The selection of the proper rigid object was guided by the particular characteristics of the application considered, that is surface defects marking for quality control in the automotive industrial manufacturing. As it was mentioned in Chapter 1, under the current industrial settings, the rigid body panels are unfinished at the stage of inspection. Therefore, the texture and color properties of their surface are not strongly contrasting or easily detectable, to help in solving the pose and motion estimation problem.

In the early stages of experimentation, the selected body structure consisted of a mobile robot to generate the movement, along with a yellow mini car hood mounted on its upper plate, as can be seen in Fig. 3.1a. Although this object exhibits a uniform color, its additional features including the black and yellow text stickers preempted its use in the next phases of experimentation, in which the goal was to get as close as possible to the specifications of the target application.

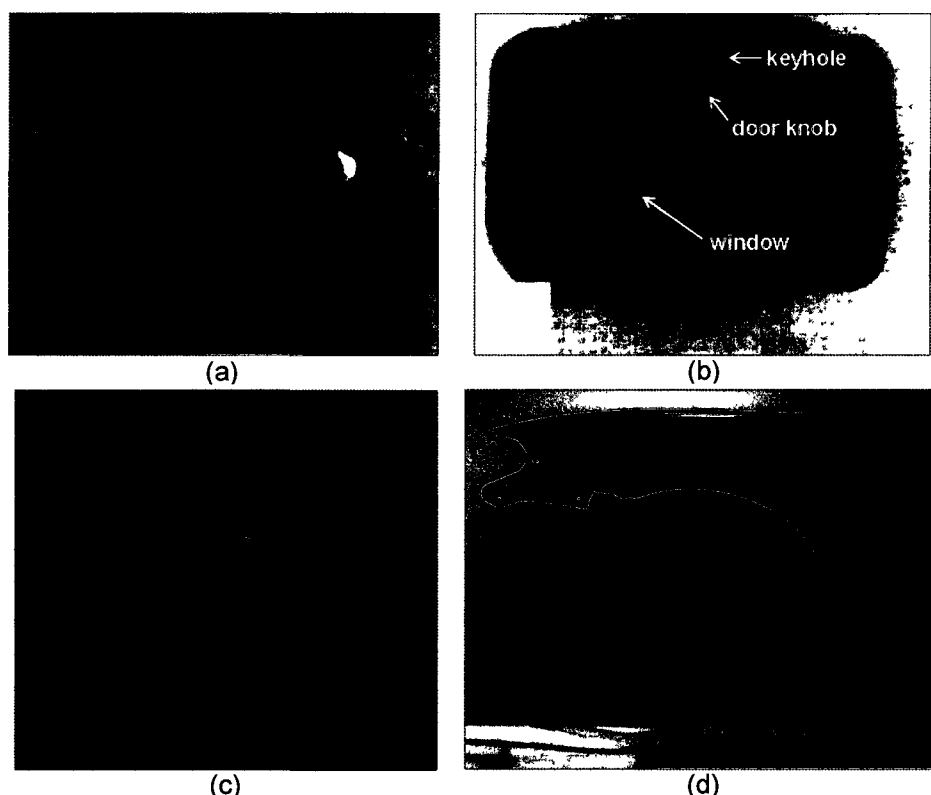


Fig. 3.1. Rigid body models used in the experimentations: (a) mobile robot with mini yellow hood, (b) mock-up car door model, (c) full-scale car door model, (d) real fender.

That is why, the following object model consisted of the same motion generator, whose upper surface was modeled as a mock-up car door, as illustrated in Fig. 3.1b. This prototype exhibited most of the visible features offered by a real automotive door panel including a simulated window space, a door knob and a keyhole. This structure did not contain other distinct features as found on the yellow mini-car hood, previously considered.

In order to overcome the vibration effects caused by the mobile robot during motion, and with the goal of reproducing as closely as possible a realistic industrial assembly process in the laboratory, a linear robotic system, shown in Fig. 3.2, was then introduced to better mimic an actual assembly line over a short length. This “sled” system, which can be operated by both a PC and a remote control, is able to move to a

specific position or to execute a displacement by a specific distance while controlling its velocity. The total length of the track is 54cm and the operational velocities range from 0.5cm/s to 2.2cm/s. Moreover, the sled's library gives the users the ability to control it from their own programs.

Given the availability of the sled for motion generation and the artificial appearance of the rigid body pointed out in Fig. 3.1b, a full-scale car door model was built for the subsequent experiments. When compared to the model in Fig. 3.1b, the new full-scale car door, shown in Fig. 3.1c, better reproduces the generic characteristics of any typical car door at an early stage of manufacturing, including a smoothly curved surface as well as the inner and outer frames of the window opening. In addition to this, the door model also features a door handle and some appended deformation defects.

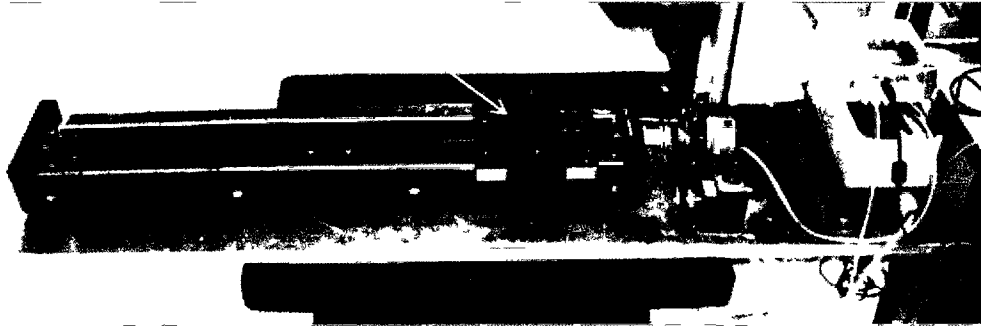


Fig. 3.2 Linear robotic system used to generate the motion of the car door

The latest car door model is attached to the mounting base of the sled system, pointed out in Fig. 3.2. Based on the relative positioning of this system with regards to the vision sensor, different car door orientations can be acquired. In this way, the real world scenarios under which the car doors, circulating on the assembly line, exhibit different orientations with respect to the vision system at different stages of their trajectory, can be successfully tested with the current platform in spite of the lack of a dedicated rotating plate. Finally, Fig. 3.1d presents a real fender model which is used for validating the generality of the pose and motion estimation system on a different type of automotive panels which share the same drawback of having a small number of distinctive visual features. As a result, the two selected panels, shown in Fig. 3.1c and 3.1d provide a reliable insight into the appearance and amount of features present over the surface of automotive panels.

3.3. Initial Vision Configurations with Multiple Cameras

A rigorous experimental process was followed to design the most suitable configuration for a multi-view vision system in the context of on-line quality control. For

that purpose, a number of different settings have been evaluated. The starting premise was the fact that merging information from several vision sensors could bring reliable data about the complete motion of the automotive panel, in any direction [72]. As a result, one of the first configurations considered made use of three Point Grey Flea2 IEEE-1394b CCD cameras with 3.5mm lenses and 640x480 pixels of resolution, which were positioned as illustrated by the VRML ("Virtual Reality Modeling Language") model in Fig. 3.3a. Under this setup, Cam_1 represents the world reference frame, and the goal of this camera configuration was to ensure a proper coverage of the object's entire trajectory in order to better estimate its motion patterns. Specifically, the right and left lateral sides of the part are imaged by Cam_1 and Cam_2 , whereas Cam_3 is located in the upper part of the scene and it provides a complete view of the object.

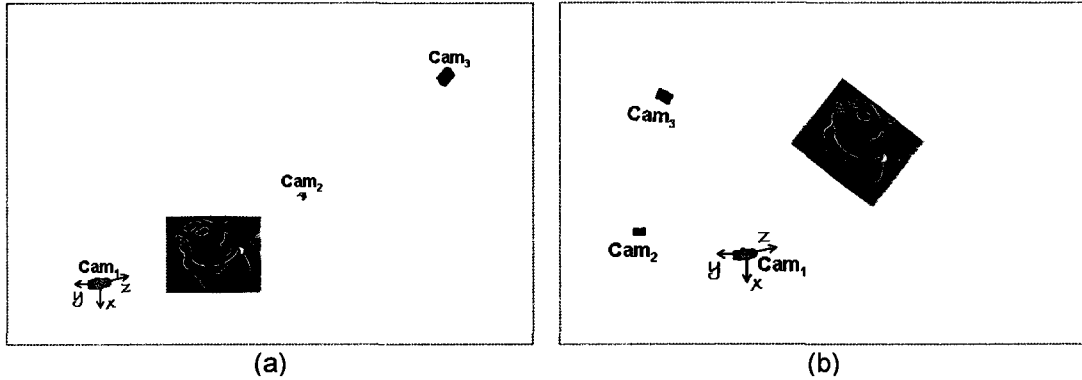


Fig. 3.3. Initial vision configurations with three cameras: (a) highly separated in 3D space, (b) grouped on the lateral side of the workspace.

The drawbacks observed with this configuration preempted its selection as a proper setup for the target application. More specifically, one of the most important limitations was linked to the relatively low amount of overlap between the cameras. Additionally, the low resolution of the cameras and the complex backgrounds in the monitored scene, which largely differed between the views, also affected the performances of this vision system. The feature matching problem in the case of the proposed industrial objects, characterized by low density of features, was even more complicated by the large spatial separation ($D_{Cam1 \rightarrow Cam2} \approx 287\text{cm}$, $D_{Cam1 \rightarrow Cam3} \approx 233\text{cm}$) between the three viewpoints, which resulted in a considerable level of perspective distortion between the grabbed frames. The coupled effects of all these limitations caused a poor accuracy for the pose and motion estimation of the moving object.

The possibility of correcting some of the identified drawbacks of the multiple view setup shown in Fig. 3.3a, guided the experimentation toward the selection of a better distribution of the cameras in the workspace. Therefore, the next tested vision

configuration, illustrated in Fig. 3.3b, represented a trifocal system, that provided a lateral view of the moving object. This acquisition configuration directly addressed the limitation originating from the small amount of overlap between the views, at the cost of narrowing down the possibility of the system to offer a better understanding of the part's displacements from perpendicular views. Conversely, the size of the real automotive body panels, and the necessity for the cameras to grab a significant amount of visual data in which the object fully appears in the three views, imposed constraints on the spatial distribution of the vision sensors. As a result, the system had to continue maintaining a substantial baseline between the three views, ($B_{\text{Cam1} \rightarrow \text{Cam2}} \approx 145\text{cm}$, $B_{\text{Cam1} \rightarrow \text{Cam3}} \approx 80\text{cm}$). This condition, together with the limitations identified with the previous setup, forestalled the final selection of such a generic configuration.

A different vision configuration, whose objectives were to preserve the coverage given by the first configuration considered, while simplifying the feature correspondence problem, is shown in Fig. 3.4.

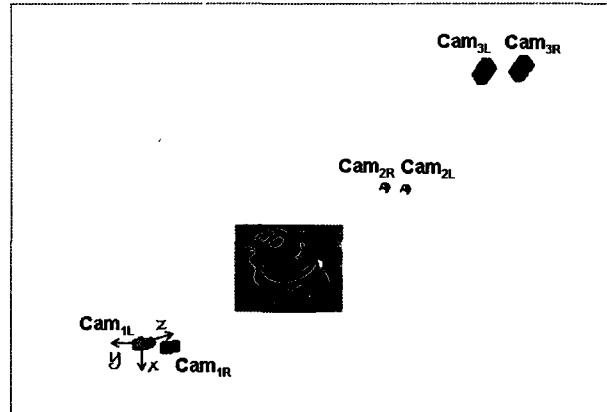


Fig. 3.4. Vision configuration with three short baseline stereo-vision sensors.

As it can be noticed, each of the cameras in the early setup were replaced by stereoscopic systems with short baselines. In this way, each of the stereo-vision sensors is able to provide a sparse structure estimation of a certain portion of the object and all this information can be successfully merged in a robust motion estimation process. Although this method demonstrated its superiority with regards to the configurations previously discussed, the computational complexity and the hardware costs preempted the final selection of such an elaborated approach. Moreover, a careful analysis on the complete set of motion patterns that can be performed by the automotive part in the context of an assembly line [72], together with the specific constraints of the marking of surface deformations, led to the design of a setup consisting of only one stereoscopic pair, which will be introduced in the next section.

3.4. Proposed Vision Configurations with Only Two Cameras

A thorough investigation regarding the motion patterns that the automotive body might exhibit on the assembly line provided the basis for selecting the most appropriate stereo configuration for the quality control application of interest. The objective of getting a considerable insight into the complete motion of the panel, which was the main goal of the setup in Fig. 3.3a, has been extrapolated to a two-view configuration illustrated in Fig. 3.5.

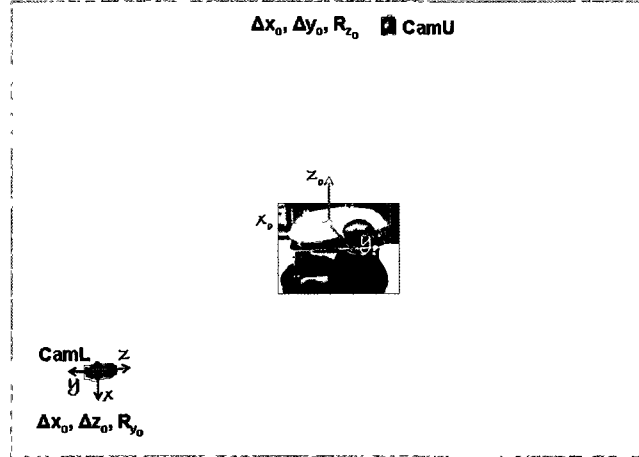


Fig. 3.5. Vision configuration with two orthogonal cameras (CamL and CamU).

In a first attempt with only two cameras, one of them was positioned over the inspected part and pointed downward, while the second camera collected a lateral view of the moving body. By taking into account the reference frame attached to the object, as illustrated in Fig. 3.5, the lateral camera, CamL, provided reliable information regarding the translation of the part along its X_o and Z_o axes, and the rotation around the Y_o axis, R_{Y_o} . Additionally, the upper camera, CamU, was able to give a consistent insight into the translation of the object along the X_o and Y_o axes, together with the rotation around the Z_o axis, R_{Z_o} . This configuration was also able to measure the rotation of the rigid body around its X_o axis. However, in the cases in which the pose and motion estimator relies on a fixed set of features on the structure of the rigid part, a rotation around its X_o axis will result in a gradual change of the feature set which will be replaced by new keypoints detected on the visible patch of the inspected object. Nevertheless, this type of rotation is less likely to happen under assembly line settings.

Similar to the configurations in Section 3.3, in order to apply stereo-triangulation for recovering a sparse 3D structure of the object, the feature correspondence problem had to be solved. As expected, the major limitation of the vision system in Fig. 3.3a, related to the low amount of overlapping regions in between the views, extrapolated in

the case of the configuration shown in Fig. 3.5. Thus, it resulted in a direct impact on the number of features than could be extracted and matched. To illustrate this effect, Fig. 3.6 presents two frame segments grabbed with the configuration in Fig. 3.5 during the movement of the object. Apart from the low overlap, the other limitations identified in the previous section, related to the low number of features present on the surface of the object and the difficulty of performing feature matching between views that are affected by such a considerable level of perspective distortion, proved this approach to be inadequate to pursue the development.

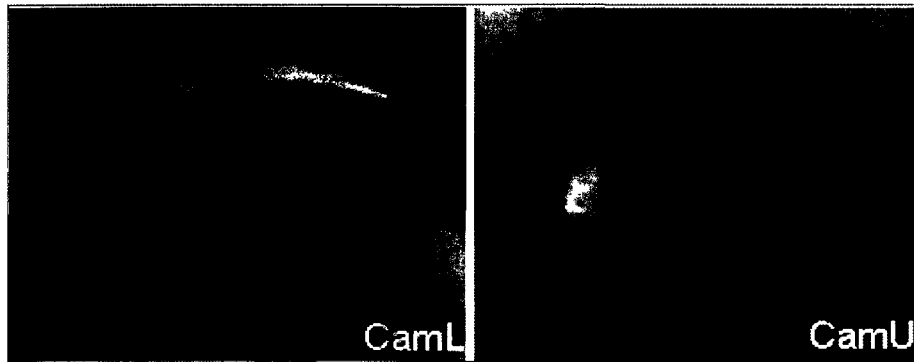


Fig. 3.6. Segments from the frames grabbed by CamL and CamU during the tracking.

Considering the reference frame of the object, shown in Fig. 3.5, the dominant components of movement, under the settings of an assembly line, are the translations along the X_o and Y_o axes, together with the rotation around the Z_o axis, R_{zo} . The upper camera, CamU, is able to provide insight about all these displacements characterizing the motion of the rigid body. As a result, given the prerequisite of relatively straight motion imposed by the assembly line and the objective of minimizing the limitations identified with the setup of Fig. 3.5, the acquisition system was modified to measure the motion information with a single stereo-vision configuration located above the assembly line and pointing downward, as shown in Fig. 3.7a. Under this configuration, the limitations of low overlap, perspective distortion and matching difficulty were consistently reduced. In Fig. 3.7a, the reference frame of the stereoscopic sensor is assigned to CamR with the optical axis, Z , pointing downward. Additionally, the rigid body was changed with the mock-up car door model illustrated in Fig. 3.1b in order to better resemble to a real car door panel. According to the reference frame of the car door, illustrated in Fig. 3.7a, the X_oY_o plane defined the surface of dominant motion of the object. Therefore, the motion pattern of the panel, positioned at an approximate depth $Z_{\text{CamR}} \approx 180\text{cm}$ from CamR, was mainly a translation along the X_o axis, which could also be coupled with displacements along the Y_o axis and minor rotations around Z_o axis.

Following experimentation, a baseline, $b \approx 25\text{cm}$, between the two cameras was found to provide better results than the cases with closer vision sensors, under the same configuration. This was also enforced by the well-known fact that the larger the baseline the better the depth estimation [36, 42, 45]. The experimentation conducted on the mock-up car door shown in Fig 3.1b, revealed that the stereo-vision configuration illustrated in Fig. 3.7b, with its principal axis pointing perpendicularly to the object and the main direction of motion, represented the most suitable acquisition strategy for estimating the pose and motion of the automotive part circulating on the assembly line [72].

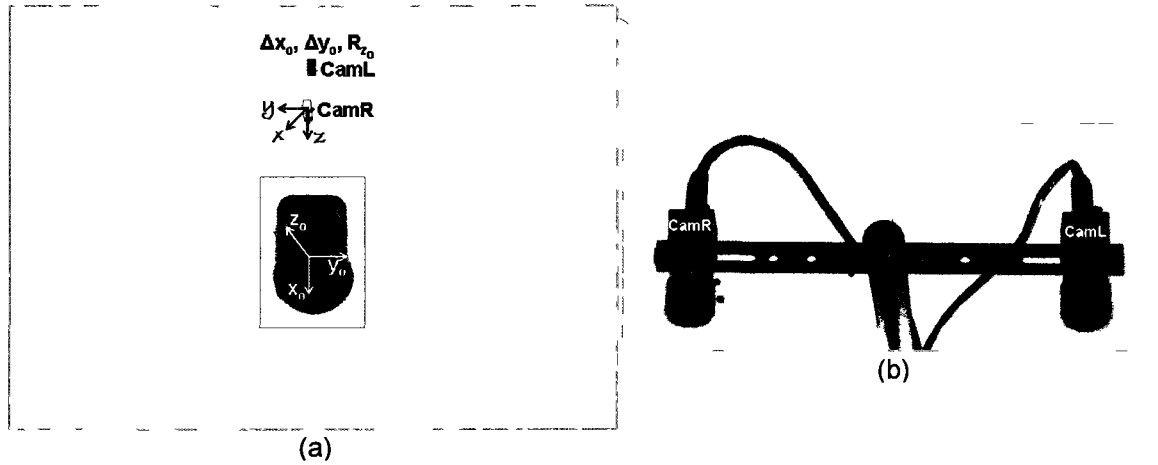


Fig. 3.7 Ceiling mounted stereo sensor: (a) cameras configuration, (b) stereoscopic sensor

As explained in Section 3.2, the fact that the mobile robot considered at that point, whose upper face was designed to mimic a car door, was manually driven by a remote control, resulted in vibration effects during the movement. Additionally, the use of the full-scale car door model, shown in Fig. 3.1c, resulted in another modification to the design of the final vision setup, whose configuration is illustrated in Fig. 3.8a. Under these settings, the stereo-vision sensor in Fig. 3.7b was moved to the lateral side of the experimental work cell which is now composed of the sled system, shown in Fig. 3.2, and the thin car door prototype, whose principal face cannot be inspected by a ceiling-mounted vision system. Moreover, the stereoscopic system, illustrated in Fig. 3.8b, is preferably located almost perpendicularly to the panel's surface, whereas its world reference frame, O_R , is attached to the right camera, CamR. In addition to this, a baseline, $b \approx 44.5\text{cm}$, was experimentally determined between the cameras, since it provides improved accuracy in reconstructing the sparse structure of the object. The rigid car door structure, which is positioned on the 54cm sled system acting as an assembly line, is located at approximately $D_{\text{CamR}} \approx 310\text{cm}$ from the acquisition system.

The selection of such a considerable distance to the cameras was guided by two objectives. Firstly, the necessity of analyzing a sufficient amount of visual data in which the object appears in both views is crucial for robust motion estimation and visual servoing. Secondly, the high occupancy levels that can be present in a car manufacturing cell, including multiple collaborative robots, also motivate the choice of a substantial distance between the stereo-vision sensor and the work scene.

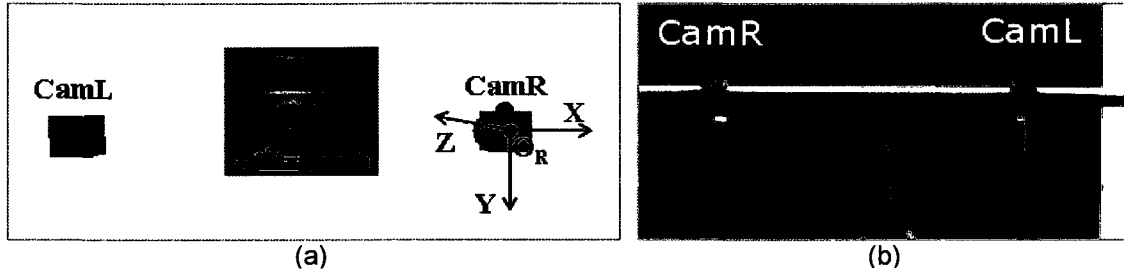


Fig. 3.8. Final stereo-vision sensor: (a) cameras configuration, (b) physical stereo-vision system.

Under this last configuration, the acquisition system is able to reliably measure the movement of the car door in the XY plane defined with respect to CamR, which resembles the motion pattern that the rigid object should exhibit on an industrial conveyor. Thus, these two final systems, shown in Fig. 3.7a and Fig. 3.8a are found to be the most appropriate configurations from the perspective of the proposed application of surface deformation defects marking for autonomous quality control in the automotive industry. Apart from the gradual improvements achieved in the process of reproducing the automotive inspection cell in the laboratory, the design of the acquisition system was guided by the need to minimize the impact of the limitations identified with the previous configurations.

The selection of the lenses has been driven by the need to determine the largest focal length, in order to minimize the distortion effects while still being able to observe the entire motion sequence in both views. For the defined baseline, a pair of 8.5mm CCTV lenses are used in the final setup. Furthermore, the decision of fully calibrating the stereoscopic sensor was guided by three important considerations. Firstly, the inspected scene does not contain a high level of details, that would be required if an approach involving the estimation of the fundamental matrix from a consistent set of pre-computed feature point correspondences had been selected [12, 45, 47, 49]. Moreover, even with such solutions there is a need for providing the system with some real world measurements in order to compute the scale factor [45] and acquire exact 3D reconstruction. Secondly, the need for precise pose and motion estimation is imperative for the proposed application which involves robotic interaction with moving panels.

Finally, the stereo-vision system is static in its environment, therefore the calibration procedures can be performed only once, before starting the pose and motion estimation process.

Under these conditions, the intrinsic calibration, which relies on a framework developed by Bériault [76], is computed for each camera using several checkerboard views as proposed by Zhang [77]. The intrinsic matrix and distortion coefficients are calculated and saved in a calibration file for each camera. The extrinsic calibration [76] builds upon a combination of the techniques proposed by Chen *et al.* [78] and Ihrke *et al.* [79]. The procedure consists of waiving a stick of a known size, with two attached LEDs, in front of the synchronized cameras. The algorithm uses each extracted pair of frames to compute the location of the lights in the image planes and then estimate the relative position between the cameras. Then, the known scale factor, defined by the distance between the LEDs, permits the conversion to real world units, in order to quantize the translation components of the rigid transformation between the two vision sensors. Finally, one of the cameras is considered as the world reference frame and the rigid transformation between the two vision sensors is appended to the calibration files.

3.5. Description of the Complete Robotic Work Cell

In this section, the experimental platform developed for surface deformations marking on automotive body panels is detailed. The complete framework [75] for the surface deformation defects detection and marking is shown in Fig. 3.9, where the functional block for this part of the research is highlighted in blue. The other major block of this integrated solution, developed by Yogeswaran [75, 80], is responsible for automatically detecting the surface deformation defects and will only be briefly described here.

Firstly, a 3D imaging module for data acquisition, which consists of a structured light sensor (SLS) [74, 81] generates a colored dense 3D reconstruction of the surface profile of the panel under inspection. The SLS uses of a stereoscopic pair of cameras and projected lighting under the form of a bi-dimensional pseudo-random color pattern which is scrolled over the surface [74, 75, 81]. The embedded stereo-vision system is composed of two Lumenera CCD Cameras and 8.5mm Computar lenses. The maximum resolution of the cameras is 1392x1040pixels and the maximum frame rate at this resolution is 15fps. The pattern projected onto the object creates a set of artificial feature

points that compensate for the lack of prominent keypoints, as usually occurs over the surface of a typical automotive panel during assembly.

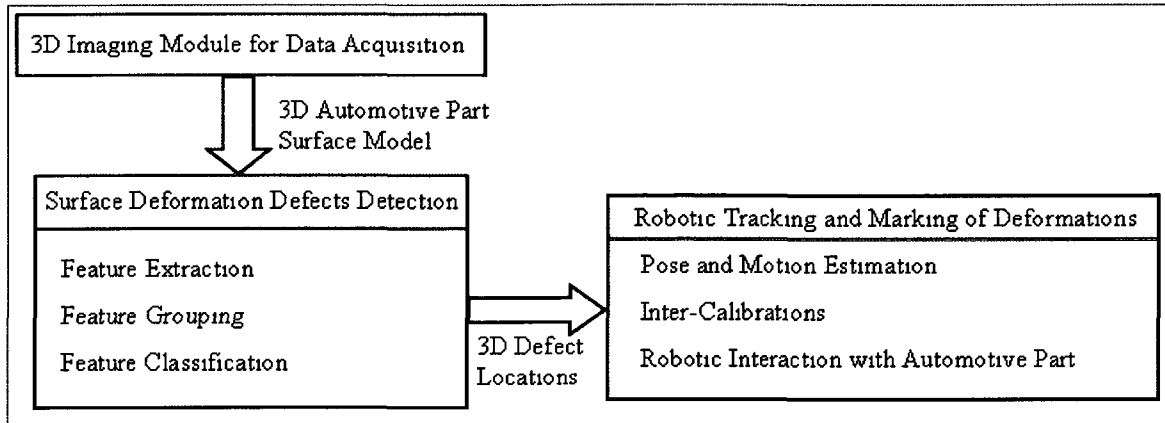


Fig. 3.9. Complete deformations detection and marking framework.

Secondly, the 3D surface model of the object provides the input to the surface deformations detection sub-system which is responsible for 3D feature extraction, grouping and classification. Finally, the output generated by this sub-system groups the 3D locations of the surface deformations, expressed with respect to the left camera of the SLS sensor, CamL_{SL} . A complete view of the integrated surface deformations detection and marking system is presented in Fig. 3.10.

In the prototype, the SLS is integrated in the same work cell as the stereo-vision sensor used for estimating the pose and motion (SSPME) of the automotive part. As a result, the two platforms associated with the deformation detection and the robotic marking respectively, are integrated in a compact architecture due to the limited space available under laboratory settings. One of the limitations caused by such an integrated configuration is linked to the low level of parallelization between the two processes. However, the principles of the proposed integration and the particular characteristics of the two sub-systems permit the transition to a two-station approach in which the marking operation can be performed at a subsequent stage from the defects detection procedure. Thus, the deformations detection system can process data acquired on the following panel while the marking is executed over a panel previously inspected, using two separate stations along the assembly line. This would directly improve the efficiency and production rates.

As shown in Fig. 3.9, the robotic marking sub-system estimates the pose and motion of the panel on the assembly line, and performs the path planning to guide the marking. In order to guarantee consistent movements between the inspected panel and the robot's end-effector, two inter-calibrations need to be performed. The first one

involves the computation of the rigid transformation between the right camera (CamR) of the SSPME and the base reference frame of the robot, O_B , as shown in Fig. 3.11a. The second calibration relates CamR with the reference frame of the SLS, $CamL_{SL}$. The approach selected for this latter inter-calibration does not rely on the fact that the SLS and the SSPME coexist within the same work cell, and can be successfully extrapolated to the general framework with the operations distributed over two different stations. Since the positions of the surface deformations are defined with respect to $CamL_{SL}$, the inter-calibrations make it possible to transfer these deformation locations into the robot's reference frame, to guide the marking operation.

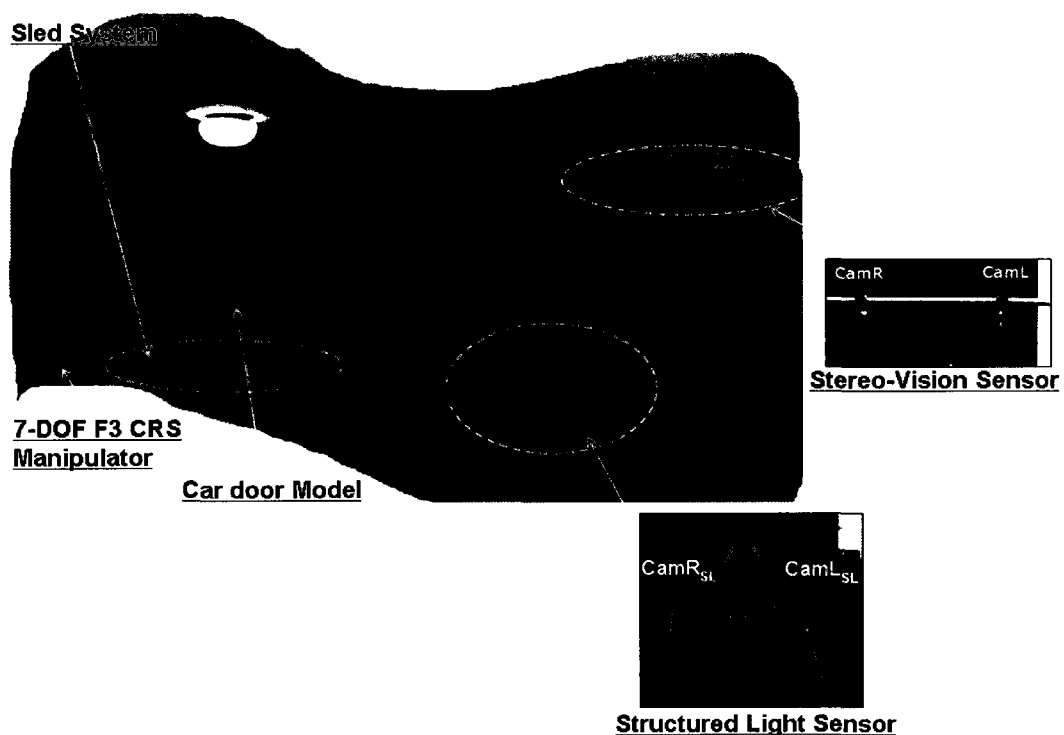


Fig. 3.10. Complete experimental setup for autonomous deformations detection and marking.

The actual interaction with the automotive panel is performed by an F3 manipulator from CRS Robotics [82] that has 6 revolute joints, and is mounted on a 2m CRS track [83]. The 7-DOF F3 CRS serial manipulator is controlled by a CRS C500C digital programmable controller [84], which embeds several communication methods, including the RS-232 asynchronous link that permits the integration with the current controlling computer with no need for additional communication adaptors. The C500C controller also includes a version of the inverse kinematics of the robot, which is essential for converting a 3D location, defined with respect to the robot's base or end-effector, to individual joint displacements. The software provided by CRS, Robcomm3,

embeds a compiler for the RAPL-3 robotic programming language [85], allowing customized applications to be programmed in order to control the robotic system. For the current application, one of the scripts written in RAPL-3 language includes a method for accepting user input over the serial communication for controlling the robot. Additionally, different tool prototypes were added to the robot at various phases of testing, as seen in Fig. 3.11a and Fig. 3.11b.

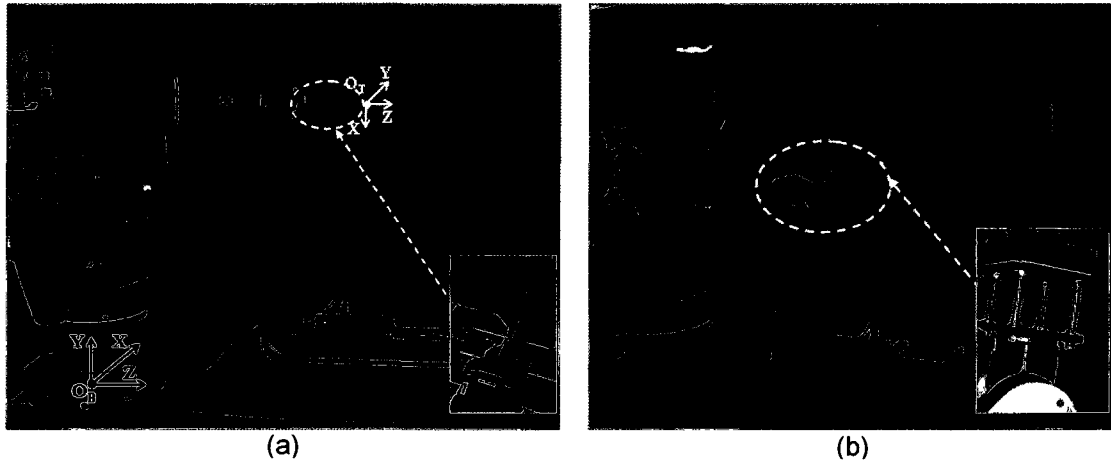


Fig. 3.11. Manipulator robot with different tool prototypes: (a) pointing tool, (b) stamping tool.

The first pointing tool, shown in the bottom-right segment of Fig. 3.11a, was used with an initial prototype in which the objective was to point the deformation areas over the surface of the car door panel, in order to validate the accuracy of the robotic marking system. Figure 3.11a illustrates the reference frame attached to the pointing tool, O_T . The second type of end-effector is a stamping tool, displayed in the bottom-right part of Fig. 3.11b, and retained in the final configuration, with which the robot stamps the surface defects by actually getting in contact with the static door panel. This stamping end-effector uses a spring-loaded plate as its tip, which allows a spring compression of approximately 1.5cm. In this way, the small amount of error that might be present in the depth components of the locations computed for the identified deformations is compensated by the compliance of the hardware. This preserves the integrity of the automotive part during the robotic interaction.

3.6. General Framework for Pose and Motion Estimation

This section introduces the high-level structure of the proposed pose and motion estimation solution which is the central component of the robotic tracking and marking system, highlighted in Fig. 3.9. In the first part, the proposed analysis investigates the challenges identified in the process of estimating the pose and motion of industrial

objects. As a result, the most appropriate methodology for pose and motion estimation is selected, and its basic principles are established. In the second part, a high-level description of the overall selected process of pose and motion estimation is presented to outline the most important blocks of the proposed solution.

3.6.1. Pose and Motion Estimation Approach Selection

The pose and motion estimator, shown in Fig. 3.9, represents one of the fundamental components of the robotic system working in interaction with moving bodies. As discussed in Section 2.3.1, the characteristics that a vision system should have, in order to be effectively integrated within a visual servoing robotic application, were classified by Mulligan [50] and Yoon *et al.* [30] into three categories. Specifically, the pose and motion estimator has to run in a “timely manner”, especially when the visual guidance data is to be incorporated within a framework involving robotic acting on moving objects. Secondly, the information regarding the appearance and dynamics of the inspected objects should be reliable and extremely accurate. Finally, the safety and availability attributes of the vision system must be considered as well, since an effective autonomous system should exhibit fault tolerance.

The goal of this section is to evaluate these three key characteristics, against the challenges and particular aspects related to the considered industrial settings, in order to determine the best solution for the pose and motion estimation of the automotive panel. In the next paragraphs, the challenges identified in the current research will be discussed, by referring to the automotive part models illustrated in Fig. 3.1c and Fig. 3.1d, together with the final passive stereo-vision system presented in Fig. 3.8. Additionally, the assembly line displacements are reproduced by the linear robotic (sled) system displayed in Fig. 3.2.

3.6.1.1. Challenges and Constraints Analysis

As mentioned in Section 3.2, one of the most important challenges of the conducted work is related to the general appearance of the target industrial objects, which exhibit a low density of sharp and unique features visible over their surface. Additionally, in order to maximize flexibility, the pose and motion estimator needs to run without an exact 3D CAD model of the inspected objects. As a result, the conducted investigation extends the usual bounds of computer vision, which usually assumes richly textured objects, by addressing a practical problem that is still under extensive research.

Chang *et al.* [66] have identified a second challenge, related to robotic systems relying on visual servoing data, which consists of the lack of robust and objective methods for empirically evaluating the pose and motion estimations of the rigid body. In addition to this, independently of the selected tracking approach, the visual processing system has to triangulate a group of points belonging to the inspected part in order to estimate the pose of the body and its associated inter-frame rigid motion. To achieve this, the stereo-correspondence problem has to be solved. This represents another challenge of the current research, since the tracked objects exhibit uniform color patterns, which introduces ambiguity in the matching process. Also, the necessity to estimate the motion for a considerable amount of time imposes constraints on the baseline of the stereo-vision system, introducing perspective distortion between the views. Additionally, the properties of the target objects have a major impact on the methodology involved in the tracking of the industrial part.

In order to simplify the segmentation of the industrial object from the background, Yoon *et al.* [30, 65] make use of a white panel that is positioned behind the tracked part, represented by an engine cover, on which a robotic arm performs a classic “peg-in-hole” experiment. Therefore, the generality level of the proposed solution [30, 65] is noticeably affected from the perspective of industrial integration, as factory settings are usually characterized by complex backgrounds. In addition to this, the pose and motion estimation system proposed by Yoon *et al.* [30, 65] requires the existence of at least three circular regions over the structure of the rigid body, which might not be the case for a large variety of automotive panels. Another difficulty is related to the high occupancy levels present in a factory environment. Under the current research, the manipulator robot will appear in the scene since the overall goal is to mark the surface deformation defects. Moreover, the possibility of temporary appearances of different other objects or even factory attendants in the view is also a reasonable assumption. These aspects have a considerable impact on the tracking and matching processes, since the pose and motion estimator should exhibit robustness to other entities, present in the scene, while remaining focused on the target industrial object. For this reason, the pose and motion estimation solution should embed a supervisory layer that will have to continuously monitor the feature extraction, tracking and matching processes in order to provide reliable visual servoing information to the robotic station. However, the supervisory sub-system should balance the trade-off between the precision of the pose and motion estimation versus computational complexity in order to allow for real-time processing.

3.6.1.2. Background Subtraction Approach

The general pipelined implementation of the assembly and quality control tasks in an industrial manufacturing setting supports the assumption that the passive stereo-vision sensor can grab a pair of frames before the appearance of the rigid body in the workspace. Consequently, the use of background subtraction approaches seemed appropriate for the selection of a region-based pose and motion estimation solution. However, two important limitations preempted the selection of such an approach.

The first category of limitations was related to the low resolution of the cameras, the lack of an exact 3D CAD model of the rigid body, and the small variations in the reflectance properties of the industrial objects' surface. Moreover, the inspected part might also have welding spots or marker lines, indicating regions to be processed in future assembly operations, which have a strong impact on the segmentation process.

The second category of problems was related to the complexity of the factory background, as well as the occupancy level of the scene monitored by the passive stereo-vision sensor. In order to illustrate this effect, under laboratory settings, Fig. 3.12 shows two resized frames grabbed by CamL (displayed in Fig. 3.8) during the tracking cycle. Figure 3.12a displays the experimental scene at the beginning of the tracking cycle, whereas Fig. 3.12b illustrates a frame in which the robot and two attendants appear in the monitored scene.

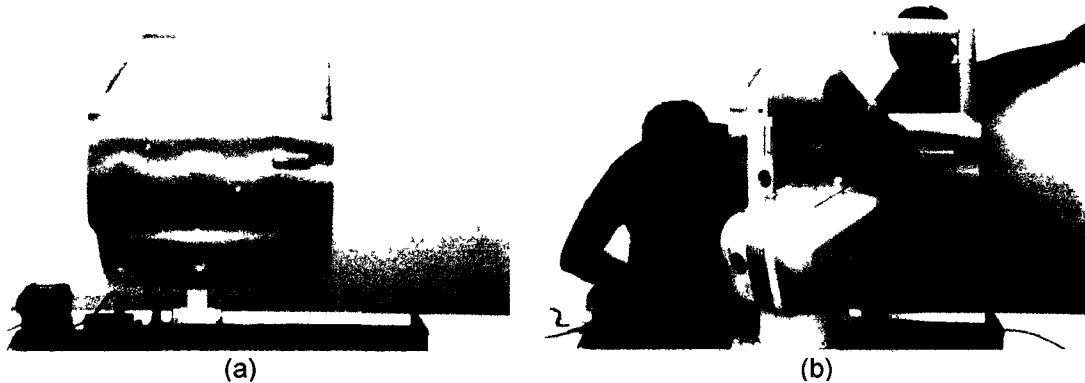


Fig. 3.12. Scaled frames grabbed by CamL, during the tracking sequence: (a) beginning of the tracking cycle, (b) when the complexity of the monitored scene is increased.

The two categories of identified limitations had a strong effect on the thresholding needed for background subtraction and caused a segmented car door model exhibiting multiple holes over its structure. Moreover, the region matching in the stereo-frames, the tracking process, as well as the selection of keypoints that needed to be triangulated in order to get some insight into the 3D structure of the objects became very difficult under

these conditions. As a result, the effects of the challenges associated to the particular characteristics of the current industrial application had a strong effect on the possibility of the region-based pose and motion estimator to provide reliable, fast and safe visual servoing data to the robotic station.

Although the region-based approach was not selected for further experimentation, the background subtraction results, obtained at the start of the tracking sequence, sustained its integration in the process of triggering the pose and motion estimation, when the industrial object had fully appeared in the inspected scene. Consequently, the background subtraction results were able to provide a rough estimate regarding the position of the target industrial object, and the assignment of a ROI (Region of Interest) needed for further selective visual processing.

3.6.1.3. Feature-based Pose and Motion Estimation Approach

By taking into account the constraints of the current research and their effects on the success of the region-based approach, the necessity for time-efficient [50] visual servoing data resulted in the selection of a feature-based pose and motion estimation methodology. This technique relies on a pre-selected set of features associated with the panel's structure, which can be uniquely identified and consistently tracked on a frame-by-frame basis, along the inspection work cell [72, 75].

The early version of the proposed feature-based pose and motion estimation framework involved the selection of the strongest keypoints in the structure of the rigid body, by using the Shi and Tomasi corner detector [6], whereas the tracking was performed by the pyramidal implementation of the LK feature tracker [20, 22]. The only validation gate applied to the computed motion vectors was related to the "tracking error" intrinsically calculated under the OpenCV's implementation [21] of the feature tracker. Specifically, this error residual is related to the extent of which the image patch containing the tracked feature changed throughout the motion experiment. As mentioned by Jin *et al.* [36], this measure is sensitive to the differences in the intensity variation of the patches of interest. Nevertheless, the intensity variations during the movement of the object might have a higher magnitude than the difference between the feature patch and other regions in the structure of the inspected scene [35]. As a result, in the cases of slight illumination changes or variations in the reflectance properties of the object's surface ("highlights" or specular reflections), the "tracking error" becomes very high, complicating the thresholding process and decreasing the tracking robustness.

Additionally, as discussed in Section 3.6.1.2, and exemplified by Fig. 3.12b, there are possibilities for the inspected scene to become very complex during the motion sequence. Consequently, the small level of supervision associated to the feature extraction, tracking and matching processes, affected considerably this early pose and motion estimation approach, particularly its robustness and precision, which are vital for the subsequent robotic integration.

In order to overcome these difficulties, which resulted from a point-based pose and motion estimation, Yoon *et al.* [30, 65] used two different perspectives for providing the visual servoing system with the needed accuracy, robustness and fault tolerance. On one hand, the authors introduced some *a priori* knowledge about the structure of the target objects, which motivated the transition to a pose and motion estimation system relying on three circular features. However, the blob detection process was simplified under the proposed controlled background. On the other hand, as mentioned in Section 2.3.2, the hardware architecture relied on three vision systems, in order to assure proper robotic integration.

In the current research, the goal is to maintain the same vision configuration shown in Fig. 3.8, and to monitor the feature extraction, matching and tracking processes, from a software perspective, by adding a supervisory layer to the system. This supports the transition to a “supervised” feature-based pose and motion estimation system, in which a minimum amount of *a priori* knowledge about the appearance of the tracked objects is provided to the system. When compared to [30, 65], the proposed solution aims at addressing the limitations mentioned earlier, with no need for controlled background, wireframe 3D model of the target object, or complex hardware architecture. Under these conditions, the point-based pose and motion estimation approach takes the form of a “hybrid” methodology, which also embeds geometrical information, vital in the process of supervising the feature tracking and matching procedures.

The “supervised” perspective over the pose and motion estimation process represents an imperative requirement for proper robotic integration, and was successfully embedded in the visual servoing solution. The proposed pose and motion estimation approach builds upon a minimum number of macro-features (MFs) that give a signature to the topological geometrical structure of the target industrial object. These MFs, which can be extracted with a maximum of stability from the structure of the rigid body, constitute the central component of the overall visual servoing methodology, since

they also have implications in the object detection module, as well the inter-calibration between the SLS sensor and the SSPME, showed in Fig. 3.10.

3.6.2. High-Level Description of the Pose and Motion Estimation Solution

The next paragraphs introduce the major components included in the proposed “supervised” feature-based pose and motion estimation solution, whose high-level block diagram is illustrated in Fig. 3.13. To begin with, an important aspect of the proposed solution is related to the way in which the *a priori* knowledge regarding the general appearance of the objects being tracked is provided to the system. Specifically, the pose and motion estimation prototype relies on the pre-selection of a minimum set of macro-features (MFs) over the structure of the rigid body, which is manually performed by the installation engineer, only once, when configuring the inspection station for a specific type of object. These few MFs are represented by corner points which are classified as distinctive, upon the visual analysis of the automotive object. Under the mass production settings of industrial manufacturing, this initialization is a reasonable procedure.

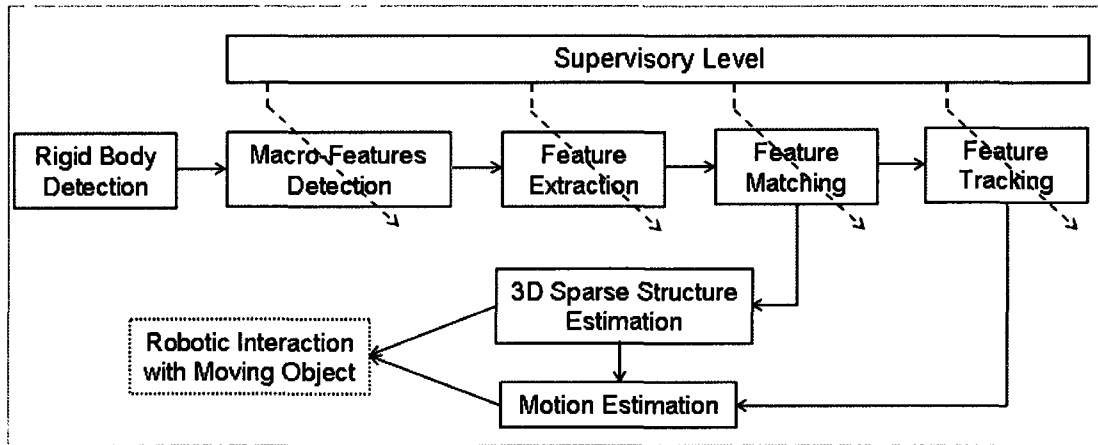


Fig. 3.13. High-level block-diagram of the proposed pose and motion estimation solution.

The pose and motion estimation system is capable of reconfiguring itself every time a new entity of the same type of object appears on the conveyor, even with slight changes in orientation and position. The frames grabbed by the stereoscopic sensor at initialization, as well as the approximate positions of the MFs are stored on the hard-disk, since they are useful for the automated reconfiguration of the system.

The initialization data is used by the first block in Fig. 3.13 which deals with the detection of the rigid body in the monitored scene. For this reason, the object detection module builds upon background subtraction, as mentioned in Section 3.6.1.2, in order to inform the pose and motion estimator about the full appearance of the panel in the view

of the two vision sensors. Following the detection of the object, the re-localization of the MFs is performed by applying the pyramidal implementation of the LK tracker [20, 22] for computing the approximate MFs' correspondences between the initialization frame stored on the disk, and the current view, under which the industrial body might exhibit slight changes in orientation with respect to the acquisition sensor. The robustness of the MFs detection module to the cases in which the pyramidal LK tracker provides erroneous optical flow vectors or incomplete tracking results is assured by the supervisory layer, which continuously monitors the set of MFs in order to provide a stable signature to the tracked automotive part.

The next two sub-systems in Fig. 3.13, are related to the feature extraction and matching procedures, which are the prerequisites for the 3D reconstruction process. Once the feature matches are computed, the full camera calibration data is used in the procedure of triangulation, which gives a sparse 3D structure of the object, at every processed frame. The pyramidal implementation of the LK tracker [20, 22] is used for the computation of the optical flow data belonging to the extracted features, during the motion cycle. Based on these motion vectors, a new 3D point cloud is generated for the current frame, and the motion estimation process is triggered. The information regarding the shape and the motion of the panel, computed from image data, is provided to the robotic station that will perform the marking task on the moving object.

As mentioned in Section 3.6.1.3, a supervisory layer is added to the pose and motion estimation system with the purpose of monitoring the feature extraction, matching and tracking processes. The latter are the components which play a fundamental role in the robustness and precision of the visual servoing system. As a result, the combination of geometrical information with the feature-based tracking methodology, are the main characteristics of the supervisory level which provide the robotic marking station with the capability to achieve accurate and time-efficient pose and motion estimations of the rigid body.

All the functional blocks illustrated in Fig. 3.13 will be described in detail in the next two chapters, together with their experimental validation results.

3.7. Chapter Summary

This chapter proposed an extensive investigation of the challenges and constraints identified in the process of designing the complete industrial application for robotic marking of surface deformations on moving automotive body panels, under

passive visual guidance. The first part of this chapter analyzed the hardware components of the industrial application whereas the high-level software framework was described in the second part.

Specifically, the first part was initiated by the description of the panels used in the laboratory experiments. Then, an empirical analysis regarding the suitability of vision configurations with more than two cameras for the target application was conducted. This led to a formal design process of a stereo-vision configuration appropriately positioned in the scene, for providing a reliable insight into all the motion patterns that the automotive part can exhibit on the assembly line.

Subsequently, a description of the complete experimental cell for automated surface deformation defects detection and marking on automotive body panels for quality control in industrial manufacturing was presented. The latter helped define the prerequisite for the high-level description of the software components needed to drive the application.

Finally, the high-level software framework analysis was performed that resulted in the selection of the most suitable pose and motion estimation solution for providing robustness, precision and fault tolerance to the robotic marking system.

Chapter 4. Feature Extraction, Matching and Tracking Processes

4.1. Introduction

This chapter introduces an analysis of the selected methodologies for the feature extraction, matching and tracking processes embedded in the proposed feature-based pose and motion estimation system, whose high-level diagram is illustrated in Fig. 3.13. As discussed in Section 3.6, the pose and motion estimation solution builds upon the pre-selection of a set of MFs, which are the most stable keypoints over the surface of the inspected object. Moreover, as it will be explained in Section 4.2, which is dedicated to the MFs' selection and object detection procedures, the approximate MFs' pre-selection process is performed only once, when the robotic marking system is configured to work in interaction with a certain automotive part. As a result, the proposed pose and motion estimation system is able to re-initialize itself for the cases in which new automotive part items, belonging to the same category, appear on the assembly line for their inspection cycle, under mass-production settings. The knowledge provided to the system by the MFs' set is also successfully embedded in the object detection module.

The objective of Section 4.3 is to identify the most proper feature extractor for the challenging task of estimating the pose and motion of industrial objects which suffer from a lack of contrasting features over their surface. Thus, a correlated stability-robustness empirical measure is introduced with the purpose of selecting the "best" feature detector to be embedded in the pose and motion estimation solution.

The principles of the feature tracking process, relying on the pyramidal implementation of the LK feature tracker [20, 22], are introduced in Section 4.4, along with the challenges of the proposed quality control application. These challenges, related to the general appearance of the inspected panels and the complexity of the monitored scene have a strong impact on the robustness of the feature tracking results, which are fundamental in the motion estimation process.

Finally, the feature matching procedure, which builds upon the pyramidal LK tracker [20, 22] and Shi and Tomasi's corner detector, is described in Section 4.5.

4.2. Macro-Features Selection and Object Detection

The challenges identified for tracking industrial objects, described in Section 3.6, resulted in the necessity of pre-selecting the most important keypoints over the surface of the automotive panel. These MFs provide the system with a stable topological structure of the inspected part which is very useful in the object detection procedure, as well as the supervision of the feature tracking and matching processes. The goal of this section is to describe and experimentally validate the MFs selection and object detection modules, which are the first two functional blocks of the high-level structure of the proposed pose and motion estimation solution, shown in Fig. 3.13.

4.2.1. Selection and Refinement of Macro-Features

The manual pre-selection of the MFs is performed by the installation engineer when the pose and motion estimation system is mounted and calibrated for inspecting a given type of automotive body part. Specifically the MFs pre-selection sub-system is manually triggered upon the full appearance of the rigid body in both stereo-views, whereas, at the same time, two synchronized initialization frames are acquired by the stereoscopic sensor. Subsequently, a graphical user interface (GUI) showing the frame grabbed by CamL is provided to the installation engineer, who manually selects the approximate locations of the MFs over the surface of the automotive part. The goal of the pre-selection of MFs is to provide the system with the most stable keypoints, which are present over the structure of all automotive parts belonging to the same type of part. Under these settings, the corner extraction process, which will be described in Section 4.3, becomes insensitive to particular features of individual items such as welding spots or markers, which might be unreliable to track.

In the case of the car door model illustrated in Fig. 3.1c, the selected MFs belong to the inner and outer frame of the car door's window. As a result, Fig. 4.1a shows the location of the MFs as they were pre-selected, together with their associated 9x9 pixels image-windows which are used in the process of refining these initial selections. The dimension of these image-windows was experimentally selected, based on the size of the panels in the image plane, when imaged from a distance of approximately 310cm, as mentioned in Section 3.4. Moreover a size of 9x9 pixels for these image-windows also provides more liberty to the initial approximate selection performed by the installation engineer, which may not pinpoint exactly the location of the strong features.

A total of ten MFs were also selected for the fender model, shown in Fig. 3.1d, as it can be noticed in Fig. 4.1b, where the refinement windows are also displayed. From the total set of MFs, seven of them belong to the contour of the fender, whereas the other three MFs are associated to the circular regions present over the surface. Moreover, the MFs are stored into a matrix structure according to their associated numerical index, which is marked in Fig. 4.1.

In order to correct the approximate positions of the pre-selected MFs, the proposed refinement stage builds upon the Shi and Tomasi corner detector [6] and its OpenCV implementation [21], for extracting keypoints in both stereo-views. However, the corner extractor is not applied on the full-size image grabbed by CamR, but only over a region of interest (ROI) which is a bounding box around the pre-selected MFs' area.

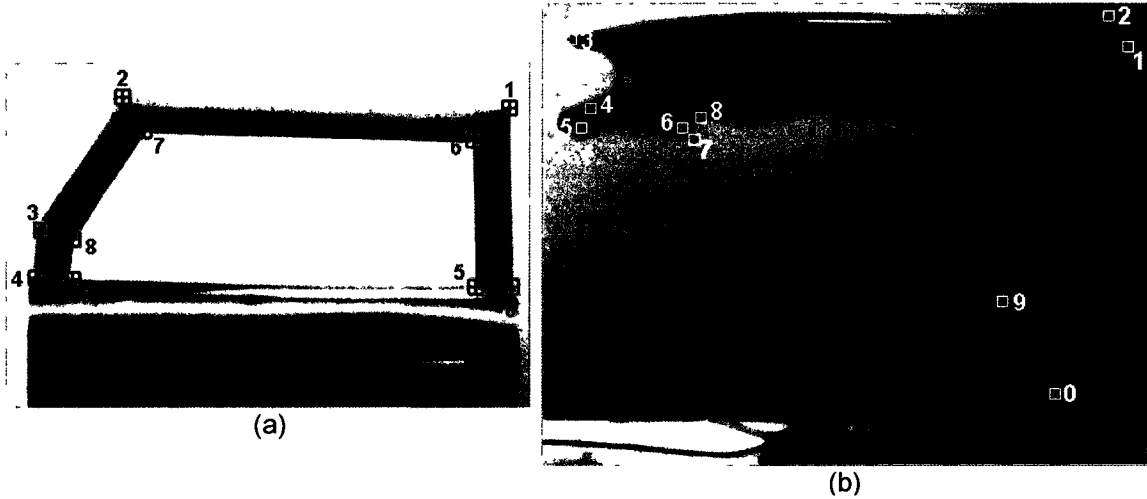


Fig. 4.1. Initially selected MFs over the surface of the: (a) car door, (b) fender.

The procedure for the extraction of the ROI builds upon the initial selection of MFs, which are processed in order to calculate the top-left starting point of the ROI, along with its associated width and height. Specifically, by analyzing the location of the MFs, the minima ($\min_x$, $\min_y$) and maxima ($\max_x$, $\max_y$) positions on the x , and y , axes of the image plane are identified. Then the top-left starting point of the ROI is provided by the pair ($\min_x - r$, $\min_y - r$), where r is a reserve of 20 pixels which is subtracted from the minima positions, with the purpose of slightly centering the MFs' area within the extracted bounding box. Subsequently the (width, height) of the ROI are given by ($\max_x - \min_x + r$, $\max_y - \min_y + r$).

According to the overall spatial distribution of the defects marking stage, shown in Fig. 3.10, the automotive part appears in the scene from the left side, therefore CamL will have a full view of the rigid body before CamR, since CamL is actually located on the

left hand side of CamR and therefore closer to the side of arrival of the part. Moreover, since the objective of the pose and motion estimation system is to characterize the movement of the automotive part for as long as possible over the conveyor, it should be noted that in the case in which the same ROI extraction approach is applied on the frame acquired by CamR, the selection of the reserve r , is dictated by the physical constraints of the image plane, as the top-left starting point of the ROI cannot have negative coordinates. The ROI's extraction is used to make the transition to a "selective" feature extraction process which is not affected by the background complexity. This has beneficial effects on the thresholding procedure, since the highest density of extracted keypoints is expected to reside on the surface of the industrial object.

The Shi and Tomasi feature detector [6, 21] with subpixel accuracy is applied on the extracted ROI with a threshold $T_{\text{Shi_Tomasi}}=0.02$, whereas a minimum distance of 9 pixels is imposed on the returned corners. In this way, the process of refining the user-selected MFs is simplified, since the minimum distance constraint avoids the detection of multiple keypoints within the 9x9 pixels windows centered on the approximate MFs selections, which are represented in Fig. 4.1. Thus, the proposed refinement procedure consists of the search for the corners extracted with the Shi and Tomasi feature detector within the windows assigned around the initially pre-selected MFs. For the case of the car door model, Fig. 4.2a shows the detected features in the extracted ROI containing the MFs' area. Figure 4.2b illustrates the refined MFs within their assigned 9x9 pixels patches.

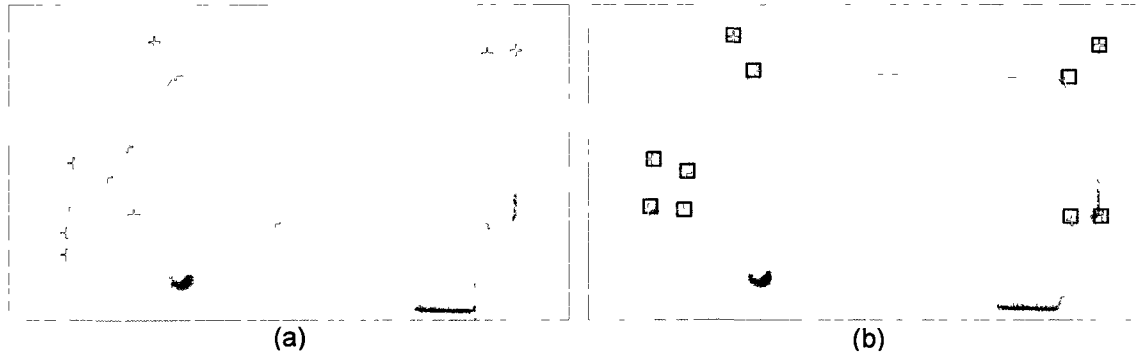


Fig. 4.2. MFs refinement for the car door model: (a) extracted features with the Shi and Tomasi corner detector, (b) refined MFs.

Once the MFs are refined in the frame grabbed by CamL, their correspondences in the image acquired by CamR, need to be computed. For doing this, advantage is taken of the fact that the cameras forming the selected stereo-vision configuration are approximately parallel and the rigid transformation between their assigned reference

frames is mainly a translation along the X axis of CamR, as shown in Fig. 3.8b. Under these conditions, the pyramidal implementation of the LK tracker [20, 22] can be used for guiding the correspondences between the two stereo-views. Thus, the pyramidal LK tracker is currently applied on adjacent frames in space. The spatial distribution of the cameras supports this approach, since the LK tracker [20] builds upon a translational model for characterizing the transformation that affects the feature patches throughout the tracking process. Nevertheless, the returned optical flow vectors are now called disparity vectors since they adjoin the synchronized stereo-frames.

In order to account for the considerable baseline between the cameras, a depth of 7 is selected for the image pyramids, whereas the size of the search window is 7×7 pixels. The OpenCV's implementation [21] of the pyramidal LK tracker provides a tracking residual for all the optical flow vectors in order to quantify the difference between the intensities of the feature patches in the two processed frames, as mentioned in Section 3.6.1.3. Under these settings, a threshold of $T_{\text{residualLK}}=350$ is used in the validation process responsible for the outlier disparity vectors' removal, at the level of the MFs detection block, embedded in the high-level structure shown in Fig. 3.13.

The supervisory level, highlighted in Fig. 3.13, is responsible for improving the robustness and accuracy of the overall pose and motion estimation prototype. The implementation of the supervisory layer in the MFs detection block is motivated by the possibilities that the validation procedure does not completely remove the outliers, or that the disparity vectors are not found for all selected MFs. Nevertheless, based on the topological structure of the MFs in the frame grabbed by CamL, and the characteristics of the stereoscopic sensor, the supervisory layer also monitors the overall distribution of the MFs in the frame of CamR, at the level of initialization. The complete details of the supervisory procedures allotted to the MFs detection block will be described in Section 5.4.3.

In the case of the car door model, the computed optical flow vectors are shown in Fig. 4.3, which illustrates two segments belonging to the synchronized frames grabbed by CamR and CamL at the initialization stage, when the installation engineer performs the MFs pre-selection. As it can be noticed from Fig. 4.3, all disparity vectors point to the proper MF region, since all the optical flow vectors have passed the validation phase in the initialization stage.

With the purpose of further correcting these approximate correspondences, the same methodology that was applied in CamL's frame for refining the pre-selected MFs,

is employed on CamR's image as well. Thus, a ROI is extracted based on the approximate MFs' matches, following the same procedure described for CamL's frame. The only difference is that the reserve, r , is selected in a way in which the ROI is not extending farther than the actual physical space of the frame grabbed by CamR. Then, the proposed refinement stage relying on the Shi and Tomasi corner detector [6, 21] with sub-pixel accuracy is applied on the ROI which results in the refined MFs, identified in 9x9 pixels windows centered on the approximate correspondences.

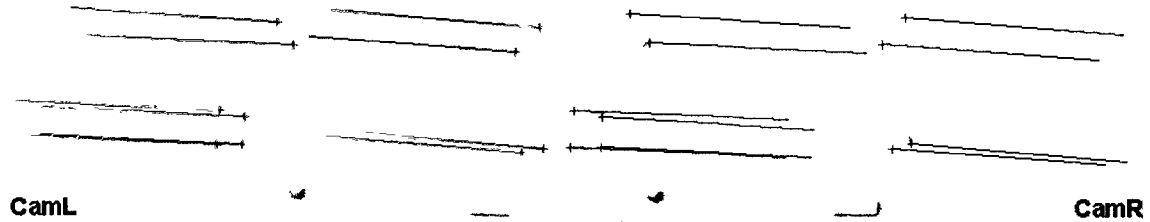


Fig. 4.3. MFs disparity vectors computed with the pyramidal LK tracker for CamL/CamR's initialization frames of the car door model.

Figure 4.4a represents a frame segment from the initialization frame grabbed by CamR, in which the refined car door's MFs are marked. Additionally, in Fig. 4.4b are illustrated the refined MFs belonging to CamR's initialization frame, for the case of the fender model. The locations of the MFs in the two views, along with their associated initialization frames, are stored in an initialization folder. All this information will be used in the MFs' structure re-initialization and object detection processes, which will be described in the subsequent sections.

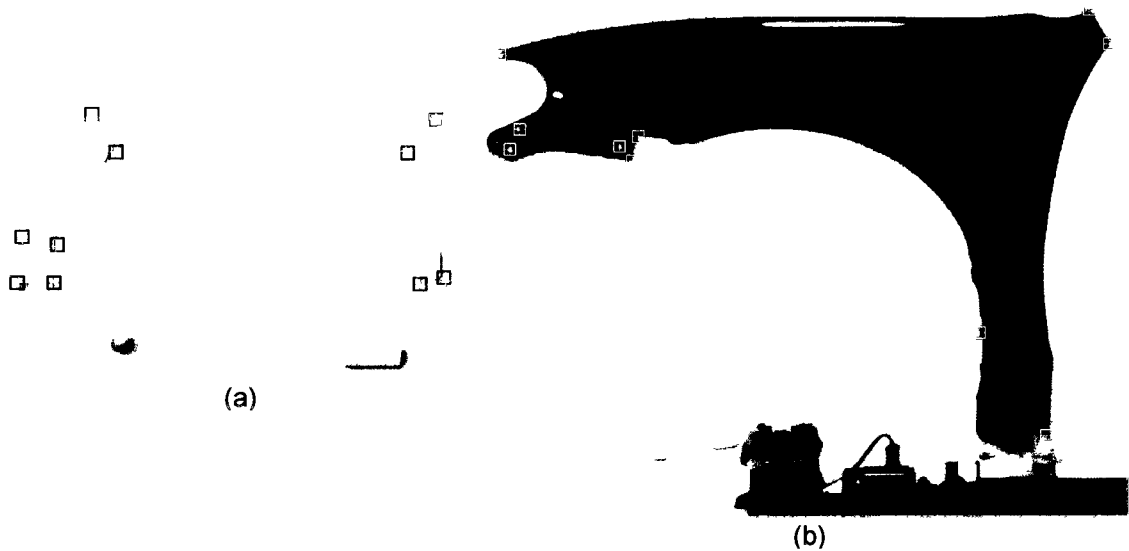


Fig. 4.4. Refined MFs in the initialization frame grabbed by CamR, in the case of the (a) car door model, (b) fender model.

4.2.2. Re-initialization of the Macro-Features Set

As mentioned in Section 3.6, the manual selection of MFs is executed only once when the inspection station is configured, whereas the system has the ability to re-initialize itself every time a new item appears on the assembly line, for it to be processed. The re-initialization process is triggered by the object detection module, which will be described in detail in Section 4.2.4. In order to validate the re-initialization procedure, two scenarios were considered under the assumption that industrial objects might enter the scene with slight differences in position and orientation with respect to the initialization case. Table 4.1 introduces the data related to the changes in the pose of the industrial object at re-initialization for two testing scenarios, in which θ_{Y_o} represents the rotation angle of the industrial body with respect to the Y_o axis of the reference frame attached to the rigid body at initialization, O_o , illustrated in Fig. 4.5.



Fig. 4.5. Image segment from the initialization frame grabbed by CamL, with the reference frame O_o attached to the rigid body.

Table 4.1. Pose changes of the rigid body model at re-initialization.

Rigid Transformation Data	Scenario 1 (with door)	Scenario 2 (with fender)
θ_{Y_o}	10°	-12°
Δ_{X_o} (cm)	3	3
Δ_{Z_o} (cm)	-10	10

Subsequently, Δ_{X_o} and Δ_{Z_o} are the displacements along the X_o and Z_o axes of O_o , from the initialization position, in which the MFs were pre-selected by the installation

engineer. As it can be seen in Table 4.1, Δ_{x_0} has a smaller magnitude than Δ_{z_0} since the goal of the object detection module is to trigger the re-initialization as soon as the industrial object fully appears in both stereo-views in order to analyze its motion pattern for a considerable time span.

In the first scenario, the selected rigid body is the car door model, while the pose changes of the second scenario are applied to the fender. Similar to the approach used for estimating the MFs' correspondences between the stereo-views, the re-initialization procedure makes use of the pyramidal implementation of the LK tracker [20, 22] and the Shi and Tomasi corner detector [6, 21].

Specifically, the initialization frames, collected at system's configuration, and their associated MFs' positions, together with the newly extracted frames, called the re-initialization frames, under the two scenarios, are the inputs of the pyramidal LK tracker, which is applied twice, that is once for each vision sensor. In this way, the newly extracted frames are registered with the initialization images. Then, the rough matches between the initialization and current frames are further refined by using Shi and Tomasi's corner detector [6] with sub-pixel accuracy [21], within the same refinement technique presented in Section 4.2.1. The motion vectors computed with the pyramidal LK tracker, under both scenarios, are illustrated in Fig. 4.6, for the frames grabbed by CamL.

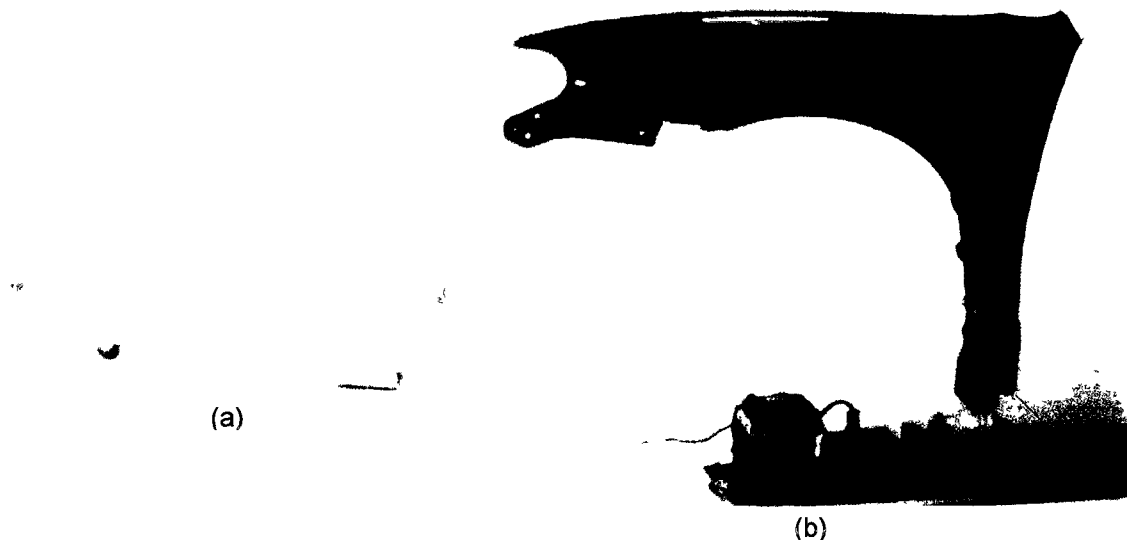


Fig. 4.6. Motion vectors computed by the pyramidal LK tracker during the re-initialization process:
(a) scenario 1, (b) scenario 2.

Nevertheless, the supervisory level, illustrated in Fig. 3.13, also performs the monitoring and correction needed for the current registration and refinement process

needed for the re-initialization of the MFs' structure. Particularly, the supervisory layer provides the MFs' re-initialization sub-system with the needed robustness for the cases in which the registration between the initialization and re-initialization frames contains erroneous motion vectors, or the optical flow vectors for some MFs are discarded in the current validation gate, relying on the tracking residuals. These monitoring and validation gates will be described in detail in Section 5.4.3.2. Finally, experimentation has shown that the re-initialization of the MFs' set, with the embedded implications of the supervisory layer, performs satisfactorily for all the scenarios in which the pose changes are bounded within the following ranges $\theta_{y_0} \in [-30^\circ, 30^\circ]$, $\Delta_{z_0} \in [-45, 45]$ cm.

In this way, the stereo-vision system used for estimating the pose and motion of the automotive part can also be integrated in a work cell in which the assembly line is curved in the region in which the object will appear at re-initialization. Thus, in these cases, which might be imposed by the factory's space restrictions, the changes in the pose of the rigid bodies at re-initialization might vary between slightly larger bounds than in the regular case involving a straight assembly line setting.

4.2.3. Topological Structure Buffers

Once the re-initialization procedure is finished and the current automotive part item is reliably registered with its associated prototype, the information regarding the position of the MFs, which records the topological structure of the tracked industrial object, is stored into two types of buffers which are important in the monitoring phase of the feature tracking and matching processes, embedded in the supervisory layer.

The first category is represented by the buffers of 2D distances which are populated with the Euclidean distances between each MF and all the other MFs which form the extracted MFs' set on the same object and in the same view (re-initialization frame), as shown in Fig. 4.7a for the car door model. The Euclidean distances are computed in 2D, using the image coordinates of the detected MFs, for the two stereo-views respectively.

Secondly, the displacements in the x and y directions between each MF and all the other MFs present in the detected set are stored in the buffer of relative distances, as shown in Fig. 4.7b for the car door model. These two classes of buffers characterize the object's 2D topological structure in both stereo-views of the acquisition system.

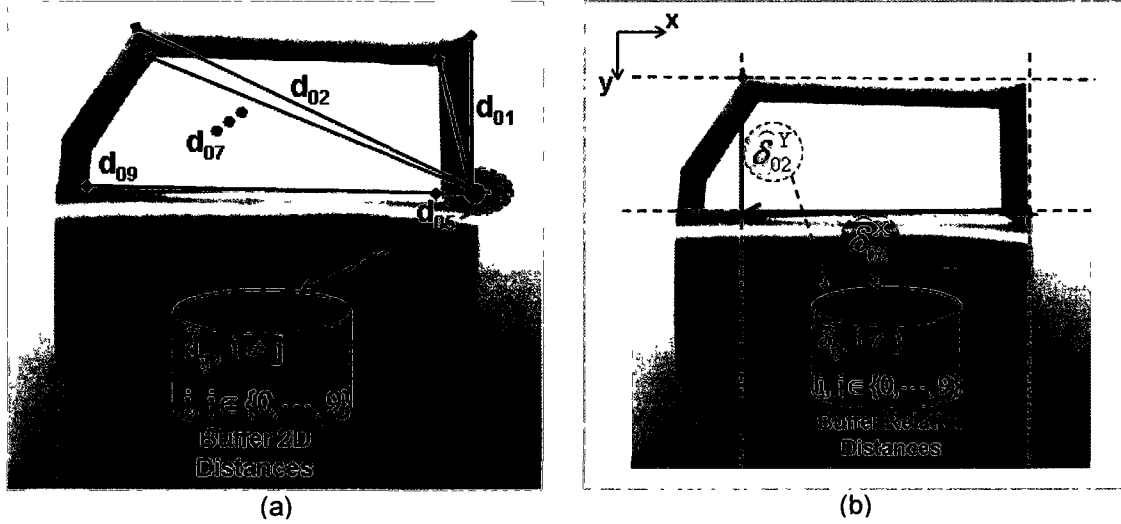


Fig. 4.7. Topological structure buffers for the car door model: (a) buffer of 2D distances, (b) buffer of relative distances.

4.2.4. Object Detection

The goal of the proposed object detection module is to achieve high precision and low computational complexity. Instead of selecting complex approaches which rely on color histograms and statistical modeling of the background and foreground, the proposed approach builds upon a “guided” background subtraction methodology, which makes use of the pre-selected topological structure of the automotive part, performed at system initialization. Therefore, apart from the advantages of the MFs’ pre-selection in the selective feature extraction and monitoring of the feature tracking and matching processes, which will be further detailed in Section 5.4, the MFs’ topological structure is also successfully integrated in the object detection module.

Therefore, prior to the initialization procedure performed by the installation engineer and the arrival of the automotive part in the field of view of the cameras, a set of synchronized frames, called background images, is acquired by the stereoscopic sensor. Then, at the end of the MFs’ pre-selection process, the absolute difference between the initialization and background frames is computed for each view. The resulting images are further processed by imposing a threshold $T_{\text{diff}}=0.15$ for the ratio between the intensity of each pixel and the maximum intensity of the corresponding image of absolute differences. Specifically, the pixels exhibiting an intensity ratio greater or equal than the threshold are assigned the value “1” and correspond to the regions which are now occluded by the new entities in the scene, while the others receive the

value “0”. The resulting binary images, associated to both initialization frames, are shown in a scaled version in Fig. 4.8.

As seen from Fig. 4.8b, and already mentioned in Section 4.2.1, the automotive part item appears first in the field of view of CamL, therefore the object detection module needs to be sensitive to the full appearance of the rigid body in the view of CamR, which corresponds to the moment in which the pose and motion estimator has to be triggered. The proposed object detection technique adds a signature to the pre-selected MFs in order to exhibit robustness to the saturation effects manifested by the reflectance of the objects’ surface, as well as the cases where the automotive part item enters the monitored scene with slight changes from the initialization pose. Nevertheless, in the quality control application considered here, it is assumed that the scaling effects, caused by the appearance of the industrial items slightly closer to the cameras, or the displacements with respect to the initialization pose, are not substantial.

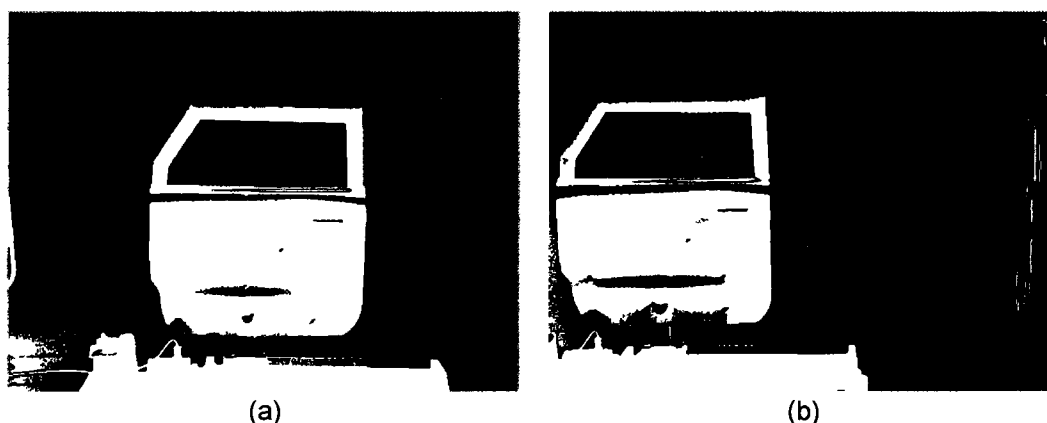


Fig. 4.8. Scaled binary images resulting from the background subtraction methodology in (a) CamL, and (b) CamR's initialization frames

For the binary image shown in Fig. 4.8a, which is associated to CamL's initialization frame, 21x21 pixels patches are centered on each of the refined pre-selected MFs. These patches are extracted and provide the signature of the MFs. The size of these patches was experimentally selected, based on the dimensions of the panel in the image plane, as well as the necessity of the object detection system to be robust to the cases in which the current pose of the panel is different from the pose considered at initialization. In the case of the binary image related to CamR's initialization frame for the car door model, illustrated in Fig. 4.8b, the MFs' patches share the same dimensions as in CamR, except for the window linked to MF₄, as indexed in Fig. 4.1a, which has a 19x19 pixels size, as imposed by the resolution of CamL's frame. The automatic patch extraction process does not require any user intervention for the

special cases in which a 21x21 pixels window cannot be extracted due to resolution constraints. Therefore, in these situations the maximum size for the feature patch is selected such that the extracted signature region does not extend outside the image. Nevertheless, the thresholding process, which will be presented shortly, is designed in a way in which it can automatically suit these special cases. The extracted patches are stored in matrix representation, in the initialization folder, along with the initial MFs selections, the initialization and background frames, and the topological structure buffers, introduced in Section 4.2.3.

The flow diagram associated to the object detection module is shown in Fig. 4.9. The process is initiated by the extraction of synchronized frames by the stereoscopic sensor. The frame extraction rate, whose selection is dictated by the general settings of the assembly line, is set to $f_{extr}^{OD} = 0.5\text{Hz}$ for the current experiments in which the speed of the sled system is $v_{ss} \approx 1.4\text{cm/s}$. Once the synchronized set of frames has been acquired, the processing is initiated in CamL's frame, as the automotive part will first appear in this view. The procedure of background differentiation is applied between the current extracted frame and the background frame stored in the initialization folder, following the same approach that was used for generating the binary images associated to the initialization frames.

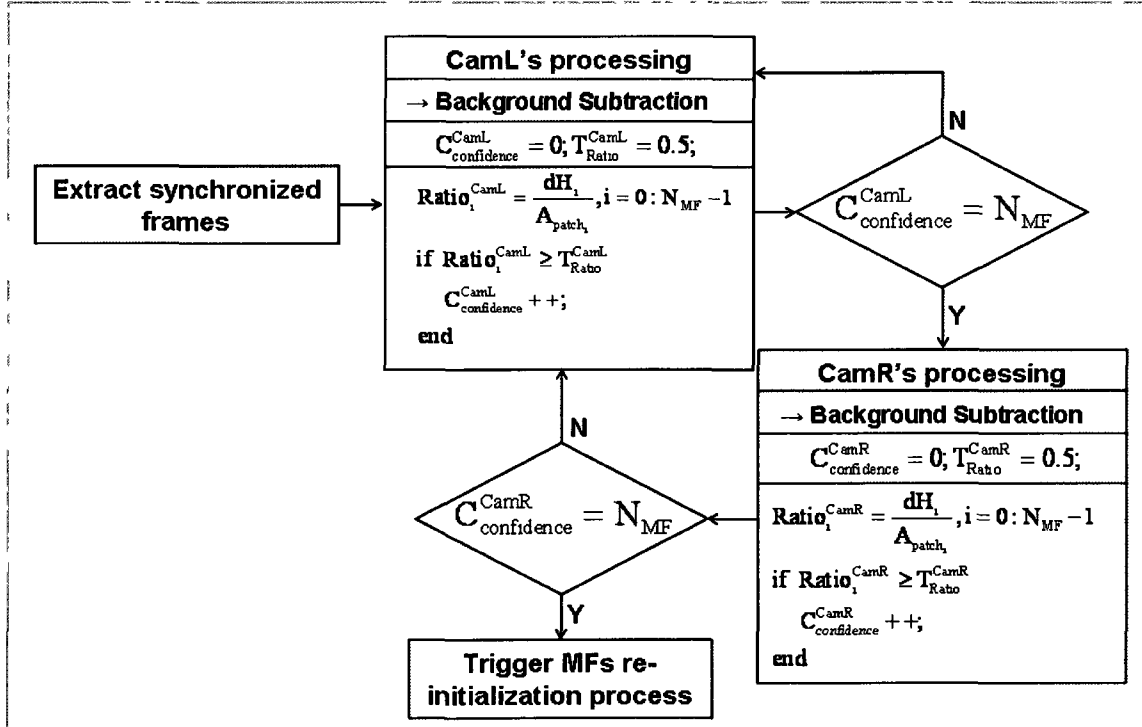


Fig. 4.9. Flow diagram of the object detection procedure.

The absolute difference between the current frame and the background frame is computed, with a threshold $T_{\text{diff}}=0.15$ imposed on the ratio between the intensity of each pixel of the image of absolute differences and the maximum intensity present in this differentiation image. The resulting binary image is used to extract the 21x21 pixels binary patches, based on the data regarding the refined locations of the MFs in the initialization frame grabbed by CamL. The assumption is made that all inspected automotive items will appear from the same direction as in the initialization phase, with possible slight variations in orientation with respect to CamR, or with small displacements along the X_o and Z_o axes, as illustrated in Fig. 4.5, from the initialization pose. Nevertheless, the proposed object detection algorithm is also completely functional in the case in which the sense of motion of the inspected part is inverted, thus the automotive part is first viewed by CamR. In addition to this, the user has the possibility of selecting the camera that is closer to the region of appearance of the rigid body. As soon as the binary patches are obtained, a “confidence” indicator $C_{\text{confidence}}^{\text{CamL}} = 0$ is assigned to CamL’s current frame. Then, the Hamming distances between the extracted binary patches and their correspondences, stored in the initialization folder, called MFs’ ground-truth patches, are computed for all the N_{MF} MFs.

As illustrated in Fig. 4.9, the “confidence” indicator is incremented each time the ratio between the Hamming distance, dH_i , associated to each $\{MF_i, i = 0, \dots, N_{MF} - 1\}$, and the area of the binary patch, A_{patch_i} (which is 441 pixels² for a 21x21 pixels patch) is greater or equal than the imposed threshold $T_{\text{Ratio}}^{\text{CamL}} = 0.5$. In the case in which, for the current extracted frame, the “confidence” indicator is smaller than the total number of MFs expected, N_{MF} , CamL’s processing block waits for the next extracted frame. Conversely, as soon as $C_{\text{confidence}}^{\text{CamL}}$ reaches the value of N_{MF} , CamR’s processing block is triggered. The latter relies on the same methodology as the one employed in CamL, as shown in Fig. 4.9. Once the “confidence” measure in CamR’s current frame, $C_{\text{confidence}}^{\text{CamR}}$ becomes equal to the total number of MFs, the full-view detection of the automotive part, by both vision systems, is considered complete, and the MFs’ re-initialization block, described in Section 4.2.2, is triggered.

In order to illustrate how the appearance of the extracted patches changes in time, while the automotive part gets in the field of view of the cameras, Fig. 4.10a shows the binary windows extracted for MF_1 , MF_7 and MF_9 , in the last five frames grabbed by

CamL, until the full detection of the automotive part, at time t_d . It should be noted that the full detection of the automotive part in CamL is not related to the moment in which the automotive part can be seen completely in CamL's view, but the moment in which the rigid body shares approximately the same location as in the initialization frames, corresponding to a full-view of the part in both stereo-views.

At time t_d , the "confidence" indicator for CamL, became equal to the number of MFs, and the processing block of CamR's frame was triggered, which also resulted in a "confidence" indicator, equal to N_{MF} , completing the object detection cycle. Figure 4.10a also illustrates the selected ground-truth (GT) MFs' binary patches extracted from the CamL's initialization frame, as well as the obtained values for $C_{confidence}^{CamL}$ in the last five processed frames. Additionally, Fig. 4.10b shows the GT MFs' binary patches, from the CamR's initialization frame, as well as the patches acquired from the only frame processed by CamR, at time t_d . In both the initialization frame and the currently processed frames, the X_o axis of the car door model, shown in Fig. 4.5, is approximately parallel to the X axis of CamR, illustrated in Fig. 3.8a. By inspecting Fig. 4.10a, it can be seen that with the advancement of the car door in the view of CamL, the extracted binary patches become more similar to their GT correspondences, resulting in a direct increase of the "confidence" indicator associated to that view. Additionally, for the three selected MFs, in Fig. 4.10b it can be noticed that the GT patches and their correspondences in the frame grabbed by CamR at t_d , also share a similar appearance.

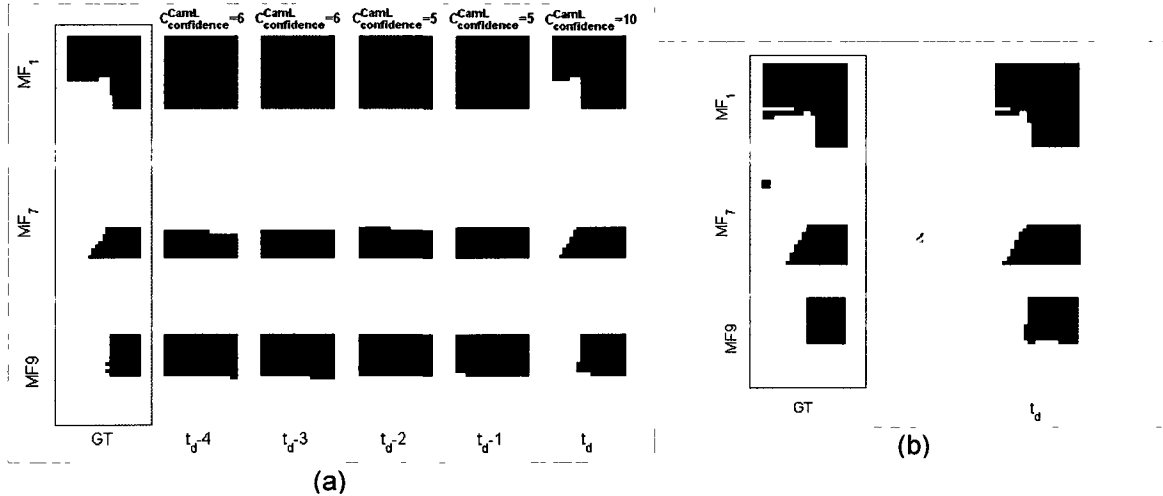


Fig. 4.10. MF₁, MF₇ and MF₉ binary patches for the ground-truth (GT) initialization frames and for (a) the last five frames extracted by CamL, until the full automotive part's detection, (b) the frame extracted by CamR at the time of detection.

In the scenario associated to the results shown in Fig. 4.10, the motion of the object happened in a plane approximately perpendicular to the principal axis of the stereoscopic sensor. However, as mentioned in Section 4.2.2, the automotive part items might appear in the view of the stereoscopic sensor with slight changes from the initialization pose. With the purpose of validating the robustness of the object detection module to these situations, a rotation of $\theta_{Y_o} = 10^\circ$ with respect to the Y_o axis, coupled with a displacement of $\Delta_{Z_o} = 10\text{cm}$ along the Z_o axis, shown in Fig. 4.5, were applied to the car door model, during the sequence processed by the object detection module. The results are illustrated in Fig 4.11.

Figure 4.11a shows the obtained ratios, $\{\text{Ratio}_i^{\text{CamL}} = \frac{dH_i}{A_{\text{patch}_i}} \mid i = 0, \dots, N_{\text{MF}} - 1\}$,

between the Hamming distance associated to each MF and the area of the binary patch for ten frames processed by CamL, from $t_0 + 2s$, until the detection time, $t_d = t_0 + 20s$, where t_0 represents the time when the object detection system was triggered. The formula for the ratios has been selected from the perspective that the maximum Hamming distance between the extracted MF's binary patches and their GT correspondences, has the same magnitude as the area of the MF's GT patches. As a result, the ratios are bounded to the interval $[0, 1]$, which simplifies the thresholding.

By inspecting Fig. 4.11a, it can be noticed that at the time of detection, t_d , all the ratios, $\text{Ratio}_i^{\text{CamL}}$ associated to the MFs' binary patches, exceed the threshold $T_{\text{Ratio}}^{\text{CamL}} = 0.5$. Similar to the effects in the previous scenario, the "confidence" indicator acquired by processing CamR's frame grabbed at t_d , was equal to N_{MF} , and the object detection process was finalized.

The appearance of the extracted binary patches, related to MF_2 , MF_3 and MF_8 , for the five CamL's frames processed within the time span $t_p = t_0 + 12s, \dots, t_d$ are displayed in Fig. 4.11b along with their associated GT binary patches loaded from the initialization folder, as well as the values for the confidence indicator, $C_{\text{confidence}}^{\text{CamL}}$, which can also be computed by checking how many MFs exceed the threshold in Fig. 4.11a. The satisfactory results illustrated in Fig. 4.11 demonstrate the functionality of the proposed object detection module for scenarios in which the automotive part items appear on the assembly line with slight variations from the pose exhibited at initialization.

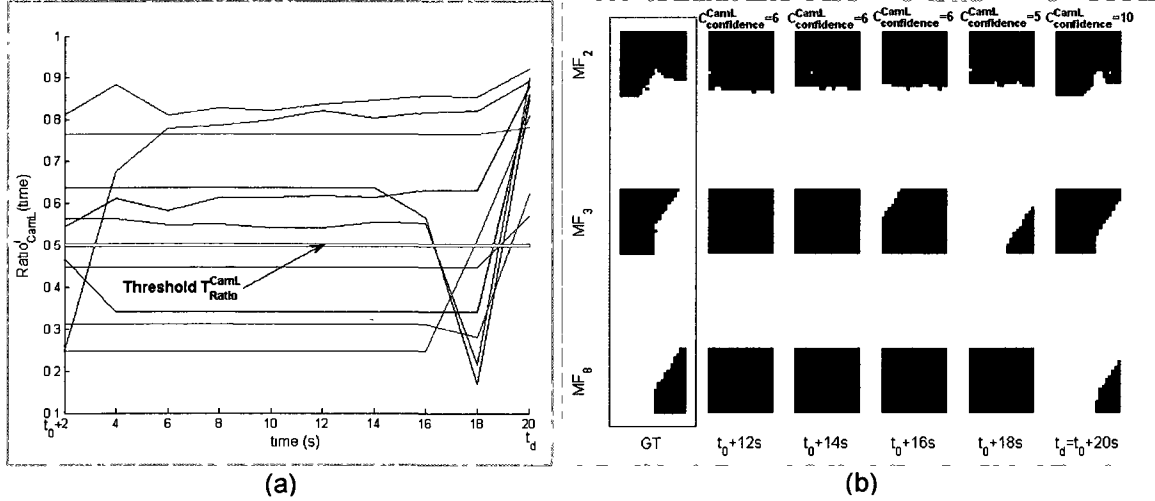


Fig. 4.11. Object detection results in CamL's frames: (a) $\{\text{Ratio}_i^{\text{CamL}} \mid i = 0, \dots, N_{\text{MF}} - 1\}$ for ten processed frames until the detection of the object at time t_d , (b) MF₂, MF₃ and MF₈'s binary patches for the GT initialization frames and for the last five frames processed until t_d .

To conclude, the principles of the proposed approach, related to the threshold selected for the ratios $\text{Ratio}_i^{\text{CamR}(\text{CamL})}$, correlated with the requirement that all the MFs' binary patches should pass the imposed threshold, provide the system with the needed robustness for the cases in which slight changes in illumination are present in the monitored scene, or the reflectance properties of the parts' surfaces vary throughout the object detection process. These effects have a noticeable impact on the appearance of the binary patches, since the resulting background subtraction image contains holes over the automotive panel's surface, as seen in Fig. 4.8. The suggested object detection solution is also robust to situations where the automotive parts appear in the view of the cameras with slight variations in orientation relative to the initialization pose.

4.3. Feature Extraction Analysis

The feature extraction process represents one of the fundamental tasks for successful operation of the system in the industrial application considered, since it has a high impact on the performance of the proposed feature-based pose and motion estimation solution. The goal of this section is to evaluate and select the most appropriate feature point detector that can be integrated in the system designed for the challenging task of estimating the pose and motion of industrial objects that suffer from a low amount of details over their surfaces. Therefore, the suitability of some of the most popular computer vision tools for keypoint extraction, which can be successful when dealing with richly textured objects, is thoroughly investigated under the constraints of

the industrial application considered in this work. Additionally, the properties of stability and robustness, which are two of the most important characteristics of a “good feature extractor” [12] will be analyzed.

Specifically, the robustness property is linked to the insensitivity to noise, and the proportion between false positives and true localized feature points. The false positive detected features include the keypoints related to noise patterns in the images or improperly extracted features [12]. The stability property [12], which also embeds the “repeatability” of feature extraction, is linked to the capability of the corner detector to identify the same points even though the images suffer from perspective distortion, zoom or illumination changes. This analysis is intrinsically testing the variability of feature detection [12] as well, since the input images contain a very low amount of details. The variability characteristic is linked to the ability of the feature extractor to still detect several feature points despite the nature of the analyzed image. The first part of this section is dedicated to the SIFT keypoint detector [9] and its suitability to the current research. Subsequently, a correlated robustness-stability measure is introduced with the purpose of selecting the most appropriate feature detector for the proposed application.

4.3.1. SIFT Keypoint Detector

The SIFT keypoint detector [9], introduced by Lowe in 2004, has received a lot of attention over the past six years [15, 86, 87], because of its capability of extracting features which are invariant to scaling and rotation. That is why the current section provides an analysis of the results obtained by using the SIFT extractor in the current research settings.

The SIFT demo program [88], implemented in MATLAB [89] was used for extracting keypoints and computing correspondences between adjacent frames in the space and time domains, as grabbed by the stereoscopic sensor in Fig. 3.8. The SIFT feature correspondence process relies on the descriptors that are attached to the extracted keypoints, which embed information about the local image gradients in the regions around the detected features, at different scales. In order to test the feature detection capability, the SIFT keypoint extractor was applied on a frame grabbed by CamL at the beginning of the motion cycle (shown in Fig. 3.12a), with the inspected scene containing just the car door mounted on the sled system. The results illustrated in Fig. 4.12a, in which each extracted keypoints has an associated vector, point out the low

variability of the SIFT detector, exemplified by the low density of features extracted on the surface of the object of interest.

As it can be noticed from Fig. 4.12a, the highest number of features is found on the sled system, whereas the majority of keypoints detected on the car door are located on the curvature region associated to the bottom frame of the door window. Additionally, from Fig. 4.12a it can also be observed that this curvature region exhibits a specular reflection effect (“highlight”) caused by the light source present in the environment.

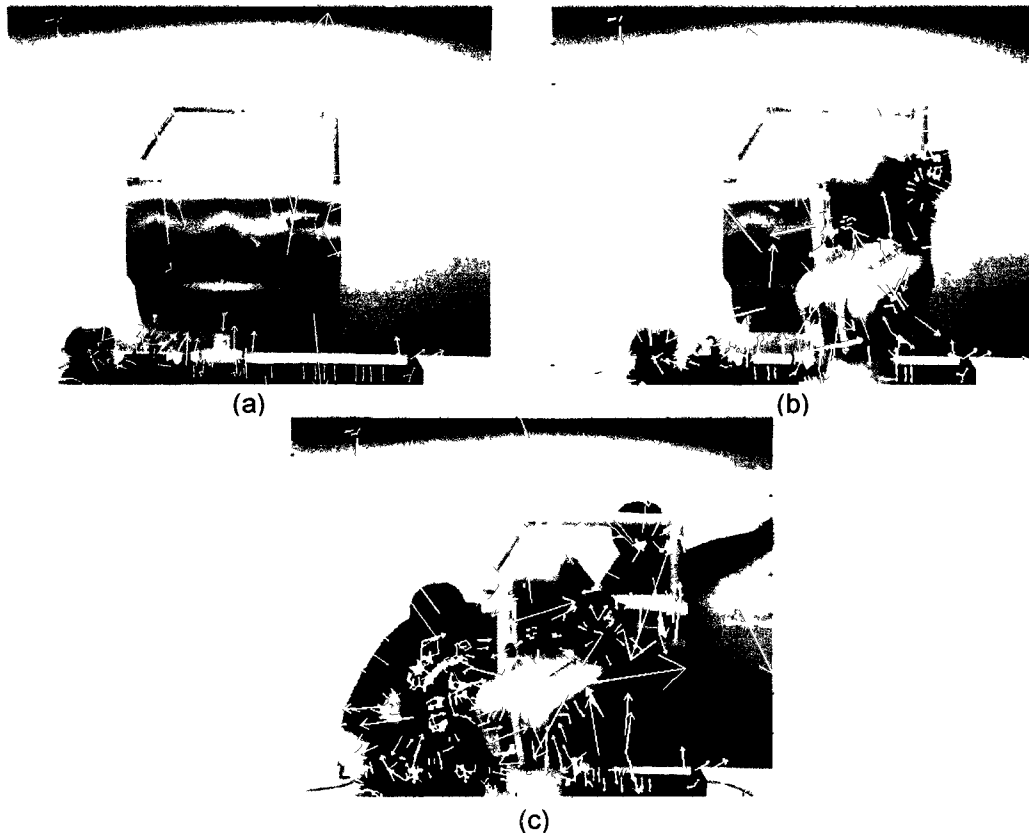


Fig 4.12. SIFT keypoints in: (a) a scene with only the car door, (b) a scene with the car door and the manipulator robot, (c) a complex scene with the car door, two human workers and the robot.

As discussed in Section 3.6.1.2, the proposed pose and motion estimation system needs to embed sufficient robustness for successful integration in the actual manufacturing environment. That is why, the constraints imposed by such an industrial setup need to be addressed in all the stages that are included in the system developed. The SIFT detector was also applied on two frames in which the complexity of the inspected scene was increased, due to the appearance of the robot and factory workers. As expected, in the two examples shown in Fig. 4.12b and Fig. 4.12c, the density of features detected with SIFT is largely affected by the amount of details present in the

inspected scene. As a result, in Fig. 4.12b the majority of extracted keypoints belong to the manipulator robot and the sled system. Moreover, Fig. 4.12c shows the increased density of features extracted in the region of the two workers.

Therefore, the SIFT feature extraction process strongly degrades the ability of the pose and motion estimator to exert insensitivity to entities in the scene other than the target industrial objects with a low amount of details on their surface. In addition, the average computational time associated to the feature detection process with SIFT was approximately $\Delta t \approx 1.5s$, which compromises the real-time processing capabilities needed for the target application.

From the matching perspective, Fig. 4.13a illustrates two segments extracted from the stereo-frames grabbed at re-initialization, together with the correspondences computed by SIFT, for the case of the car door. As it can be observed from Fig. 4.13a, only two of the ten MFs, belonging to the car door's window-frame, and marked with blue crosses, are identified as keypoints and are correctly matched. The correspondences set also includes an outlier, represented in red in Fig. 4.13a, as well as numerous features related to noise patterns in the images.

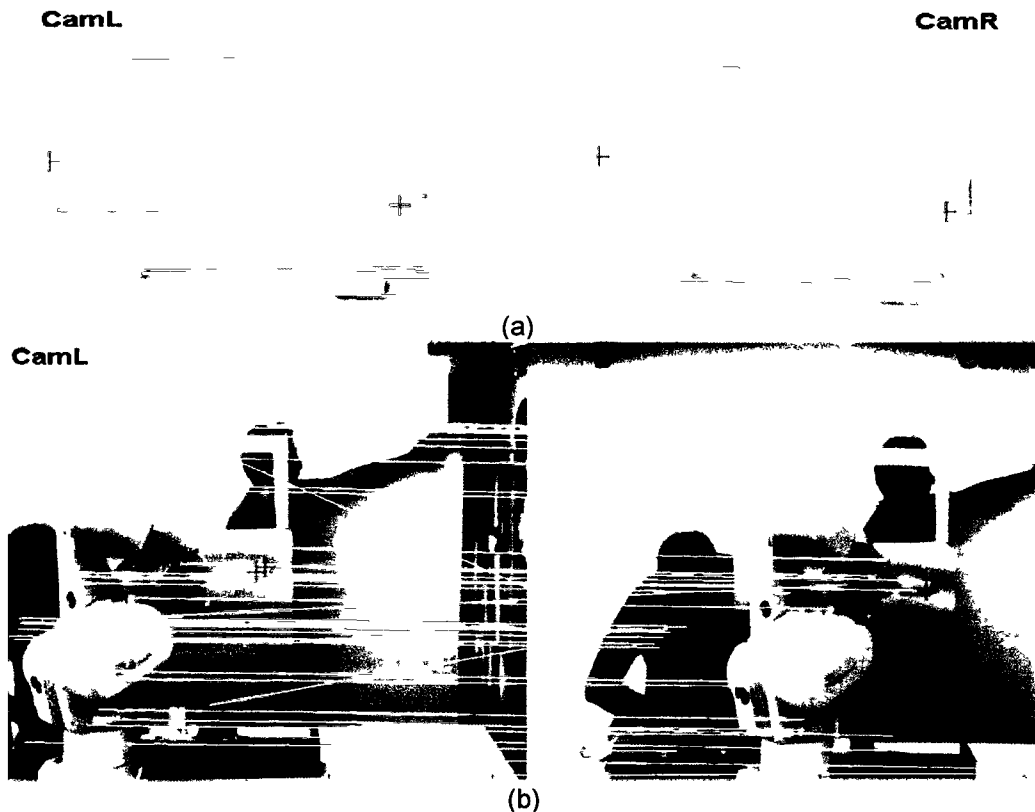


Fig. 4.13. SIFT matching results on: (a) frame segments grabbed at the beginning of the tracking sequence, (b) full-size frames with higher scene complexity.

In order to exemplify the case in which the SIFT matching procedure is applied on full-size images, Fig. 4.13b shows the correspondences computed by the SIFT detector on the more complex frame shown in Fig. 3.12b, and its correspondent stereo-image grabbed by CamR. As expected, only two correspondences, marked with blue crosses, belong to the car door, whereas all the rest provide ineffective matches for the keypoints detected on the robot and the human workers. The computational time for applying SIFT matching [9, 88] on frame segments, containing just the region of the MFs, is relatively low ($\Delta t_s^{\text{CORR}} \cong 0.4s$) as opposed to $\Delta t_f^{\text{CORR}} \cong 3.2s$ when applying SIFT matching on the full-size grabbed frames.

The suitability of SIFT matching for integration with the pose and motion estimation is preempted by two main factors. First, the SIFT detector is only sensitive to rich textured regions. Therefore applying SIFT matching between subsequent frames over time, as part of a feature tracking process in an industrial setting, would result in considerable errors for the object's pose and motion estimation. Specifically, the majority of the motion vectors would be associated to the robot, people, or other interferences appearing in the environment. Second, at the basis of the frame partitioning process, needed for the extraction of the frame segments containing the MFs' region, is the tracking methodology embedded in the pose and motion estimator prototype, as discussed in Section 4.2.1. As SIFT appears to be highly sensitive to other entities around the object of interest, accurate partitioning would be severely compromised.

Specifically, the tracking results obtained by applying SIFT matching [9, 88] on subsequent frames in the time domain were highly affected by the low "repeatability" of the feature extraction. This had a strong impact on the "lifetime" associated to the detected keypoints in the initialization frame. This phenomenon is illustrated in Fig. 4.14, which shows the keypoints detected in the segments extracted from the 1st and 21st frames grabbed by CamR during the tracking of the car door. As shown in Fig. 4.14, the vectors associated to the features detected on the car door's surface differ substantially between the two views, and not only in the area occluded by the robot. This strongly affects the feature tracking process, for which these feature vectors represent the main source of information.

Thus, the temporary appearance of different objects in the scene, which changes the density of details, strongly affects the keypoint detection process together with the matching procedure that might provide misleading information, when used with adjacent frames in time, under a tracking application. The particularities of the overall SIFT

extraction process highly affect the “lifetime” of the MFs, which are critical for the pose and motion estimation of automotive parts, as it can only rely on a limited number of features.

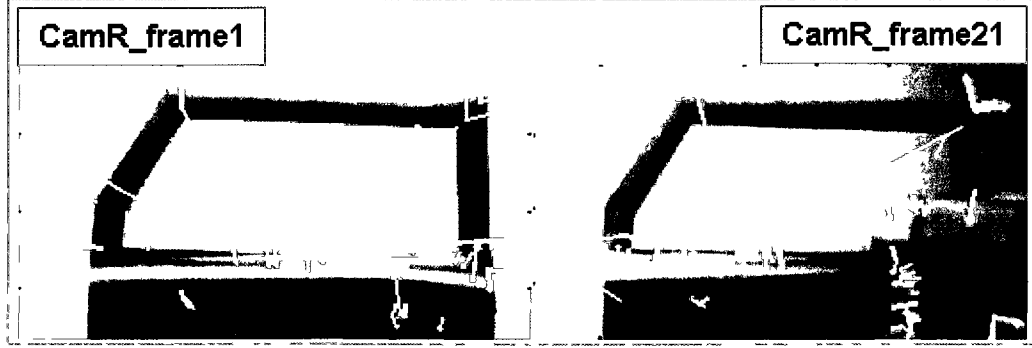


Fig 4.14. Image segments from CamR frames in the tracking sequence, with SIFT keypoints.

Based on these results, the SIFT keypoint extractor demonstrated its incompatibility for application on objects exhibiting a low level of surface texture and details. As a result, it was discarded from the set of candidate feature-point detectors.

4.3.2. A Correlated Stability-Robustness Empirical Measure

In order to empirically evaluate the correlated stability-robustness property of the feature extractors, a setup in which the sled system was positioned approximately parallel to the X axis of CamR, represented in Fig. 3.8a, was selected for the generation of experimental data. The rigid body considered was the car door model shown in Fig. 3.1c, having $N_{MF}=10$ MFs, as illustrated in Fig. 4.1a. The velocity of the sled system was set to $v_{ss} \approx 1.4\text{cm/s}$, whereas the inter-calibration, between the stereo-vision sensor and robot’s base, which will be discussed in Section 6.2.1, was used to transform the pose and motion estimation data in visual servoing information for the robotic system. Moreover, the manipulator robot was programmed to point MF_0 , as indexed in Fig. 4.1a, during the motion experiment. Under these settings, an image capture rate of $f_{extr}=0.5\text{Hz}$ was selected, which resulted in $N_{frames}=20$ frames processed throughout the total movement of the rigid body over the sled of 54 cm in length.

For assessing the combined stability-robustness performance of different feature extraction processes, a quality measure called “success percentage” is introduced:

$$P_s[\%] = \frac{1}{N_{frames}} \sum_{i=1}^{N_{frames}} \frac{(n_{MF}^{cd})_i}{(n_{MF}^{total})_i} \quad (4.1)$$

where $(n_{MF}^{cd})_i$ is the number of correctly detected MFs (in the i-th frame) within a

neighborhood of 5x5 pixels around the keypoints shown in Fig. 4.1a, and $(n_{MF}^{total})_i$ is the total number of detectable MFs in the i -th frame. It should be noticed that $(n_{MF}^{total})_i$ is not necessarily equal to the total number of MFs contained by the car door's window frame, because of the occlusions occasionally caused by the manipulator robot, also present in the scene. Without loss of generality, the “success percentage” was computed only with the results obtained by processing the frames grabbed by CamL, illustrated in Fig. 3.8a.

Other than SIFT, the feature extractors investigated in this experimentation include the Harris and Stephens corner detector [1] with the corner's strength measure given by eq. (2.2), followed by the same feature detector but with Noble's validation gate [2, 3], the SUSAN corner extractor [8], and finally, the Shi and Tomasi feature detector [6]. For the Harris and Stephens corner detector with Noble's validation gate, and SUSAN feature extractor, the MATLAB [89] framework developed by Garcia [90] was used in the analysis, whereas for the Harris and Stephens extractor [1] and the Shi and Tomasi's approach [6], their OpenCV [21] implementation was selected. All the feature extractors were applied on image segments containing the window frame ROI which was automatically extracted from all the grabbed frames, by using the tracking system, which will be described in the next section.

Due to the robust optical flow calculation and the constrained motion pattern of the car door which was a translation along the X axis of CamR, as in Fig. 3.8a, with no changes in orientation, the positions of the MFs within the ROI were not affected by the movement of the object. Specifically, a 5x5 pixels patch was manually set for all the MFs, shown in Fig. 4.1a, within the initial ROI belonging to the first frame grabbed by CamL that contains a full view of the car door, as indicated by the triggering signal provided by the object detection module. Then, the process of computing $(n_{MF}^{cd})_1$ consisted in verifying if the detected MFs lie within the pre-assigned patches, whose positions remained the same, with respect to the ROI, throughout the entire duration of the experiment. Figure 4.15 illustrates the feature extraction results obtained with the four selected approaches applied on the 10th frame segment during the motion of the rigid body. In this case, $(n_{MF}^{total})_{10} = 8$ is the total number of detectable MFs, which were all correctly detected by Shi and Tomasi's corner extractor [6] and Harris's detector [1] with Noble's validation gate [2, 3].

In Fig. 4.15, the correctly detected corners are marked with the symbol “●”, whereas for the erroneously detected features the symbol “○” is used. In order to

automatically compute $(n_{MF}^{total})_1$, a 20x20 pixels patch was centered on the MFs in the ROI extracted from the first frame grabbed by CamL.

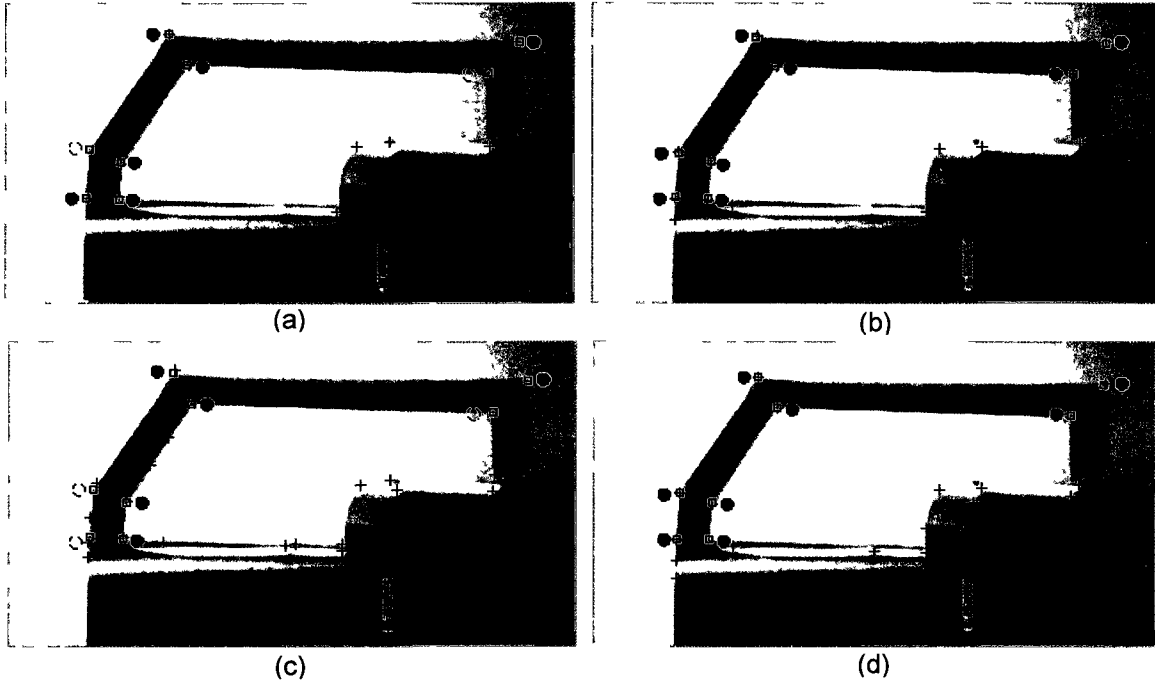


Fig 4 15 Feature extraction results in MFs 5x5 pixels patches (a) Harris and Stephens corner detector, (b) Harris and Stephens corner detector, with Noble's corner strength measure, (c) SUSAN corner detector, (d) Shi and Tomasi corner extractor

A similarity measure quantizing how much the intensities of the selected region have changed during the tracking, when compared to the distribution of intensities in the first grayscale frame, was used to estimate the total number of detectable MFs. More exactly, the similarity measure consisted of the sum of squared differences (SSD) between the corresponding intensity values in the MFs' patches. By taking into account the characteristics of the experimental setup for the current test, the SSD similarity measure was sensitive enough to identify MF regions that were occluded by the manipulator robot.

The results obtained for the "success percentage" with each selected feature point extractor are summarized in Table 4.2. In the case of the Harris and Stephens corner detector [1], a threshold $T_{Harris_Stephens}=0.01$ was used for the corner strength measure. As it can be noticed from Table 4.2, a considerable improvement in the success percentage is obtained by applying Noble's validation gate [2, 3] with the same threshold, $T_{Noble}=0.01$. As mentioned in Section 2.2.1, this improvement was expected, since Noble's validation gate [2, 3] diminishes the sensitivity of the Harris and Stephens

corner extractor [1] to image patches having contrast variations. Under these settings, Noble's validation gate is more robust since it reduces the amount of difficulty associated with the process of setting up the threshold for the Harris and Stephens corner detector.

Table 4.2. "Success Percentage", $P_s(\%)$, for the four feature extractors.

Feature Extractor	Success Percentage $P_s(\%)$
Harris and Stephens corner detector	57.14%
Harris and Stephens corner detector with Noble's validation gate	75.25%
SUSAN corner detector	47.52%
Shi and Tomasi corner detector	77.08%

The SUSAN corner detector [8] was used with its default thresholds [90] and gave the lowest success percentage from all of the studied feature extractors. The Shi and Tomasi corner detector [6], used with the threshold $T_{\text{Shi_Tomasi}}=0.02$ gave the highest success percentage for the proposed stability-robustness correlated measure.

Thus, the feature extraction block, shown in Fig. 3.13, builds upon the Shi and Tomasi corner detector [6, 21], which is selectively applied on the ROIs containing the area of the MFs, as discussed in Section 4.2.1. Moreover, the supervisory layer, shown in Fig. 3.13, is responsible for monitoring the ROI extraction process, to ensure successful selective feature extraction.

4.4. Feature Tracking Analysis

After ensuring that features are extracted with stability and reliability, a second essential component of the proposed methodology for pose and motion estimation is the feature tracking process. Here the goal is to assign a motion vector, or optical flow, for each extracted feature, which quantifies the displacement exhibited by the rigid body between two successive frames.

This section proposes an analysis of the feature tracking approach and is divided in two parts. First, the pyramidal implementation of the LK tracker [20, 22] is introduced, along with some preliminary optical flow results. Second, the constraints imposed by the considered application, introduced in Section 3.6, guide the analysis of the limitations related to the pyramidal LK tracker, which will be resolved by the introduction of a supervisory sub-system to be detailed in Section 5.4.

4.4.1. Preliminary Optical Flow Results

The feature tracker introduced by Lucas and Kanade [19, 20] received a lot of attention in the last two decades. Moreover, its pyramidal implementation developed by Bouguet [22] has solved the limitation of the iterative LK tracker [20], to identify large motions, by properly balancing the trade-off between local accuracy and robustness. As a result, the pyramidal LK tracker [20, 22] was selected for integration with the proposed pose and motion estimation solution. It is the central component of the feature tracking block, illustrated in Fig. 3.13.

Since accurate and robust pose and motion estimation is fundamental for robotic interaction with a moving object, the validation of the optical flow data is imperative. The proposed outlier removal gate [72], at the level of the feature tracking block that is shown in the high-level diagram of the pose and motion estimation solution (Fig. 3.13), consisted of imposing a threshold of $T_{\text{residualLK}}=350$ for the tracking error returned by the OpenCV's implementation [21] of the pyramidal LK tracker [20, 22]. This tracking residual is given by the root mean square (RMS) error of the pixel intensity values between the corresponding feature patches, and is defined as:

$$\text{Tracking_Error}(x, y) = \sqrt{\sum_{k \in W} [I_t(x + k) - I_{t+1}(y + k)]^2} \quad (4.2)$$

where I_t, I_{t+1} are two subsequent images in time, (x, y) are the corresponding features between the two frames, and W represents the image patch centered on (x, y) .

Figure 4.16 illustrates the motion vectors, computed for the two industrial objects, by applying the pyramidal LK tracker with the suggested outlier removal gate. By analyzing Fig. 4.16, it can be noticed that all the optical flow vectors point to their proper MFs, and they share a similar magnitude, which is an expected behavior, in the case of rigid bodies' motion. Apart from providing satisfactory feature tracking results, the validation gate was also integrated in the first version of the proposed pose and motion estimation system [72], in which the stereoscopic sensor shown in Fig. 3.7, and the rigid body structure, illustrated in Fig. 3.1b, were used in the experimentation. Nevertheless, in this first prototype, the view of the inspected object was never occluded by the manipulator robot or people present in the scene.

The next section discusses the limitations of this validation gate when the system is transposed in a more realistic industrial environment and highlights the necessity for a more advanced monitoring architecture, which will be embedded in the supervisory layer that will be detailed in Section 5.4.

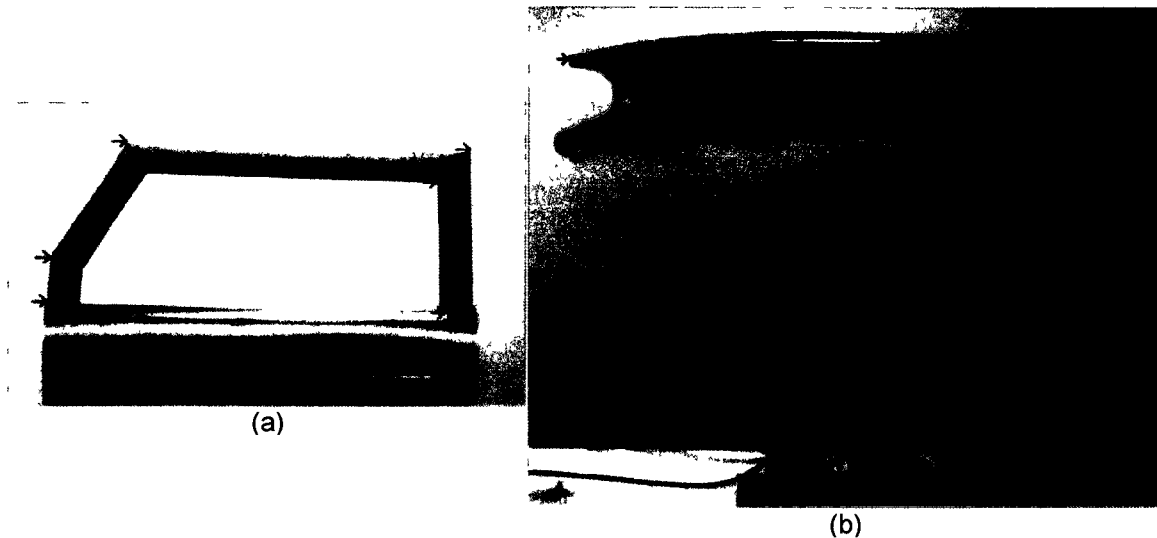


Fig. 4.16. Preliminary optical flow results computed with the pyramidal LK tracker for: (a) car door, (b) fender.

4.4.2. Limitations of the Pyramidal LK Tracker

As mentioned in Section 2.2.2, the sources of error associated to the tracking methodology developed by Lucas and Kanade [20] have been divided by Weber and Malik [24] into four different classes. The first class, called “stochastic” is caused by sensor noise. The second category is represented by the errors due to the failure of the brightness constancy assumption, which happens in the cases of occlusions or photometric variations. The pyramidal implementation [22] of the LK tracker solves the third class of errors, labeled as “systematic”, which are related to the situations in which a large displacement is present in-between the processed frames. Finally, the last category is linked to the cumulative errors in the motion prediction process, which produce a drift of the features throughout the tracking procedure. The requirement for solving the remaining three classes of errors motivated the transition to robust tracking, for which different solutions have been proposed in the literature [17, 18, 20, 23-27, 35, 36].

The second class of errors related to the LK tracker [20] is caused by the increase in the complexity level of the monitored scene, throughout the tracking sequence. In the current research, this represents a very relevant and representative problem associated with industrial manufacturing installations. In such environments, quality control stations and assembly line do not operate alone, but are often surrounded with robots and workers moving continuously. In order to exemplify this second class of errors, Fig. 4.17 illustrates the motion vectors computed with the pyramidal LK tracker

[20, 22] during the tracking process for cases in which the inspected scene also contains robot or human appearances.

The results shown in Fig. 4.17 originate from the feature tracking block, indicated in Fig. 3.13, along with the outlier removal gate discussed in Section 4.4.1, and with no intervention from the supervisory layer. It can be noticed that all the motion vectors (represented in red), which have passed the validation gate (in Fig. 4.17a), are outliers and none of them point to the correct MF, as indexed in Fig. 4.1a. The use of such erroneous data with the robotic interaction thread should be definitely avoided.



Fig. 4.17. Pyramidal LK tracking results, in the cases in which the complexity of the monitored scene is increased by the: (a) appearance of the manipulator robot, (b, c) temporary appearance of people in the scene.

In the second case, illustrated in Fig. 4.17b, only one motion vector, that corresponds to MF_2 and is surrounded by a light-blue ellipse, points to the correct MF position, in spite of the fact that the person present in the scene is not occluding any of the valid MFs. Similarly, in the third case, shown in Fig. 4.17c and related to the fender

model, only four motion vectors (related to MF_0 , MF_1 , MF_2 and MF_3 , as indexed in Fig. 4.1b), surrounded by ellipses with discontinuous-line borders, indicate their correct positions, whereas the other three motion vectors are affected by severe tracking errors.

As it can be observed from Fig. 4.17, the temporary appearance of other entities in the scene strongly degrades the robustness of the pyramidal LK tracker, leading to critical tracking errors. By comparing with Fig. 4.16a, in which the motion vectors were validated with the same procedure, the results shown in Fig. 4.17 demonstrate important limitations of the pyramidal LK tracker and strongly support the need for robust monitoring. Another interesting effect, which can be noticed from Fig. 4.17, is related to the fact that the erroneous motion vectors are not only obtained in the occluded region or the image segments in which the local intensities have changed due to the appearance of other entities. They occur in the rest of the image regions as well. The latter is definitely an effect of the pyramidal implementation [22] of the iterative LK tracker, which builds upon changes in image resolution in order to identify big motions. Moreover, the changes in illumination or in the reflectance properties of the object's surface affect the pyramidal LK tracker in the same manner as the occlusions.

Finally, the last class of errors [24] affecting the pyramidal LK tracker [20, 22] is linked to the drift accumulating with the tracking. Thus, Fig. 4.18 exemplifies this phenomenon with two segments extracted from the first and eighteenth processed frames throughout the tracking, in an experiment without the presence of the robotic arm.

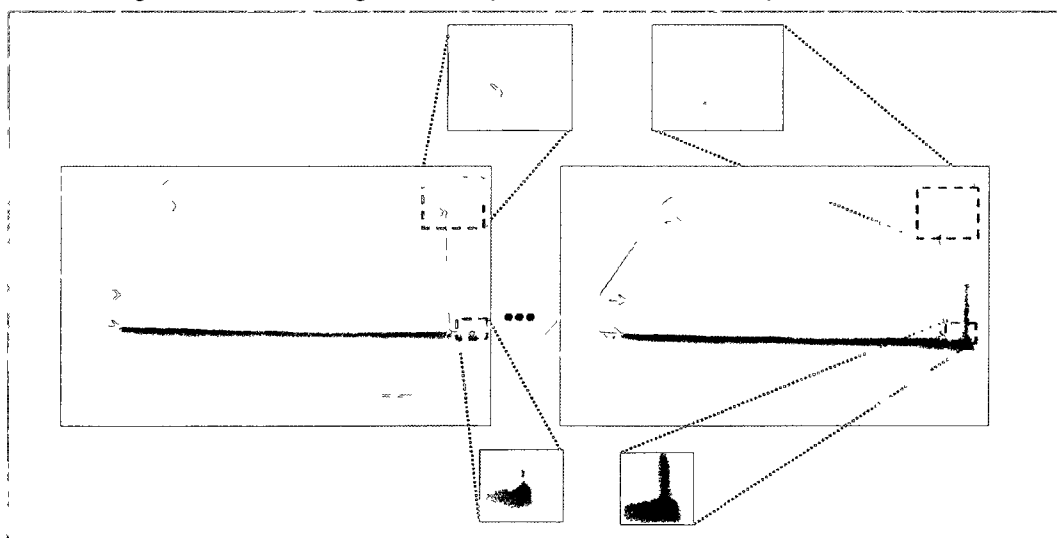


Fig. 4.18. 1st and 18th frame segments with highlighted MFs affected by the drift accumulating along the tracking by using the pyramidal LK tracker.

As noticed in Fig. 4.18, the drift cumulating along the tracking noticeably affects the MFs belonging to the right side of the inner and outer frame of the door window. In

the zoom that can be seen in the upper part of Fig. 4.18, the motion vectors (in red) associated to the 1st frame point to their correct MFs, specifically MF_1 and MF_6 , as indexed in Fig. 4.1a. Subsequently, the motion vectors belonging to the same region of the car door in the 18th frame, do not point anymore to their associated MFs, especially the vector related to MF_1 . Moreover, a drift similar to the one exhibited by the motion vectors coupled to MF_6 , can be noticed in the zoomed regions around MF_0 , shown in the lower part of the figure. Additionally, since the velocity control module of the sled system has a response similar to a ramp function (until the imposed reference speed is reached) and due to the uniform frame extraction process associated to this experiment, the norm of the motion vectors in Fig. 4.18 is different in the two processed frames. However, regardless of the norm, they should all point to the correct location of the car door's MFs.

Due to the accumulated drift, the 2D topological structure of the industrial object, which is defined by the MFs in the image plane, changes throughout the trajectory of the object. As a result, the error in the pose and motion estimation of the industrial part increases, causing inaccuracies in the visual servoing data which is vital for the robotic guidance.

The drift accumulating along the tracking was also analyzed by Shi and Tomasi [6] by means of the intensity dissimilarity measure between the feature patches during tracking. The authors concluded that the inter-frame feature correspondence process is inadequate for applications that require the matching, for a finite size image patch, over a considerable time span. More specifically, the feature tracking process is subject to cumulative errors in the cases in which the trajectories are integrated over time [36].

A simplistic way of solving this issue would be to perform feature extraction on every frame, such that the feature tracker is continuously changing the feature set, resulting in the minimization of the time spans for the trajectory integration. Although this approach might be useful for the ego-motion estimation of a mobile robot navigating in static scenes with rich textures, its applicability to the task of robotic interaction with moving parts would result in poor performances. The main reason for this conclusion comes from the requirement of the pose and motion estimator to exhibit insensitivity to the presence of other external entities that are typically present and moving in the scene, such as the robot and the workers. Therefore, by applying this "pair-wise" approach between all successive frames would results in tracking multiple entities at the same time, especially the ones that exhibit a higher density of features.

To conclude, the phenomena associated to the classes of errors affecting the pyramidal LK tracker are expected to be apparent under the typical factory environment of the automotive industry, especially those related to the increase in the complexity of the monitored scene and the drift accumulating along the tracking [24]. Therefore the need for validating the optical flow information is imperative before using it to guide the robot. Moreover, the practical constraints of the application, defined in Section 3.6, justify the need for extending the lifetime of the MFs over the entire motion of the object. This can be achieved by supervising the feature tracking and matching processes as will be proposed in the following chapter. In this way, the effects of the two dominant types of errors can be reliably alleviated. Only in this way, the feature-based pose and motion estimation system is able to exhibit the required accuracy for guiding the manipulator robot during the surface deformation defects marking procedure.

4.5. Feature Matching Analysis

The sparse 3D reconstruction of the automotive part, which will be discussed in Section 5.2, is motivated by two important factors. On the one hand, the 3D recovery of the MFs, together with the inter-calibration between the acquisition system and the robot's base, which will be discussed in Section 6.2.1, make possible the integration of the pose and motion estimation data within a visual servoing "look-and-move" architecture [63]. In this way, the robot can be guided to point one of the MFs and the robustness of the feature extraction, tracking and matching processes can be reliably tested. Therefore, the sparse 3D reconstruction of the object is a very important prerequisite of the robotic integration process, which permits robust testing, even in the cases in which the 3D surface defects detection module is not already integrated in the final prototype for robotic marking of deformations.

On the other hand, as it will be explained in Section 6.2.2, the inter-calibration between the stereoscopic sensor used for pose and motion estimation (SSPME) and the 3D surface modeler, CamL_{SL}, makes possible the transfer of the deformation defects' locations into CamR's reference frame, at the beginning of the tracking process, which is triggered by the object detection module. However, in order to plan the on-line marking operation, the 3D location of the deformations, with respect to CamR, has to be updated during the tracking, according to the motion estimations. Nevertheless, at the basis of the motion estimation process are situated the 3D MFs' point-clouds computed for the previous and current frames, as it will be discussed in Section 5.3.

The sparse 3D reconstruction of the inspected scene relies on the feature matching procedure which is responsible for finding corresponding points between different views of the given scene. As mentioned in Section 2.2.3, the two prerequisites for feature matching are related to the visibility of the keypoints in all inspected views, followed by the assumption that the difference in appearance between the corresponding image patches should be fully characterized by a restricted space transformation.

This section, which is dedicated to the feature matching block, shown in Fig. 3.13, provides an insight into the selected approaches for the feature correspondence problem and illustrates the challenges identified throughout the experimentation process. Specifically, the first part of this section is related to the popular “Torr-Tool” for structure and motion estimation, whose central component is represented by the robust matching process. As a result, the suitability of “Torr-Tool” for the feature matching component of the current application is tested. Secondly, the particularities of the feature correspondence block are discussed, along with some preliminary results and the encountered limitations.

4.5.1. “Torr-Tool” for Feature Matching

Torr [49] has proposed a “structure and motion toolkit” developed in Matlab [89] in order to provide working examples for all the components included in the pose and motion recovery process. Therefore, it was considered interesting to validate the suitability of the “Torr-Tool” [49] framework for the current application, which extends the usual bounds of computer vision (i.e. assuming objects with rich textures) due to the appearance of the tracked objects, which exhibit a very low density of apparent details on their surface.

From the feature extraction perspective, Torr’s toolkit makes use of Harris’s corner detector [1]. For the experimental validation, a pair of ROIs containing the MFs of the car door model were extracted from the synchronized frames grabbed by the stereoscopic sensor, at the beginning of the tracking process, in which the robot was not present in the scene. The Harris corner detector [1] was applied on both ROIs, with the parameters, $(N_{\text{corners}}, \sigma_{\text{smooth}}, w) = (50, 1, 6 \text{ pixels})$, where N_{corners} is the maximum number of requested corners, and $(\sigma_{\text{smooth}}, w)$ represent the standard deviation and width of the Gaussian filter used to smooth the images before applying the corner extraction. The results provided by Harris’s corner detector [1] on the two ROIs are shown in Fig. 4.19.

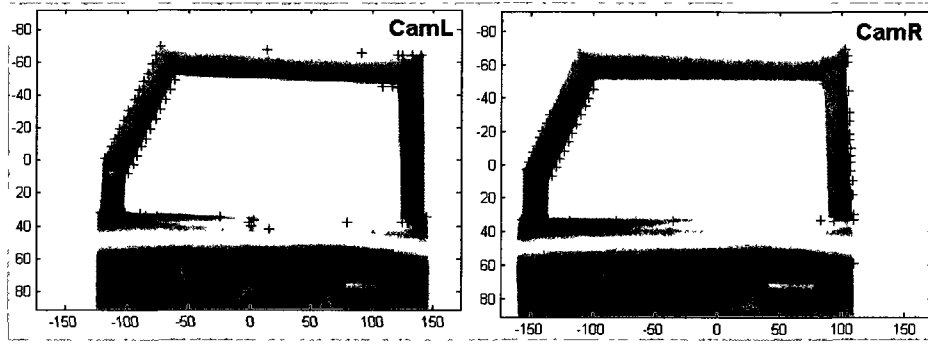


Fig. 4.19. Harris corners detected with Torr's toolkit.

Since the implementation of the Harris corner detector [1] did not include any support for imposing a minimum distance between the returned features, from Fig. 4.19 it can be noticed the high density of corners extracted on the left (CamL and CamR) and right sides of the car door's window frame (CamR). The first technique [49] for the feature correspondence process uses a correlation matching approach which builds upon the sum of squared differences (SSD) between the intensities of the feature patches in the two processed images. The author [49] acknowledges the limitations of this proposed method for the cases in which the stereo-frames exhibit significant changes in perspective and lighting.

For the current validation, the correlation matching procedure was applied with the parameters $(\Delta_{max}, w_{corr}) = (30, 7)$ pixels, where Δ_{max} is the size of the search window, and w_{corr} represents the size of the correlation window. The computed disparity vectors are illustrated in the ROI extracted from the frame acquired by CamL in Fig. 4.20a. Additionally, the corners detected in CamR's ROI are also represented in Fig. 4.20a in order to provide an insight into the norm of the disparity vectors. From Fig. 4.20a it can be noticed that the matches contain a high number of outliers since multiple corners from CamR's ROI point to the same feature in CamL's ROI, generating ambiguity for the correspondence process.

With the purpose of estimating the fundamental matrix between the two views, Torr [49] also proposed a robust methodology that builds upon MAPSAC ("Maximum A Posteriori Sampling Consensus") for performing maximum likelihood estimation in the presence of outliers. For the experimentation, the parameters $(in_{max}, T_{in}) = (10, 5)$ have been used, where in_{max} represents the maximum number of samples to be drawn and T_{in} is the threshold for inliers. Although this technique is able to remove a part of the outliers present in Fig. 4.20a, the final set of correspondences is still erroneous, as shown in Fig. 4.20b, leading to a low accuracy estimation of the fundamental matrix. Additionally,

when compared to Fig. 4.20a, in Fig. 4.20b are illustrated the disparity vectors in both views, as performed by “Torr-Tool” with the purpose of observing a similar orientation for all these vectors. Unfortunately, in the case of the inspected industrial object with few distinctive features, this effect cannot be reliably observed, as the set of matches still contain a considerable number of outliers.

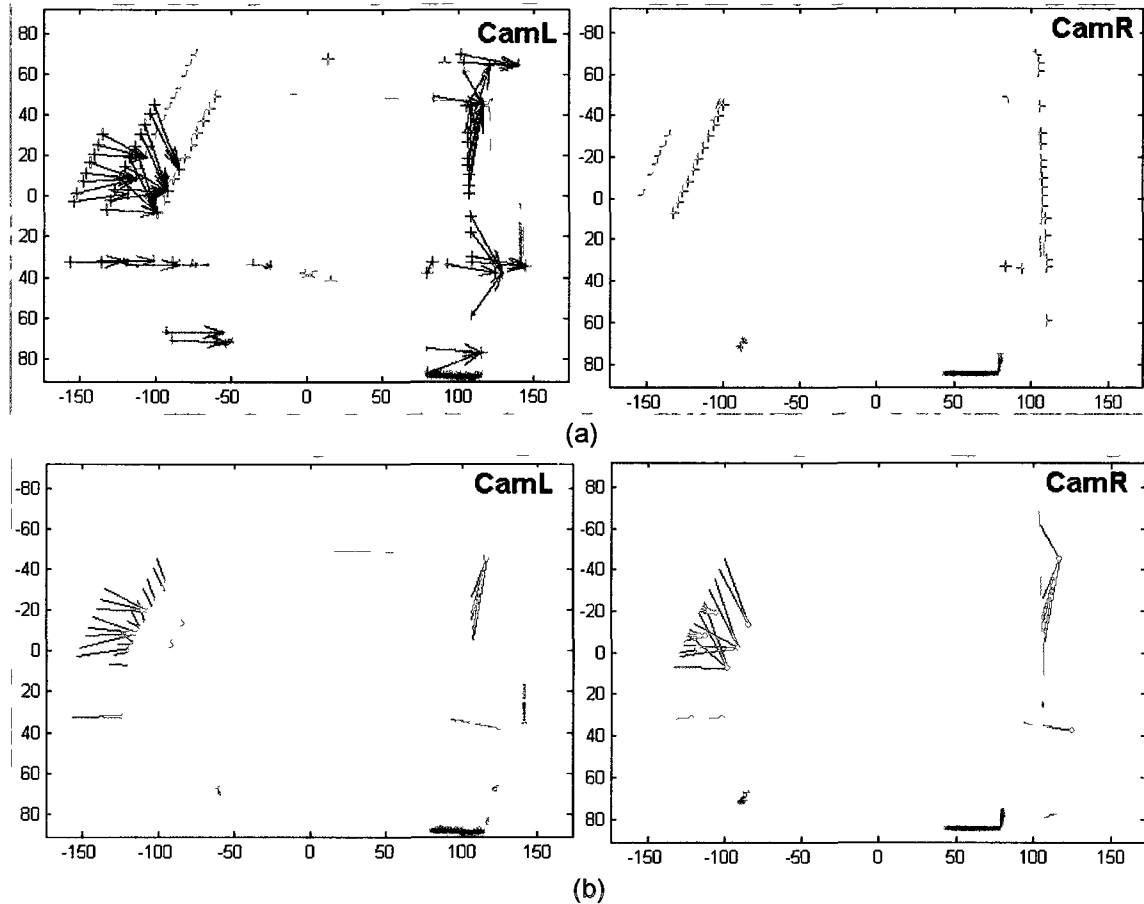


Fig 4 20 Feature matches computed with Torr's toolkit using (a) correlation-based approach, (b) MAPSAC-based approach

To conclude, the suitability of Torr's toolkit on the proposed application is highly affected by the general appearance of the industrial bodies which exhibit a small number of unique, sharp features over their surface. Additionally, the feature correspondence results highlighted in Fig. 4.20 support the decision of fully calibrating the stereo-vision system, since the appearance of the rigid objects is not a good support for the methodologies performing fundamental matrix estimation based on feature matches [31, 39, 46-49]. Therefore, the classical techniques for feature matching [31, 38-40, 41, 46-49] which provide satisfactory results on richly textured objects require a substantial amount of adaptation, when transferred to industrial bodies having fewer features.

4.5.2. Feature Matching Sub-system

The basis of the feature matching approach was introduced in Section 4.2.1 where the problem of feature correspondence between the pre-selected MFs over the structure of the automotive panels was addressed. As a result, the particularities of the final stereo-vision configuration for pose and motion estimation, in combination with the approximately parallel configuration of the cameras, supported the use of the pyramidal implementation of the LK tracker [20, 22] in guiding the stereo-matches between the two inspected views. A similar methodology, relying on the optical flow computation platform developed by Bulthoff *et al.* [91], was proposed by Mulligan [50] with the goal of providing guidance to the feature matching process.

The feature matching block, included in the high-level diagram of the pose and motion estimation solution (Fig. 3.13), builds upon the pyramidal LK tracker [20, 22] and the Shi and Tomasi corner detector [6], as discussed in Section 4.2.1. Based on the index associated to the MFs, as shown in Fig. 4.1, and their preserved order during the tracking, the actual MFs' matches, in each processed frames, are known upon the computation of the motion vectors in each view.

In the early stages of the pose and motion estimation solution [72], based on the selected image capture rate f_{extr} , the correspondences were refined with different frequencies $f_{refine} = \alpha \cdot f_{extr}$, where $\alpha \in [0, 0.5]$. The refinement technique built upon the Shi and Tomasi corner detector [6], as described in Section 4.2.1. However, under the current setup in which the robot is also present in the scene, as well as human workers, this refinement technique has been discarded, in order to avoid the correction of the MFs with features extracted over the surface of different entities, except the industrial body. Nevertheless, the supervised perspective over the feature tracking process, that will be described in Section 5.4, provided satisfactory feature matching results, with no need of re-applying the Shi and Tomasi corner extraction.

Since the proposed matching technique relies on the pyramidal implementation of the LK tracker [20, 22] and Shi and Tomasi's corner detector [6], the challenges described in Section 4.4.2 have a considerable impact on the feature correspondences as well. The drift accumulating along the tracking, the occlusions, the temporary appearance of external entities in the scene, as well as the photometric variations, affect the precision of the feature correspondences, introducing errors in the sparse 3D structure of the rigid body, *a posteriori* computed.

As a result, the inaccuracies of the feature tracking and extraction processes have a mixed effect on the stereo correspondence error, which, according to Olson [35] has the strongest impact on the estimation of the 3D position of the matched features. That is why, the feature monitoring and validation phases, embedded in the supervisory layer, which will be described in Section 5.4, will also lead to improvement of the feature matching precision.

The full calibration of the SSPME makes possible the process of verifying the accuracy of the computed feature correspondences. Under the stereo-vision settings, the projections of an unknown point X from the 3D space, in the two image planes, π_R and π_L are related by the epipolar constraint. Therefore, the projection x , on one of the image planes, of the unknown 3D point X , can provide information about the location of its match, y , in the second image. Specifically, the matching point in the adjacent image in space must lie on the epipolar line, which can be computed with the fundamental matrix of the image pair. As a result, the fundamental matrix, which characterizes the relationship between points in one view, and their associated epipolar lines in the other view, is defined as:

$$F = (M_{intL}^{-1})^T \cdot E \cdot (M_{intR}^{-1}) \quad (4.3)$$

where M_{intL} , M_{intR} are the intrinsic matrices of CamL and CamR, and E is the essential matrix, computed with the extrinsic calibration data. The 3x3 fundamental matrix has rank 2 and 7 degrees of freedom (DOF). According to the stereoscopic sensor illustrated in Fig. 3.8, in which CamR is the associated world reference frame, it can be written:

$$x_{CamL}^T \cdot F = l_{CamR} \quad (4.4)$$

in which x_{CamL} is the projection of the 3D point X on CamL's image plane, and l_{CamR} is the epipolar line on which the projection of X in CamR's image plane should be situated. By imposing this epipolar constraint, eq. (4.4) can be rewritten as:

$$x_{CamL}^T \cdot F \cdot x_{CamR} = 0 \quad (4.5)$$

where x_{CamR} is the projection of X on CamR's image plane. Additionally in eq. (4.5), x_{CamL} , x_{CamR} are expressed in homogenous coordinates.

By taking into account the eq. (4.3), a closed-form solution for the fundamental matrix can be computed upon the full calibration of the stereo-vision sensor. As a result, the accuracy of the feature correspondences can be directly assessed, with no need for applying robust techniques for the estimation of the fundamental matrix, based on feature matches between the stereo-views. Nevertheless, the results discussed in

Section 4.5.1 provided important insight into the limited capabilities of the current application to be successfully integrated within a robust framework for fundamental matrix estimation.

Based on the fundamental matrix, calculated with the calibration data, two different measures are proposed for validating the accuracy of the computed feature correspondences. The first suggested measure, called “epipolar residual” is related to the exactitude with which the matches validate the epipolar constraint. Therefore, the epipolar residual, computed with the formula located in the left side of eq. (4.5), which should be ideally zero, is continuously monitored for all the computed correspondences, throughout the motion sequence of the automotive part.

Under the settings of the pose and motion estimation for the car door panel or fender, once the location of the MFs has been determined in the frames grabbed by CamL, their correspondent epipolar lines in CamR’s frames are computed in OpenCV [21]. Figures 4.21a and 4.21b show two segments from the 1st and 6th frame, grabbed by CamR during the motion experiment, in which the epipolar lines corresponding to the MFs, detected in CamL’s frame, have been represented. Nevertheless, in Fig. 4.21c are illustrated the epipolar lines in CamR’s frame, associated to the MFs detected in the 4th frame processed by CamL during the tracking sequence of the fender. In all the cases represented in Fig. 4.21 it can be noticed that the detected MFs in CamR’s frame are situated on their corresponding epipolar lines, as imposed by eq. (4.5).

The second measure associated to the accuracy of the feature matches is given by the distance of the MFs, in CamR’s frames, to their associated epipolar line. For a specific $MF_i^{CamR}(x_i, y_i)$, its correlated epipolar line l_i^{CamR} , computed in the frame I_{CamR} , can be expressed as:

$$l_i^{CamR}(a, b, c) = \{(x, y) \in I_{CamR} \mid ax + by + c = 0\} \quad (4.6)$$

in which (a,b,c) are the parameters of the epipolar line. Then, the distance from $MF_i^{CamR}(x_i, y_i)$ to its associated epipolar line l_i^{CamR} can be calculated using the formula:

$$d(MF_i^{CamR}, l_i^{CamR}) = \frac{a \cdot x_i + b \cdot y_i + c}{\sqrt{a^2 + b^2}} \quad (4.7)$$

Therefore, these two proposed measures are monitored throughout the entire motion experiment, since they are strongly linked to the accuracy of the feature matching procedure. As a result, the two measures are integrated in the motion estimation subsystem with the purpose of discarding the 3D points that have been reconstructed from

inaccurate correspondences in the image planes. By performing this test, the precision of the motion estimation can be improved, although it relies on a limited number of MFs. Nevertheless, it should be mentioned that these measures strongly depend on the precision of the intrinsic and extrinsic calibration data.

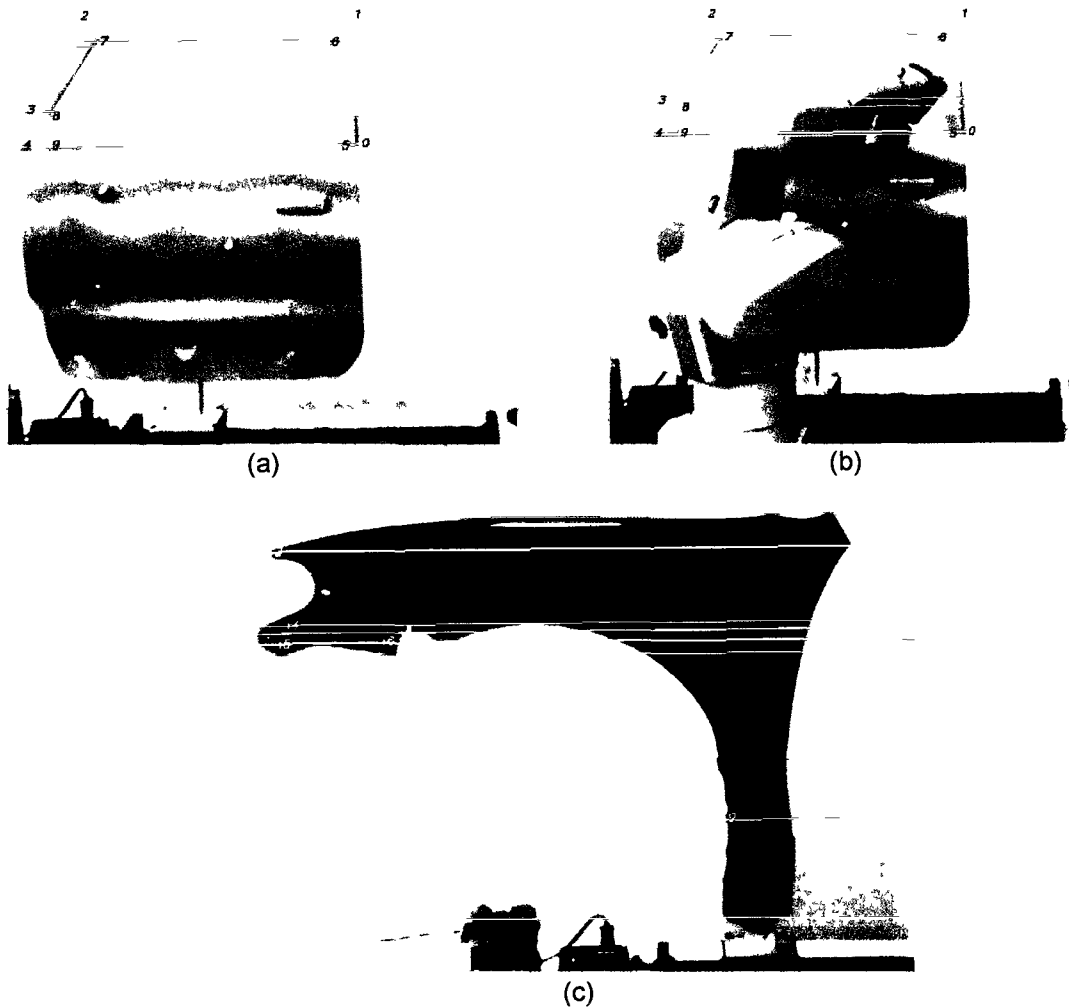


Fig. 4.21. Epipolar lines associated to the MFs in three segments extracted from the (a) 1st and (b) 6th frame grabbed by CamR during the motion experiment of the car door. (c) 4th frame grabbed by CamR during the tracking of the fender

4.7. Chapter Summary

This chapter presented an extensive evaluation of the feature extraction, tracking and matching processes, which are the fundamental components of the proposed pose and motion estimation solution. Additionally, the MFs' selection and object detection procedures are discussed in the first part of this chapter.

One of the fundamental merits of the proposed pose and motion estimation solution originates from the manner in which the particularities of the industrial quality control application are combined with popular feature extraction, tracking and matching techniques. On one hand, the suitability of five of the most popular feature detectors is tested on the car door model, which exhibits a low amount of details. This analysis resulted in the selection of the Shi and Tomasi [6] corner detector for integration in the feature extraction process.

On the second hand, the challenges of the proposed quality control application on the robustness of the feature tracking process, relying on the pyramidal implementation of the LK feature tracker [20, 22], are also investigated. As a result, the severe tracking errors obtained for the cases in which the robot or people temporarily appear in the MFs' region, support the transition to a robust tracking methodology.

Finally, the particularities of the stereoscopic sensor support the integration of the pyramidal LK tracker within the feature matching process, which also relies on the Shi and Tomasi corner detector. As a result, the limitations of the pyramidal LK tracker have also a strong impact on the feature matching system, which will also benefit from the continuous monitoring and supervision, which will be discussed in Section 5.4.

Chapter 5. Supervised Pose and Motion Estimation Solution

5.1. Introduction

This chapter details the methodologies selected for 3D reconstruction, motion estimation and robust supervision, and completes the description of the proposed pose and motion estimation solution. In Section 5.2 the problem of sparse 3D reconstruction is addressed, since it is vital to the process of generating the 3D macro-features' (MFs) point cloud that will be used in the motion estimation process. In addition to this, two different 3D reconstruction approaches, the former relying on a least-squares formulation and the latter building upon a geometric model, are analyzed. Section 5.3 introduces the problem of motion estimation from two distinct perspectives. Specifically, a "rotation-first" least-squares approach is compared with a methodology which solves simultaneously for both the rotation and translation components, based on Euclidean geometry principles.

The transition to a robust pose and motion estimation solution can only be accomplished by adding an upper layer to the general architecture relying on feature extraction, tracking, and matching, followed by 3D reconstruction and the estimation of rigid motion. This original layer, called supervisory level, is described in Section 5.4. Its primary objective is to make sure that the robotic marking station receives "timely-mannered", accurate and fault-tolerant visual servoing data. When compared to different visual servoing architectures, relying on multiple vision sensors, 3D CAD models of the inspected industrial objects, and controlled backgrounds [30, 64, 65], the proposed supervisory system solves, from a software perspective, the challenging task of estimating the pose and motion of an automotive part exhibiting very few reliable features over its surface, and performing its motion in a complex factory environment.

Finally, the validation of the accuracy of the pose and motion estimations is analyzed in Section 5.5, in which a metrology system is introduced as a low-cost alternative to the complex architecture developed by Chang *et al.* [66].

5.2. Sparse 3D Structure Estimation

The stereo-correspondence problem, addressed in Section 4.5, together with the validation and correction gates embedded in the supervisory level, which will be discussed in Section 5.4, provides the refined MF matches which are the input to the 3D

sparse structure estimation sub-system, illustrated in Fig. 3.13. Then, as mentioned in Section 3.4, the full calibration of the passive stereoscopic sensor is also used in the 3D reconstruction process, resulting in real world units to describe the sparse topology of the tracked object with respect to the world reference frame, O_R , attached to CamR, as displayed in Fig. 3.8.

For the 3D sparse structure reconstruction, two different methodologies were selected, from the variety of approaches described in Section 2.2.4. The first technique is related to the linear triangulation [45], whereas the second method corresponds to the mid-point triangulation [42].

Under the linear triangulation framework, given an unknown point in 3D space, X , expressed in homogenous coordinates, and its homogenous projections $x_{CamR}(x', y', 1)$, $x_{CamL}(x, y, 1)$, on the two camera planes π_R and π_L , it can be written:

$$\begin{cases} x_{CamR} = P_{CamR} \cdot X \\ x_{CamL} = P_{CamL} \cdot X \end{cases} \quad (5.1)$$

where P_{CamR} , P_{CamL} are 3x4 projection matrices computed with the camera calibration data. By imposing the condition that for each equation of the above system the cross product between the right and left sides should be null, the homogenous scale factors can be eliminated, resulting in a system of four equations in four homogenous unknowns, $A \cdot X = 0$, with:

$$A = \begin{bmatrix} x p_{CamL}^{3T} - p_{CamL}^{1T} \\ y p_{CamL}^{3T} - p_{CamL}^{2T} \\ x' p_{CamR}^{3T} - p_{CamR}^{1T} \\ y' p_{CamR}^{3T} - p_{CamR}^{2T} \end{bmatrix} \quad (5.2)$$

where p_{CamL}^{iT} , p_{CamR}^{jT} are the i -th and j -th rows of P_{CamL} and P_{CamR} , respectively. Finally, the solution, X , is represented by the singular vector corresponding to the smallest singular value of the matrix given by eq. (5.2).

As specified in Section 2.2.4, the linear triangulation approach has the advantage of providing a closed-form solution, by using a computationally efficient methodology. However, the non-optimally estimated 3D point does not exactly validate the projective geometry.

The mid-point triangulation approach provides a geometrical formulation to the 3D reconstruction problem, accounting for the frequent cases in which the rays projected from the corresponding features (in the image planes) to the 3D world do not necessarily

intersect, causing uncertainty in the 3D reconstruction process. Therefore, under this framework the computed 3D point is given by the mid-point of the segment perpendicular to the two projected rays, as discussed in [42].

The two procedures for sparse 3D structure reconstruction were applied on the set of MFs' correspondences, which were *a priori* refined with Sampson's first-order correction, which will be discussed in Section 5.4.2.1. Although the two triangulation methods exhibit considerable changes in their mathematical formulation, the obtained 3D results are extremely similar, as depicted in Fig. 5.1a in which the differences in the X, Y and Z-coordinates of the 3D MFs reconstructed with both methods, are represented for the first frame processed in the tracking procedure of the car door. These differences, which have an order of 10^{-10} cm, are computed by subtracting the 3D location of the MFs obtained with the mid-point triangulation, from the 3D results given by the linear 3D reconstruction approach.

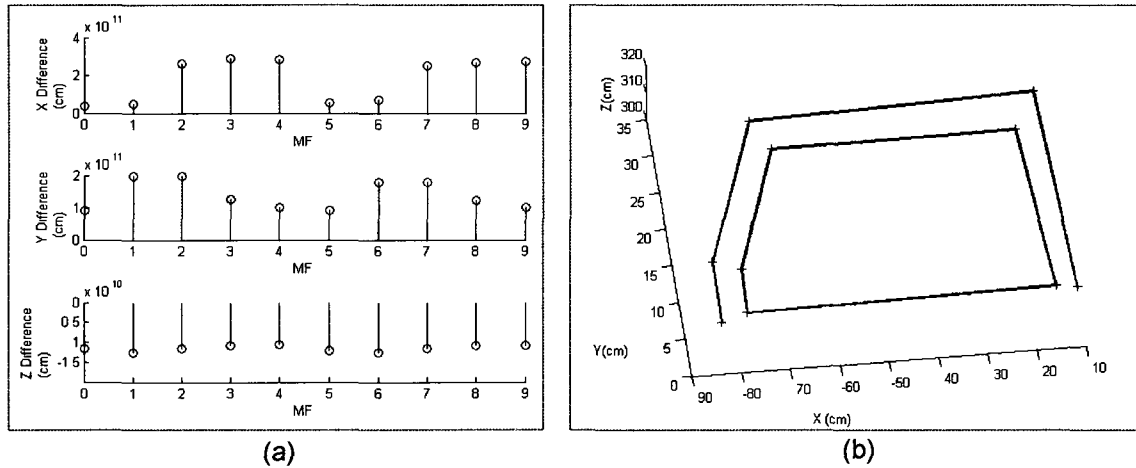


Fig. 5.1. (a) X, Y, Z-differences in the 3D location of car door's MFs, estimated with linear versus mid-point triangulation, (b) sparse 3D structure of car door's window frame with linear triangulation.

In order to further inspect the accuracy of the 3D reconstruction, the obtained 3D MFs are back-projected into their image planes, according to eq. (5.1). Similar to the effects pointed out in Fig. 5.1a, the differences in the back-projection errors (in the x and y coordinates defined in the image plane) for the two methods are very small, since these back-projection errors exhibit a 10^{-14} pixels order of magnitude with the mid-point triangulation, and 10^{-12} pixels for the linear triangulation, respectively. In Fig. 5.1b, the 3D sparse structure of the car door's window frame, obtained with the linear triangulation approach, is represented.

In conclusion, by taking into account the similar performances obtained for the two 3D reconstruction methods, in terms of the back-projection error, the simpler implementation of the linear triangulation constituted the prerequisite for its integration in the final prototype of the pose and motion estimation solution.

5.3. Motion Estimation

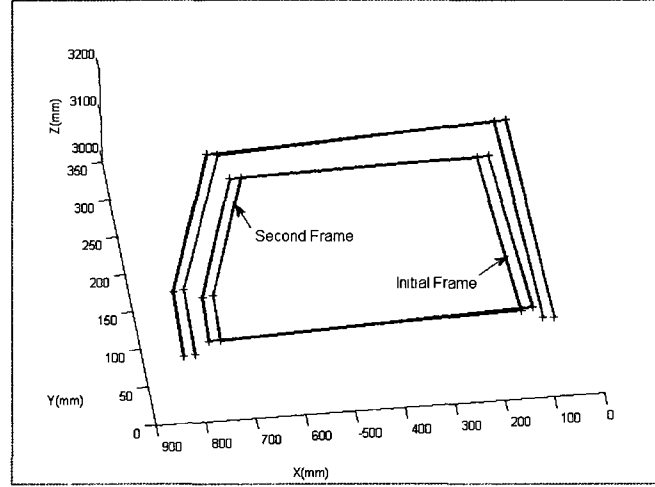
Once the 3D sparse structure of the automotive object has been reconstructed in two subsequent views, the motion estimation sub-system, illustrated in Fig. 3.13, is responsible for computing the rigid transformation exhibited by the object between the two successive captured frames. The rigid transformation linking the two sparse 3D structures, supported by the MFs, is completely characterized by a 3x3 rotation matrix R , and a translation vector T . Specifically, every 3D point, X_i^t belonging to the current structure, reconstructed at the time t_i , can be obtained from the corresponding point X_i^{t-1} in the previous frame, estimated at time t_{i-1} , according to the equation:

$$X_i^t = R \cdot X_i^{t-1} + T. \quad (5.3)$$

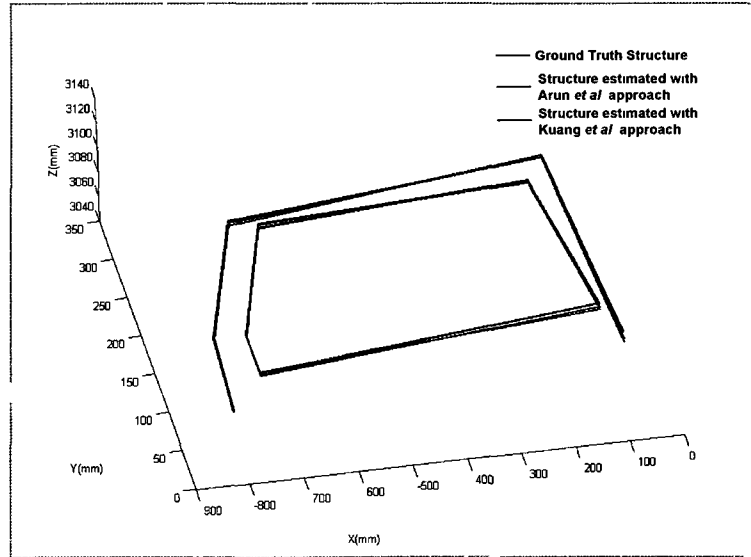
From the motion estimation approaches discussed in Section 2.2.4.3, two methodologies, which provide a closed-form solution for the (R,T) pair, were re-implemented in Visual C++ with the OpenCV library [21]. The criteria for selecting the two techniques were related to the precision obtained for the motion estimation and their associated computational complexity. The small number of MFs used in the motion estimation procedure preempted the selection of a RANSAC-based approach for robust (R,T) calculation [42, 54].

The first selected motion estimation technique, introduced by Arun *et al.* [57], represents a “rotation-first” approach, which builds upon least-squares for computing the (R,T) pair, based on the singular value decomposition (SVD) of a well conditioned 3x3 matrix. In order to eliminate the translation component, the centroid is computed for each set of 3D points, and then subtracted from each component belonging to its associated point cloud [57]. The second methodology which solves simultaneously for the (R,T) pair was developed by Kuang and Liu [59] and uses Euclidean geometry for computing the axis-angle representation of the rotation matrix, based on the fact that the increments of the difference of feature points on which the rotation has been applied are perpendicular to the direction of rotation. In order to experimentally validate the two techniques, the 3D point clouds obtained for the car door’s MFs in the first two frames processed during the

tracking sequence were used as input to the motion estimation sub-system. Under this experimental scenario, the car door's motion was mainly a translation along the X axis of CamR, as represented in Fig. 3.8a. Moreover, the 3D sparse structures, shown in Fig. 5.2a, as expressed with respect to CamR, were reconstructed with the linear triangulation, discussed in the previous section.



(a)



(b)

Fig. 5.2. (a) 3D sparse structure of car door's window frame reconstructed in the first two processed frames, (b) real/estimated 3D sparse structure in the second frame.

In the early phases of the experimentation of the pose and motion estimation solution, the two measures for validating the accuracy of the feature matches, introduced in Section 4.5.2, were used to discard the 3D MFs which had an epipolar residual greater than 0.1 or a distance to their associated epipolar line greater than 0.5 pixels. In this way, the 3D erroneous MFs were not used in the motion estimation procedure since

their weight would have substantially affected the accuracy of the obtained (R,T) pair, which is, in any case, estimated with a limited number of 3D MFs. However, after the integration of the supervisory layer, which will be described in Section 5.4, all the 3D MFs pass the proposed 3D outlier removal test, since their related epipolar residual and distance to the associated epipolar line have a magnitude inferior to 10^{-10} pixels.

The (R,T) pairs obtained with the two methods were used to estimate the location of the MFs in the second frame, extracted at time t_1 . Specifically, by considering the two sets of MFs, $\{X_i^{t_0}\}_{i=0,\dots,9}$, $\{X_i^{t_1}\}_{i=0,\dots,9}$, reconstructed at times t_0 , t_1 , the estimated set of MFs at time t_1 , $\{\hat{X}_i^{t_1}\}_{i=0,\dots,9}$ is populated by applying eq. (5.3), with the two pairs of computed (R,T). The estimated sparse structures, obtained with the two selected approaches, are represented in Fig. 5.2b along with the ground truth structure, $\{X_i^{t_1}\}_{i=0,\dots,9}$, provided by the 3D sparse structure reconstruction sub-system.

In order to further analyze the precision of the motion estimation process, two different error measures have been selected, which are related to the maximum of the absolute error, $M_{|\varepsilon|}$, and the RMSE (“Root Mean Squared Error”) ε_{RMS} . In accordance with the notation introduced above, the estimation error ε , in the case of the car door model, is a 10-dimensional vector whose elements are given by:

$$\varepsilon_i = X_i^{t_1} - \hat{X}_i^{t_1}. \quad (5.4)$$

Then, $M_{|\varepsilon|} = \max_{i=0,\dots,9} |\varepsilon_i|$, whereas the RMSE ε_{RMS} can be computed by using the formula:

$$\varepsilon_{\text{RMS}} = \sqrt{E(\varepsilon^2)} \quad (5.5)$$

where $E(\varepsilon^2)$ is the mean value of the squared estimation error vector ε^2 . Both error measures, $M_{|\varepsilon|}$ and ε_{RMS} are 3-dimensional vectors and their values are reported in Table 5.1.

As it can be noticed from Table 5.1, the technique proposed by Arun *et al.* [57], provides better motion estimation results than the approach introduced by Kuang and Liu [59], despite the slightly higher computational complexity of the latter. Thus, the superiority of Arun’s *et al.* method over the second methodology is observed in all the components of $M_{|\varepsilon|}$ and ε_{RMS} .

Table 5.1. Error analysis for the motion estimators.

Error Measure		Arun's <i>et al.</i> Method	Kuang and Liu's Method
Maximum of the Absolute Error $M_{ e }$ (mm)	$M_{ e }^X$	0.45	0.63
	$M_{ e }^Y$	0.78	1.89
	$M_{ e }^Z$	2.24	6.7
Root Mean Squared Error ϵ_{RMS} (mm)	ϵ_{RMS}^X	0.24	0.32
	ϵ_{RMS}^Y	0.38	0.95
	ϵ_{RMS}^Z	1.33	2.89

With the purpose of improving the precision of Kuang and Liu's approach [59], a hybrid version of the two approaches was also tested. Since Kuang and Liu also provided a method of estimating the rotation, based on the *a priori* known translation, the integration of the two methods was straightforward. Specifically, the translation vector estimated with the first technique was used to compute the rotation matrix with the second approach. Although this technique resulted in smaller errors than the approach proposed by Kuang and Liu [59], the methodology introduced by Arun *et al.* [57] maintained its superiority, in both accuracy and computational efficiency, when compared to this "hybrid" method. As a result, the technique of Arun *et al.* [57] was privileged for the final pose and motion estimation prototype.

5.4. Proposed Supervisory Level

The challenges identified for the feature extraction, tracking and matching processes, together with the strong requirement for precise pose and motion estimation data were the prerequisites for the development of the supervisory layer, illustrated in Fig. 3.13. The purpose of this section is to describe the techniques embedded in the supervisory level, which is responsible for the continuous monitoring and validation of the extracted features, along with their associated motion vectors and disparities.

This section is divided in two major parts which describe the coarse and fine level supervision procedures of the feature tracking and matching processes, which both aim at improving their precision, stability and robustness. Also, the implications of the supervisory system in the MFs detection process are discussed. The objective of the

coarse-level supervisory system is to correct the severe tracking errors associated to the LK tracker in the case of occlusions, illumination changes and variations in the reflectance properties (“highlights”) of the automotive panel throughout the motion sequence. The fine-level supervisory system further refines the precision of the feature tracking and matching processes, by correcting the drift accumulating with the tracking, which is characteristic to point-based feature trackers. While Kak *et al.* [30, 64, 65] use three different vision sensors in order to design a fault-tolerant, reliable architecture for robotic interaction, the current work tries to solve a considerable part of the problems they identified, from a software perspective, thus with no further complications of the hardware platform. The two parts of the supervisory layer provide the system with the robustness to function under industrial settings with complex backgrounds, unlike the approach described by Kak *et al.* in which a controlled environment with distinctive background, along with a 3D wire-frame model of the tracked part, are necessary for proper pose and motion estimation.

Before detailing the coarse and fine level supervisory layers, it should also be noticed the involvement of the supervisory system in the feature extraction process. As discussed in Section 4.2.1, the selective feature detection process is guided by the supervisory layer, which performs the extraction of the regions of interest (ROIs) containing the industrial part. Nevertheless, the feature tracking results, which are carefully monitored by the supervisory sub-system, have a strong influence in the propagation of the information regarding the ROI which contains the MFs’ area.

5.4.1. Coarse-Level Supervision of the Feature Tracking and Matching Processes

The sensitivity of the pyramidal LK tracker [20, 22] to occlusions, changes in illumination, or the momentary appearance of different objects in the monitored scene, is well illustrated by Fig. 4.17, in which the motion vectors should have pointed to the MFs shown in Fig. 4.1. Moreover, due to the selected approach of tracking the MFs in both stereo-views, the poor performance of the LK tracker has direct implications in the feature matching process as well, which highly depends on the optical flow results. The objective of the proposed validation gate, embedded in the coarse-level supervisory sub-system, is to correct the erroneous motion vectors returned by the feature tracker and to replenish the set of MFs for the cases in which the robot or people temporarily occlude a certain part of the MFs’ area. Under the settings of the proposed pose and motion

estimation system, the replenishment process is essential for the considered application, because of the limited number of MFs that are available to the estimation process.

For the design of the coarse-level validation gate, advantages are taken from the available knowledge regarding the 2D topological structure of the tracked object, given by the MFs extracted at re-initialization. Nevertheless, as it will be explained in Section 5.4.2, the 2D topological structure is also re-updated during the tracking sequence (through the implications of the fine-level supervisory layer), in order to account for the slight scaling effects that might be exhibited by the automotive part during the motion cycle. These scaling effects are manifested in the scenarios in which the rigid body gets slightly closer (or farther) with respect to the acquisition system, and can be related to motion patterns which can be characterized by a translation along the X_o axis, as represented in Fig. 4.5, coupled with slight rotations around the Y_o axis. Additionally, this type of motion is also encountered in cases in which the assembly line portion that appears in the view of the stereo-vision system contains a curved section. As a result, for the success of the developed coarse-level supervisory system, only one prerequisite needs to be guaranteed. This condition is related to the requirement that the scaling effects, exhibited by the automotive part between two subsequent frames extracted for processing are not substantial. Thus, by taking into account the characteristics of the industrial application considered, in which the automotive object does not embed a motion generator of its own, this condition is consistently validated. Based on this prerequisite, the proposed confidence measures can be performed in the 2D space, with data extracted from the images.

The re-initialization stage, that is when a new item appears on the assembly line, is related to the registration of the initialization frame, in which the assembly engineer has pre-selected the MFs, with the currently extracted frame, triggered by the object detection module. The topological structure of the MFs' area in 2D is therefore available at the beginning of the tracking cycle. As a result, advantage is taken of this re-initialization in order to extract useful knowledge that will give the "signature" of the region being tracked, in both views. This knowledge is encoded in two buffers, which are populated with the 2D Euclidean distances and the relative displacements between the MFs, as explained in Section 4.2.3.

The proposed test for correcting erroneous motion vectors is illustrated in Fig. 5.3 and consists of finding a pair of tracked MFs that have a high level of confidence, based on the assumptions made at the beginning of this section. Consequently, since the need

for having a certain level of redundancy is important in image processing techniques, the proposed validation gate, which is performed in both views, is three-fold.

The first test verifies that the difference involving the Euclidean distance between the MFs (MF_i, MF_j) forming the pair, d_{ij} , and their corresponding distance saved in the buffer of 2D distances, $\hat{d}_{ij}$ is within a ± 2 pixels threshold. If the considered pair of MFs verifies this validation gate, the second test is conducted, which checks whether the difference involving the relative distances between the selected MFs, δ_{ij} , and the “ground-truth” relative displacements stored in the buffer of relative distances, $\hat{\delta}_{ij}$ is also within a ± 2 pixels threshold.

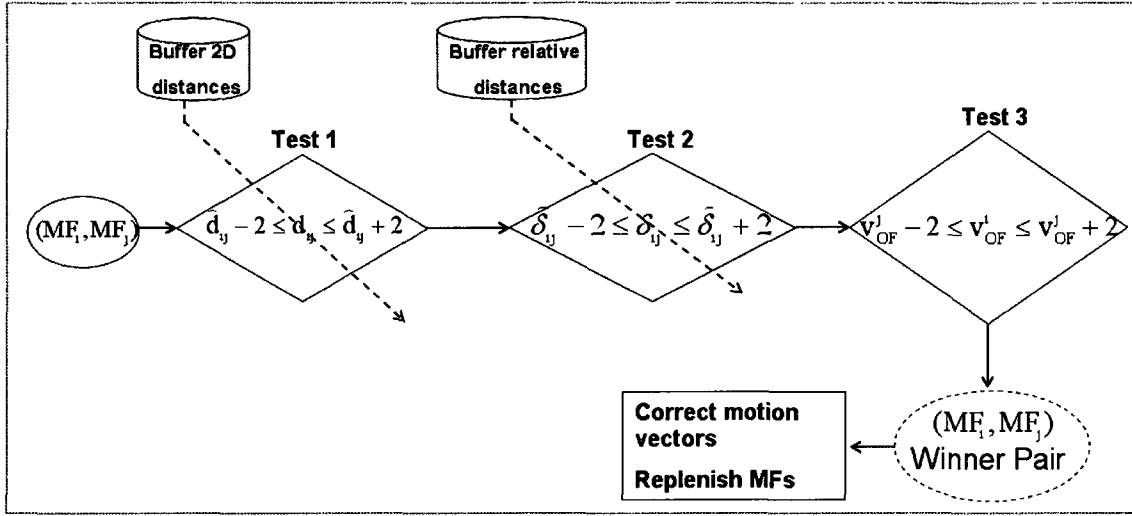


Fig. 5.3. Validation gate for finding the correctly tracked MFs pair (MF_i, MF_j) .

Finally, if the pair of MFs passes the second test as well, a third validation procedure is imposed on the norms of their associated motion vectors, v_{OF}^i, v_{OF}^j , since the keypoints belonging to a rigid structure should move in a relatively similar manner. Therefore, in this last test it is verified that the difference between the computed optical flow in the x and y directions for both MFs forming the pair is within a ± 2 pixels limit. This threshold was selected in a way that takes into account the non-linear mapping between 3D world and the image plane, according to which, the points closer to the camera seem to have larger motion vectors than the ones farther away. The three-fold validation gate, shown in Fig. 5.3, is ended as soon as a pair of MFs that passes all its embedded tests is found. This MFs pair (MF_i, MF_j) , is regarded as a “winning pair” and its embedded information about the motion vectors is used to both correct the erroneous optical flow data and replenish the set of MFs for the cases in which some of them are lost in the

tracking as an effect of occlusions, illumination changes, specular reflections or temporary appearance of different entities in the inspected scene.

Unlike the approach suggested by Kak *et al.* [30, 64, 65] for acquiring robustness to occlusions, that builds upon a 3D wire-frame model of the inspected object which is coupled with image processing techniques performed on the data grabbed by three additional vision sensors, the technique proposed in this section only makes use of a 2D topological structure of the MFs' area in the image plane. Figures 5.4a and 5.4b represent two processed frame segments, grabbed by CamL during the tracking procedure, after applying the validation gate shown in Fig. 5.3. In both frame segments, the winning MFs pair is (MF_0, MF_1) and the corrected motion vectors are represented in black, whereas the optical flow vectors related to the replenished MFs are drawn in white.

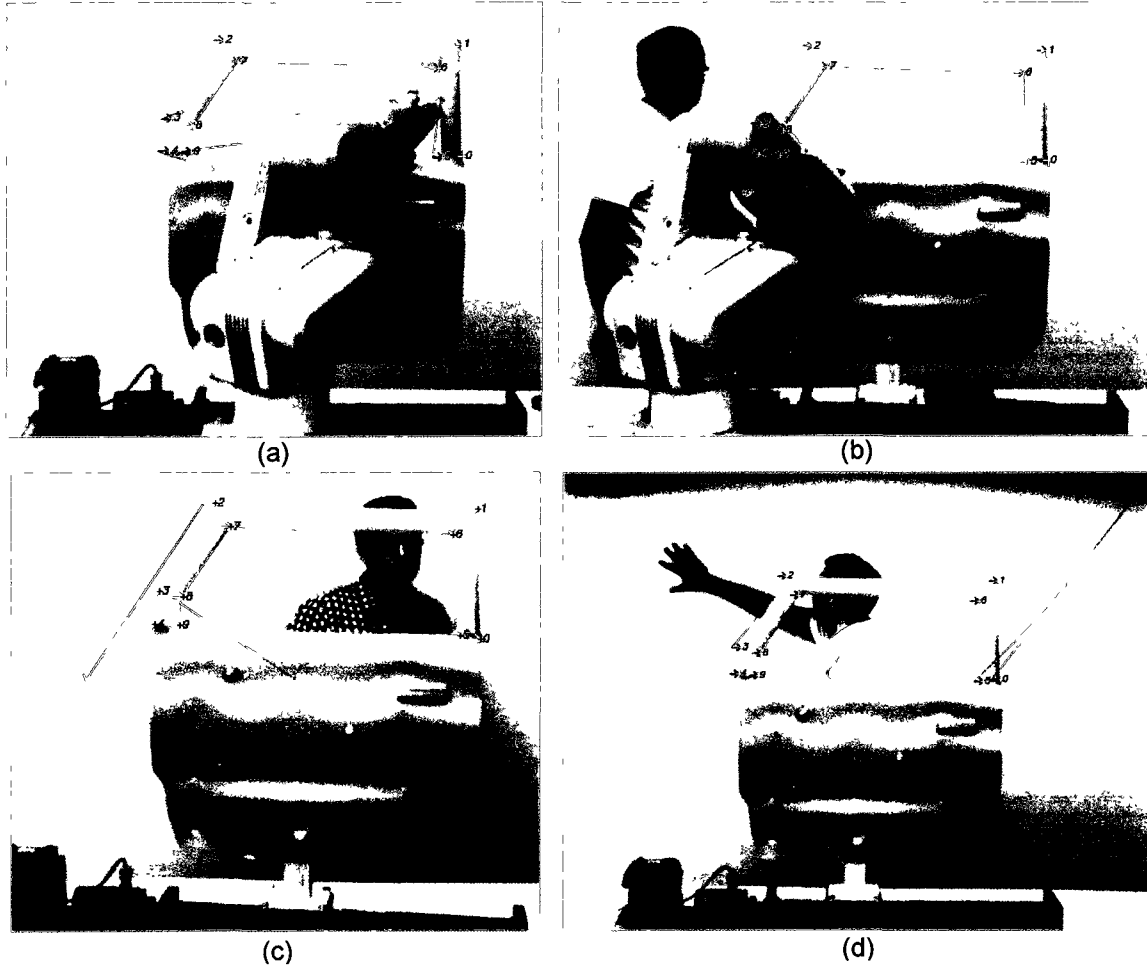


Fig. 5.4. Optical flow results after applying the coarse-level validation gate for the cases in which: (a) the manipulator robot is present in the scene, (b) a person and the manipulator robot are present in the scene, (c) a person is present in the scene and the winning pair is found in the adjacent stereo-frame (CamR), (d) a person is present in the scene and the winning pair is not found.

From Fig. 5.4a and 5.4b it can be noticed that the proposed supervisory gate provides the feature tracking process with stability and robustness to the second class of errors [24] affecting the pyramidal LK tracker [20, 22]. The latter are related to cases in which the brightness constancy assumption is not satisfied due to the increase in the complexity of the scene throughout the motion experiment. Therefore, as shown in Fig. 5.4a and 5.4b, the proposed coarse-level supervisory gate maintains the location of the MFs within their 2D topological structure, even in the cases in which the robot or even people are present in the monitored scene. As a result, the correction of the severe tracking errors, highlighted in Fig. 4.17, has a beneficial effect on the pose and motion estimation process which will continue maintaining the same precision needed for the robotic marking of deformations.

A question that arises is related to situations in which it is impossible to find a winning pair in the frame grabbed by one of the cameras, as shown in Fig. 4.17a, where all the motion vectors are erroneous, or even worse, in both grabbed stereo-frames. Therefore, in the case in which the winning pair is found in only one frame, the result regarding the correct norm of their associated motion vectors is extrapolated in the adjacent frame in space. Subsequently, based on the topological structure knowledge contained in the buffers of 2D distances and relative distances, with respect to that view a replenishment of the full set of MFs is then performed. This special case is illustrated by Fig. 5.4c which is a segment grabbed from a frame acquired by CamL in which the winning pair could not be found, and the results regarding the winning pair obtained for CamR's frame were used. In Fig. 5.4c, the red motion vectors are related to the optical flow data returned by the LK tracker, whereas the white motion vectors, which are linked to the proposed extrapolation technique, point to the proper MF location. The suggested extrapolation is reasonable for the considered setup, since the cameras forming the stereoscopic sensor are approximately parallel, resulting in minimal perspective distortion between the stereo-views. The substantial distance from the acquisition system to the region of interest containing the automotive part also supports this procedure.

Finally, in the situation in which the search for the winning MFs pair is unsuccessful in both views, use is made of the motion vectors computed for the previous set of frames in the tracking sequence, and the frame extraction process is automatically triggered. This solution is adequate for the considered industrial application where the speed of the assembly line is approximately constant. In order to better support this

special case, Fig. 5.4d shows a segment from a frame grabbed by CamL in which no winning pair could be found, and the erroneous motion vectors represented in red, would have resulted in very unsafe visual servoing control data for the robotic station. Thus the motion vectors, drawn in blue, were replenished according to the optical flow data obtained from the processing of the previous frame. Since the proposed solution for feature matching relies on the results computed with the pyramidal LK tracker in both views, the correction of the tracking error through the robustness added to the feature tracking process, has beneficial effects in the feature correspondence process as well.

To sum up, by introducing the correction gate in Fig. 5.3, the extrapolation to a “hybrid” feature tracking is performed, in which the minimal *a priori* geometrical structure of the automotive part is used to supervise the tracking and correct erroneous motion vectors. As a result, the errors affecting the pyramidal LK tracker [20, 22] due to the increase in the complexity of the scene, introduced in Section 4.4.2, are minimized with the help of the coarse-level supervisory sub-system, based on a geometric perspective, in which the configuration of the target object plays a very important role. Moreover, the tests in Fig. 5.3 provide robustness not only to occlusions or the temporary appearance of other entities in the scene, but also to photometric variations, exhibited by the panel throughout the motion sequence.

As a result, the coarse-level supervisory layer is able to solve a variety of real constraints, without changing the mathematical foundation of the feature tracker, unlike in [36] where the authors modify the optimization problem of the LK tracker in order to make it robust to illumination changes. The latter formulation still remains sensitive to occlusion effects, or to the appearance of other objects in the scene, which are omnipresent situations in an automotive assembly industrial setting. With the proposed strategy, the useful information stored in the 2D topological buffers is not only used to correct the erroneous motion vectors, but also to replenish the MFs set. Therefore, the same number of MFs, as at re-initialization, is made available to the pose and motion estimator throughout the entire tracking sequence, which increases the stability of the procedure.

5.4.2. Fine-Level Supervision of the Feature Tracking and Matching Processes

As discussed in Section 4.5.2, the early version of the feature monitoring stage [72] relied on the feature extraction data computed with the Shi and Tomasi corner

detector [6], which was applied multiple times during the motion cycle of the panel. However, this approach does not represent a reliable alternative due to the appearance of robot or people in the scene, and the slight changes in the reflectance properties of the object's surface.

This section is divided in two parts. Sampson's first-order correction of the feature correspondences, and the improvements on the precision with which the epipolar constraint is validated, are discussed in the first part. A proposed gate for correcting the drift accumulating with the tracking for further improving the precision of the tracking and matching processes is detailed in the second part.

5.4.2.1. Sampson's First-order Correction

Hartley and Zisserman [45] have studied in detail Sampson's first-order approximation, which has beneficial effects in correcting feature matches, such that the epipolar constraint is better validated. As mentioned in Section 2.2.3, Sampson's correction builds upon an optimization criterion whose objective is to minimize the sum of the distances from the matches to their corresponding epipolar lines, for obtaining consistent epipolar geometry. Specifically, for a particular set of feature correspondences $\{x_{CamR}(x, y, 1), x_{CamL}(x', y', 1)\}$, expressed in homogenous coordinates, the correction is made according to the following equation [45, 49]:

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{x}' \\ \hat{y}' \end{bmatrix} = \begin{bmatrix} x \\ y \\ x' \\ y' \end{bmatrix} - \frac{x_{CamL}^T \cdot F \cdot x_{CamR}}{(F \cdot x_{CamL})_1^2 + (F \cdot x_{CamL})_2^2 + (F^T \cdot x_{CamR})_1^2 + (F^T \cdot x_{CamR})_2^2} \begin{bmatrix} (F^T \cdot x_{CamL})_1 \\ (F^T \cdot x_{CamL})_2 \\ (F \cdot x_{CamR})_1 \\ (F \cdot x_{CamR})_2 \end{bmatrix} \quad (5.6)$$

where F is the fundamental matrix, whereas $(\Theta)_i$ represents the i -th element of the 3D vector Θ . From a geometrical perspective [49], Sampson's correction is similar to an orthogonal projection of the corresponding points into a tangent plane to the manifold of the fundamental matrix.

In order to demonstrate the beneficial effects of Sampson's first-order correction, Fig. 5.5a illustrates a graph of the epipolar residuals associated to the car door's MFs, at the beginning of the tracking sequence, before and after Sampson's refinement. Due to the low computational complexity of Sampson's correction, a threshold of 10^{-15} was imposed on the magnitude order of the epipolar residuals, which resulted in the necessity of applying the refinement twice. As it can be seen from Fig. 5.5a, the epipolar residuals related to the car door's MFs are successfully minimized. With the purpose of

providing some insight into the actual corrections in the location of the MFs in the image plane, Fig. 5.5b shows the “corrections” in the x and y coordinates of the MFs, in both stereo-views. These “corrections” quantize how much the x and y locations of the MFs changed, after applying Sampson’s first-order correction, and are simply computed by subtracting the refined coordinates of the MFs from the MFs’ coordinates before the refinement. As noticed from Fig. 5.5b, all “corrections” share a sub-pixel order, with a noticeable higher magnitude in the y coordinate.

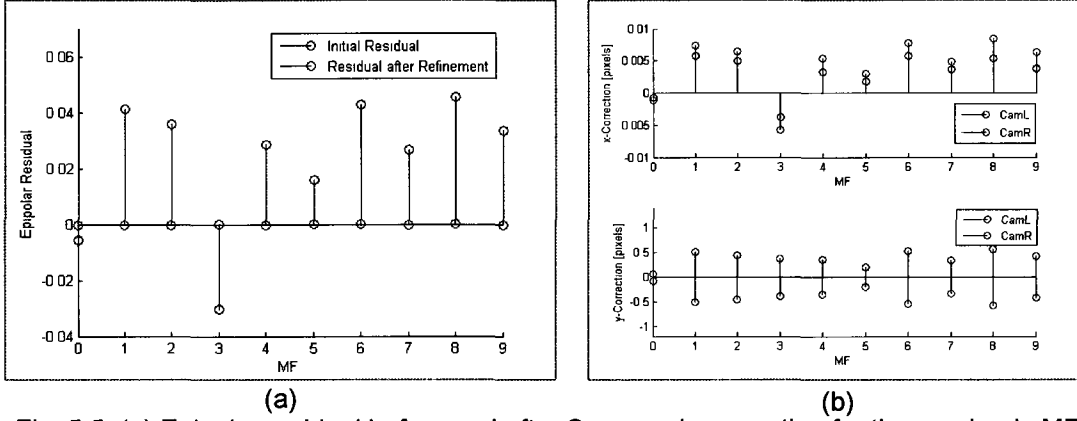


Fig. 5.5. (a) Epipolar residual before and after Sampson’s correction for the car door’s MFs, (b) corrections in the location of the MFs in both stereo-frames.

The implications of Sampson’s first-order correction on the second measure for inspecting the feature correspondences, related to the distance from the MFs to their associated epipolar line, as discussed in Section 4.5.2, resembled the same effect obtained in the case of the epipolar residuals. Therefore, applying Sampson’s refinement twice, the distances to the epipolar lines decreased with a 10^6 factor, from an initial maximum order of magnitude of 1.6 pixels.

To conclude, Sampson’s correction has important beneficial effects in better imposing the epipolar constraint among the MFs, in the cases in which the fundamental matrix of the stereoscopic sensor is accurately known. However, this refinement stage is not able to provide the corrections needed for the fourth class of errors affecting the tracking, and matching respectively, related to the drift accumulated with the tracking over long sequences. As shown in Fig. 5.5b, if the initial MFs are already displaced, due to drifting effects, the sub-pixel level corrections are not strong enough to correct the drift accumulation. This is related to the fact that Sampson’s first-order correction refines the *a priori* given correspondences, although they might be drifted (pointing to noise patterns in the area of the MFs). Although this correction has beneficial effects from a “pair-wise” perspective over the pose estimation problem, the cumulative tracking errors affect the

motion estimations, which result in inaccuracies in the robotic guidance. That is why, the fine-level supervisory layer should embed a more robust monitoring phase in order to successfully account for the small but important errors which appear in tracking the same set of features over a considerable number of frames [6, 36], as discussed in Section 4.4.2.

5.4.2.2. Proposed Fine-level Drift Correction

The immediate effect of the drift associated with the pyramidal LK tracker [20, 22] is on the “lifetime” of the MFs, and what is remarkable is the fact that, in most cases, the accumulating drift cause the tracking of a distinct feature in the area of the initially identified MF, based on the approximate selections explained in Section 4.2.1. This is proved by the fact that the norm of the motion vector associated to a drifted MF, at a particular time, is almost similar to all the other correct norms related to the stable MFs. This phenomenon can be related to the conclusions of Jin *et al.* [36], since the substitution of the MF by a different feature in its neighborhood results in the re-initialization of the tracking process for that particular keypoint, in which the inter-frame feature correspondence operation is applied on a smaller size time span. As a result, the integration time of the computed trajectory is decreased, which results in good tracking performance for that particular feature, although it might not belong anymore to the physical structure of the inspected object. As the pose and motion estimations constitute the basis of the robotic interaction with the moving panel, one of the most important effects of the drift is related to the uncertainty that is transferred to the robotic marking station, which will influence the attitude of the robot’s end-effector with respect to the deformations that need to be spotted. The considerable amount of drift accumulating with the tracking is illustrated in Fig. 5.6, in which the location of the MFs is shown in two segments, belonging to the 1st (considered as ground-truth, GT) and 20th frames grabbed by CamR during the motion experiment.

In order to increase the generality of the proposed refinement technique, in the current experiment, the sled system is displaced such that the car door model can exhibit the pose changes corresponding to Scenario 1 in Table 4.1, at re-initialization. Since the car door is rigidly attached to the sled system, these changes affect its trajectory, meaning that the motion of the object does not happen in a plane that is almost perpendicular to the principal axis of the stereo-vision system. The pose changes also introduce a slight scaling effect in the image planes, as the car door gets slightly closer to

the acquisition system during its motion experiment. As it can be noticed in Fig. 5.6, almost all the MFs get considerably drifted throughout the motion cycle, especially those in the zoomed areas, even though the coarse-level correction is already embedded in the pose and motion estimation system.

To alleviate these problems, the fine-level supervisory system consists of a new validation method introduced with the purpose of assigning a model or a signature to the limited set of MFs, to aid in the final refinement of the feature matching and tracking processes. When compared to the coarse-level supervisory gate, the fine-level validation technique privileges the testing performed in 3D, due to the small, but significant scaling effects which affect the MFs' topological structure in the image plane.

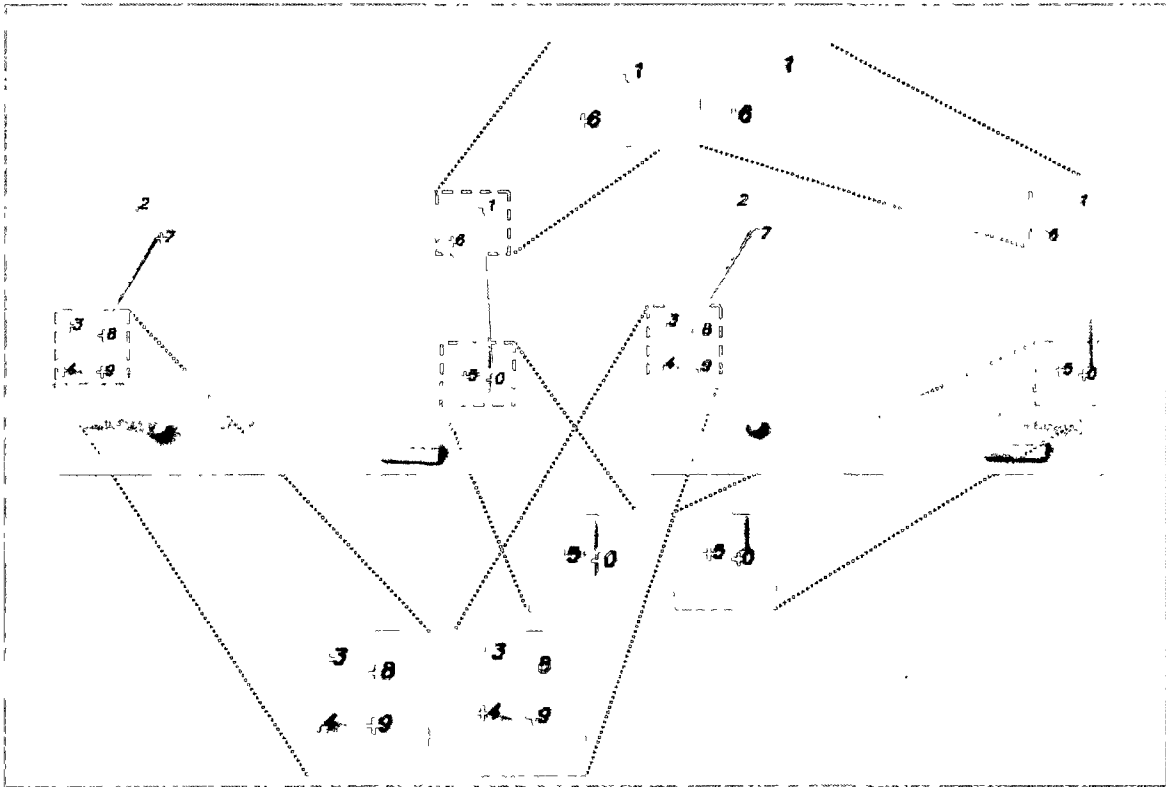


Fig. 5.6. Segments from the 1st (GT) and 20th processed frames grabbed by CamR, showing the effect of drift accumulating with the pyramidal LK tracker.

In order to attach a signature to the MFs such that the drift accumulating with the tracking can be reliably corrected, the proposed approach is inspired by an overlapping error methodology. Therefore, starting from an initial 3D structure of the MFs, corresponding to the GT frame, the primary goal is to verify how the subsequent 3D structures of the MFs, recovered in the tracking process, differ from their GT distribution. However, a challenging problem is related to the manner in which the GT and current 3D

MFs' structure can be reliably compared. Identifying the amount of displacement that needs to be applied to the GT rigid structure, in order to compare it to the currently recovered 3D structure, is a complex procedure since it involves relying on the motion estimation data. The latter is strongly affected by the precision of the feature tracking and matching results. Therefore, estimating the location of the GT structure at subsequent frame extraction times, by means of the frame-by-frame motion estimation data [57] would affect considerably the robustness of the proposed validation procedure. Nevertheless, as it was explained in Section 5.3, the motion estimation technique builds upon a least-squares approach, which also contributes to the uncertainty of the pose and motion estimator.

The proposed solution for identifying the MFs affected by drift relies on a normalization procedure [44, 45] that is also known as "pre-conditioning" in the numerical analysis literature. The general idea is to transform the 3D coordinates of the MFs defined with respect to the world reference frame, CamR, by bringing them to a normalized space, S^N . This normalization consists of two steps. First, the 3D points are

centered about their centroid $X_{\sim c} = \frac{1}{N_{MF}} \sum_{i=0}^{N_{MF}-1} X_{\sim i}$, and then scaled in a manner in which the

average distance from the origin is equal to $\sqrt{2}$, as proposed by Hartley [44, 45].

By transferring the 3D topological structure of the MFs, throughout the tracking, to the normalized space, the identification of the drifted MFs can be easily performed, with no need to estimate the location of the 3D GT structure at subsequent frame extraction times. In addition to this, the first step of the normalization procedure removes the translation component, which affects the 3D point clouds throughout the motion cycle. However, the rotation component is still present in the set of normalized MFs, and needs to be taken into account, in order to have a consistent validation technique. Although, given the relatively straight motion of the assembly line, the comparison between the normalized 3D structures can be performed directly (as the rigid structure is not significantly rotating throughout the motion experiment) an additional pre-processing step is introduced in order to acquire generality for the cases in which the motion of the rigid body also embeds a considerable rotation component. This pre-processing provides extra robustness to the proposed solution for cases in which the assembly line is slightly curved and the object might get closer to the acquisition system, or exhibit slight rotations around its Y_o axis, as displayed in Fig. 4.5. As a result, the pose and

motion estimation system acquires a higher generality level, since it is not constrained to function only under straight assembly line settings, and with object's motion patterns happening over a plane approximately parallel to the X axis of CamR, as illustrated in Fig. 3.8a. To do this, the inverse of the rotation matrix, extracted from the estimated rigid transformation relating the GT 3D structure with the current 3D structure, is applied to the normalized 3D MFs belonging to the current frame.

The normalized 3D structures for the GT and the 20th frame (with pre-processing) are illustrated in Fig. 5.7a. The flowchart of the developed fine-level supervisory gate is detailed in Fig. 5.7b. After removing the rotation component affecting the normalized 3D points of the current frame, the displacement between the two 3D normalized structures is evaluated. For this reason, at each step, a measure called “displacement residual” is computed for each MF, $\overline{MF}_i(\bar{X}_i, \bar{Y}_i, \bar{Z}_i)$, in S^N , according to the formula:

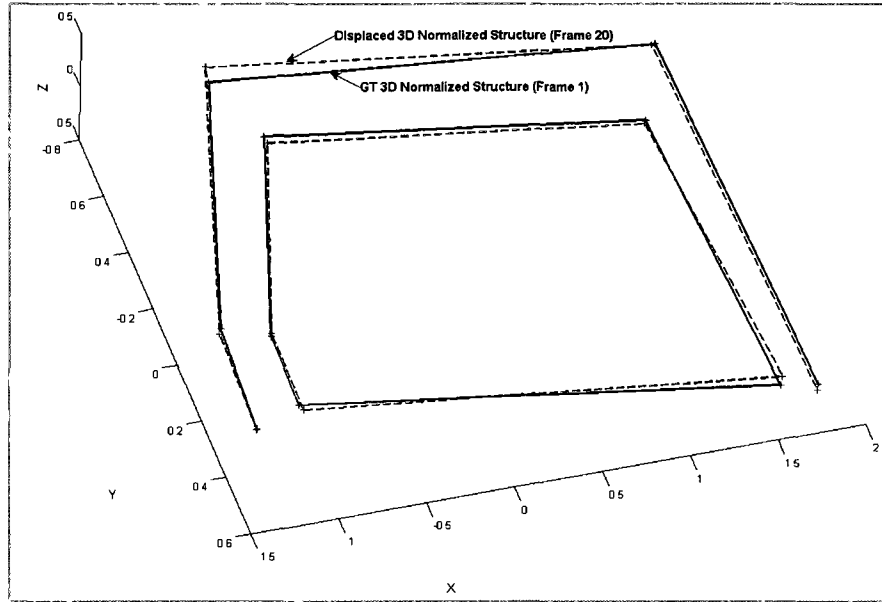
$$r_\delta(i) = |\bar{X}_i^{GT} - \bar{X}_i| + |\bar{Y}_i^{GT} - \bar{Y}_i| + |\bar{Z}_i^{GT} - \bar{Z}_i|, i = 0, \dots, N_{MF} - 1 \quad (5.7)$$

where the MFs belonging to the GT frame have the coordinates, $\overline{MF}_i(\bar{X}_i^{GT}, \bar{Y}_i^{GT}, \bar{Z}_i^{GT})$. Then, from the total set of MFs, the ones exhibiting a displacement residual greater than the threshold, $T_\delta = 0.02$, are classified as drifted, due to their locations with respect to the normalized GT structure. In the case of the data represented in Fig. 5.7a, all the MFs are categorized as drifted, since the fine-level correction gate was not applied yet.

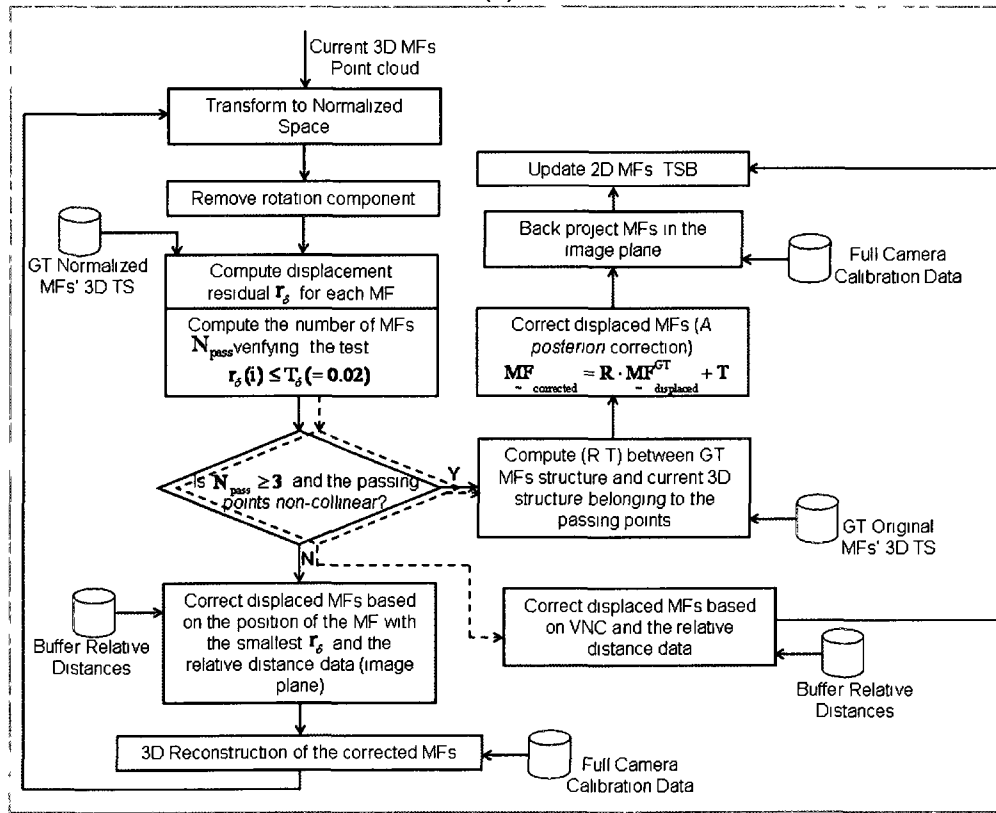
However, the correction of the drifted MFs cannot be performed by simply assigning it with the value of its corresponding MF belonging to the GT structure in the normalized space. This conclusion is evidenced by the fact that the inverse transformation to the original reference frame relies on the initial 3D positions of the MFs in the 20th frame that affected the calculation of both the centroid value and the scaling factor. That is why the proposed thresholding technique is only useful to identify the MFs that need to be corrected. Another aspect that needs to be mentioned is related to the suitability of this method for finding the drifted MFs. The method relies on the constancy of the number of MFs during the motion experiment, since a different number of MFs will result in totally different values for the centroid and the scaling factor, preempting a consistent comparison.

Nevertheless, the proposed replenishment process associated to the coarse-level supervision system is responsible for maintaining a constant number of MFs during

the entire tracking sequence, regardless of the occlusions caused by the robot arm or workers moving in the inspected scene.



(a)



(b)

Fig. 5.7. (a) 3D normalized structure of the car door's MFs in the 1st (GT frame) and the 20th frame, (b) developed fine-level supervision gate.

In the cases in which the number of MFs having a displacement residual smaller than the threshold is greater or equal to three, their associated original 3D coordinates (defined with respect to CamR) in the current frame, as well as their corresponding 3D features in the GT frame are employed in a motion estimation procedure, such that the rigid transformation that the object has undergone between the two views is computed.

However, this set of extracted MFs has to be validated, such that its components do not constitute a degenerate case, by forming a collinear configuration. The validation test (also discussed in [42]), which is performed in S^N , evaluates the distance between any two MFs in the selected set, together with the area defined by any three MFs. Since the extracted set of MFs might have a cardinal (i.e. number of elements) bigger than three, the proposed thresholding technique differs from that in [42]. Therefore, in order for the extracted set to form a non-degenerate configuration, the maximum distance between any two selected MFs has to be greater than $T_{\min_Dist} = 1.5 \cdot D_{\min}^{MFs}$, where $D_{\min}^{MFs}$ is the minimum distance between any GT MFs in S^N . In addition to this, the maximum area between any three selected MFs has to be greater than $T_{\min_Area} = 1.5 \cdot A_{\min}^{MFs}$, where $A_{\min}^{MFs}$ is the minimum area formed by any 3 MFs of the GT structure defined in S^N . The “1.5” coefficient appearing in the formulas for $T_{\min_Dist}$ and $T_{\min_Area}$ was selected experimentally. As specified in [42], these two conditions are necessary and sufficient, since by only applying the first condition on the configurations with three collinear points located far away, the rigid transformation estimation would be erroneous.

Once the extracted set of normalized MFs has passed the cardinality and non-collinearity test (meaning that the number of MFs exhibiting a displacement residual smaller or equal to T_δ is greater or equal to 3 and the extracted MFs do not form a collinear configuration), as depicted in Fig. 5.7b, the set, as defined with respect to CamR, along with its corresponding GT set, are used for estimating the rigid transformation exhibited by the rigid body between the GT and the current view. The least-squares “rotation-first” technique proposed by Arun *et al.* [57] is selected for the motion estimation process, as discussed in Section 5.3. Upon the calculation of the (R,T) pair, the MFs classified as drifted, are corrected based on their 3D location in the GT frame and the newly estimated rigid transformation, as shown in Fig. 5.7b.

The full calibration of the stereoscopic sensor makes possible the back-projection of the corrected MFs in the image plane and thus, the refinement of the 2D topological

structure of the MFs. Moreover, the average epipolar residual, introduced in Section 4.5.2, was also inspected and an order of magnitude of 10^{-12} was obtained, upon the back-projection of the corrected MFs. The proposed refinement can be categorized as an *a posteriori* correction, which relies on consistent 3D data in order to acquire robustness to scaling effects that might affect the MFs' patches throughout a more general motion cycle.

Once the topological structure of the MFs has been refined in the image plane, the buffers of 2D distances and relative distances, introduced in Section 4.2.3, are re-populated. This update procedure has beneficial effects in the case in which the rigid body gets closer to the acquisition system during its motion, resulting in scaling effects on the image plane. Moreover, the re-population of the buffer also enhances the correction phase of the coarse-level supervisory system, which relies on the overall distribution of the MFs in 2D. In this way, the drift accumulating with the pyramidal LK tracker [20, 22] is bounded, preempting the propagation of erroneous results as the ones belonging to the 20th frame, illustrated in Fig. 5.6.

Figure 5.8a shows the MFs belonging to the 20th frame, in the case in which the fine-level supervision gate is integrated with the pose and motion estimation solution, under the same experimental settings that gave the results in Fig. 5.6. Additionally, MF₉, marked in green is the only one that is detected as drifted from the GT structure, and it is corrected, at this step. Figure 5.8b shows the displacement residuals associated with the MFs' structure in the 20th frame, before and after the integration of the fine-level supervisory layer. These displacement residuals are calculated in S^N , after removing the rotation component between the structures in the 1st and 20th frame. From Fig. 5.8b, it can be observed the substantial drop in the magnitude of the displacement residuals due to the integration of the fine-level supervisory layer. Therefore, the displacement residuals associated to the MFs in the 20th frame from the prototype with the embedded fine-level supervisory sub-system are all below the imposed displacement threshold $T_\delta = 0.02$. Furthermore, Fig. 5.8c shows the MFs' structure in S^N , before and after the incorporation of the fine-level supervisory layer. Similar to the effects manifested by the displacement residuals, the new normalized structure resembles more the GT distribution of MFs, which maintain their positions over the surface of the car door model, without being drifted or "trapped" in noise patterns as the MFs belonging to the 20th frame in Fig. 5.6.

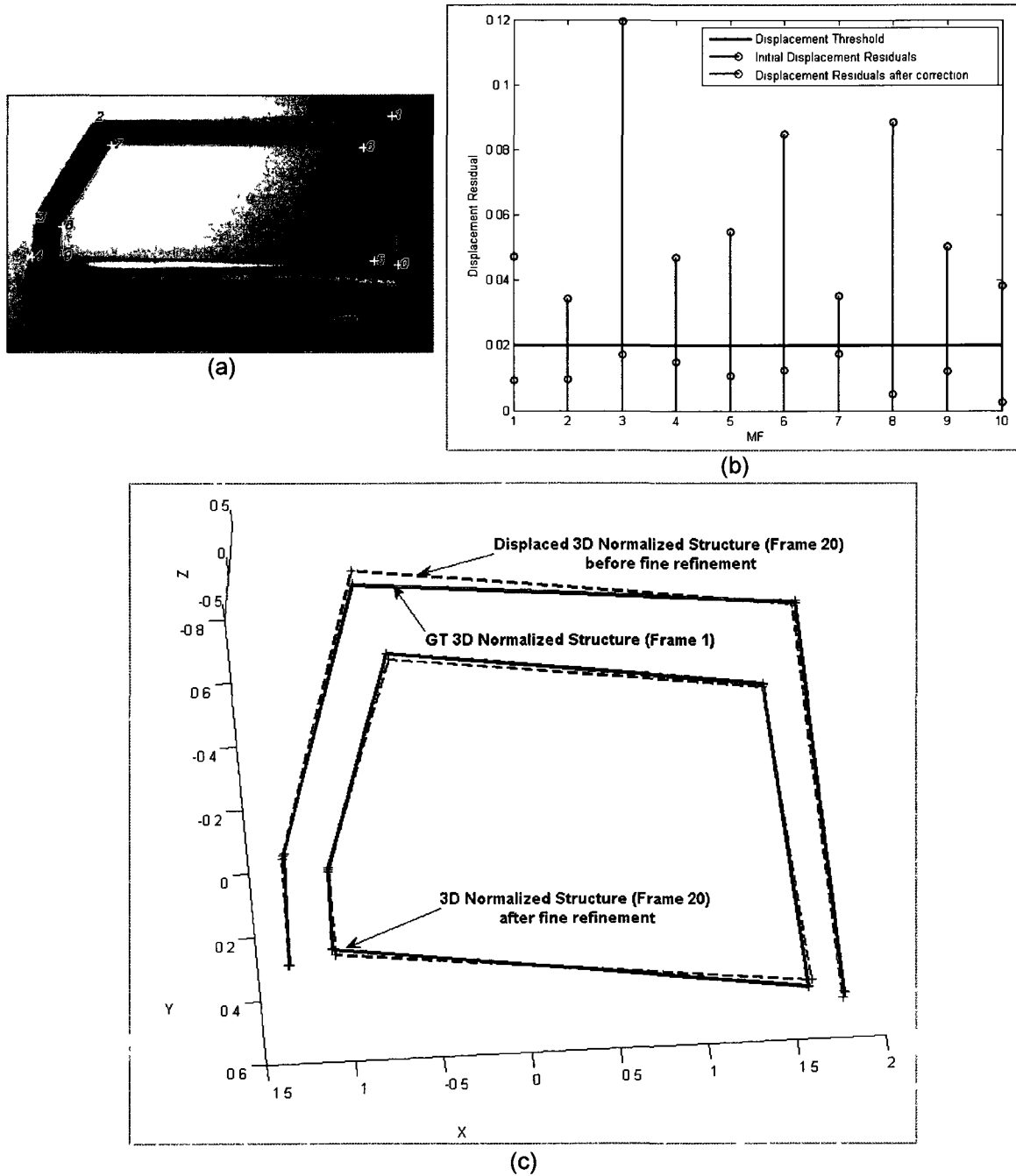


Fig. 5.8. (a) Segment from the 20th frame after the integration of the fine-level supervisory layer, (b) displacement residuals of the MFs' normalized structure in the 20th frame before/after the fine-level supervision, (c) 3D structure of the MFs' area in S^N for the GT and 20th frame before/after the fine-level supervision.

In the following paragraphs, a special case will be analyzed in which the extracted set of MFs having a displacement residual smaller than T_δ , has either a cardinal inferior to three, or the extracted MFs form a degenerate configuration. Under these conditions, as illustrated in Fig. 5.7b, the MF with the smallest displacement

residual is extracted, and based on the data stored in the buffer of relative distances, the rest of the MFs (a total of $N_{MF}-1$) are replenished in both views.

Then Sampson's correction is applied with the purpose of further refining these correspondences, which will be used in the 3D reconstruction process that relies on the linear triangulation procedure, which was discussed in Section 5.2. Upon the reconstruction, the resulting 3D structure of the MFs' area follows the same procedure including the transformation to the normalized space, the extraction of the rotation component and the computation of the new displacement residuals, as shown in Fig. 5.7b. Moreover, the decision block associated to this new set of data is marked in blue with black discontinuous line border, in order to better depict the flow (also in black discontinuous line) of triggered actions.

In the case in which the set of passing MFs satisfies the cardinality and non-colinearity conditions, the same course of actions is followed as in the first execution of the algorithm. Thus, the (R,T) pair between the current and GT 3D structures is estimated and used to correct the drifted MFs, which are then back-projected onto the image plane. Finally, the MFs' matches in the image planes are used to re-update the buffers of 2D distances and relative distances.

Although in almost 95% of the experimented cases (with different orientations of the sled system with respect to the stereoscopic sensor), the set of MFs verified the cardinality and non-colinearity tests, there might also be special situations in which these two conditions are not satisfied for the second time (i.e. the double decision block, marked in blue in Fig. 5.7b, returns "No" for two times, even after the refinement based on the MF exhibiting the smaller displacement residual and the data stored in the buffer of relative distances). To better support this particular case, Fig. 5.9a illustrates the zoomed patches around MF_1 , in the GT and the 17th frame grabbed by CamL (with an experimental setup similar to the one discussed in Section 4.3.2), in which the number of MFs classified as drifted was eight for the second time, resulting in an "N" flow triggered by the doubled decision block shown in Fig. 5.7b. By inspecting Fig. 5.9a, it can be noticed that MF_1 is affected by a considerable amount of drift, since in the 17th frame it is located above the ground-truth position, "trapped" in a noise pattern caused by the change of the light distribution in the scene during the movement of the car door.

In these rare cases, the fine correction of the MFs builds upon a variance normalized correlation (VNC)-based matching technique coupled with the methodology of block-truncation coding [16] used in image compression. This method, which was

formulated by Darrell [40], was also tested by Vincent [12], whose results demonstrated its superiority when compared to different attribute-based matching approaches relying on corner shape similarities or their associated curvature angles [38, 39].

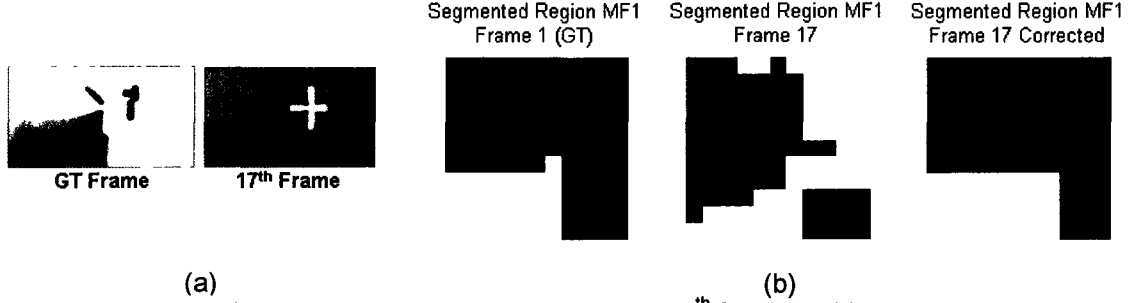


Fig. 5.9. (a) Zoomed patches around MF_1 in the GT and 17th frame grabbed by CamL during the motion cycle, (b) binary patches corresponding to the segmented 11x11 windows around MF_1 in the GT and 17th frame grabbed by CamL.

Specifically, 11x11 windows are centered on the corresponding MFs (between the GT and the current frame) that were labeled as drifted for the second time. Then the average intensity value is computed in these neighborhoods and the pixels above the mean level are assigned a value "0", whereas the other ones obtain a value "1". Moreover, the foreground [16] area consists of the region containing the MF. The first two sub-plots of Fig. 5.9b represent the obtained binary images associated to the neighborhoods belonging to MF_1 in the two considered frames. By computing the Hamming distance between these sub-images a result of $d_{\text{Hamming}}^{MF_1-\text{initial}} = 44$ pixels is obtained.

In order to correct the location of the faulty MFs, use is made of the VNC [12, 42] between 7x7 pixels neighborhoods centered on the MFs in the two analyzed frames. Thus, the VNC score is computed for the pairs containing the GT MFs and all the 49 locations in the selected neighborhoods belonging to the 17th frame. Additionally, in order to provide insensitivity to the background, a null weight was applied to its associated pixels in the 7x7 windows, when computing the VNC. Then, if the maximum VNC score in the 7x7 neighborhoods is greater than 0.8, the MF is corrected with the feature point exerting the highest VNC rate. On the contrary, a VNC score inferior to the same threshold is definitely correlated with occlusion effects, specular reflections or the temporary appearance of people/objects in the scene, which strongly affect the intensity distribution in the extracted MFs' patches. In these cases, the drifted MFs are corrected based on the relative position with respect to the MF exerting the smallest displacement residual. Once again, use is made of the buffer of relative distances characterizing the topological structure of the MFs in the image plane.

The last sub-plot of Fig. 5.9b represents the binary image resulting from the segmentation performed around the corrected MF_1 , case in which its Hamming distance with the ground-truth MF is $d_{Hamming}^{MF_1-final} = 4$ pixels. Once the entire set of drifted MFs is corrected, the buffers of 2D distances and relative distances are re-populated such that they can better characterize the current distribution of MFs in the image planes.

Finally, Fig. 5.10 illustrates four segments from the processed frames grabbed by CamL during the tracking sequence, in the cases in which the manipulator robot (Fig. 5.10a) or people (Fig. 5.10b and 5.10c), or both the manipulator robot and people (Fig. 5.10d) are present in the scene, when tracking the car door (Fig. 5.10a and 5.10b) or the fender (Fig. 5.10c and 5.10d).

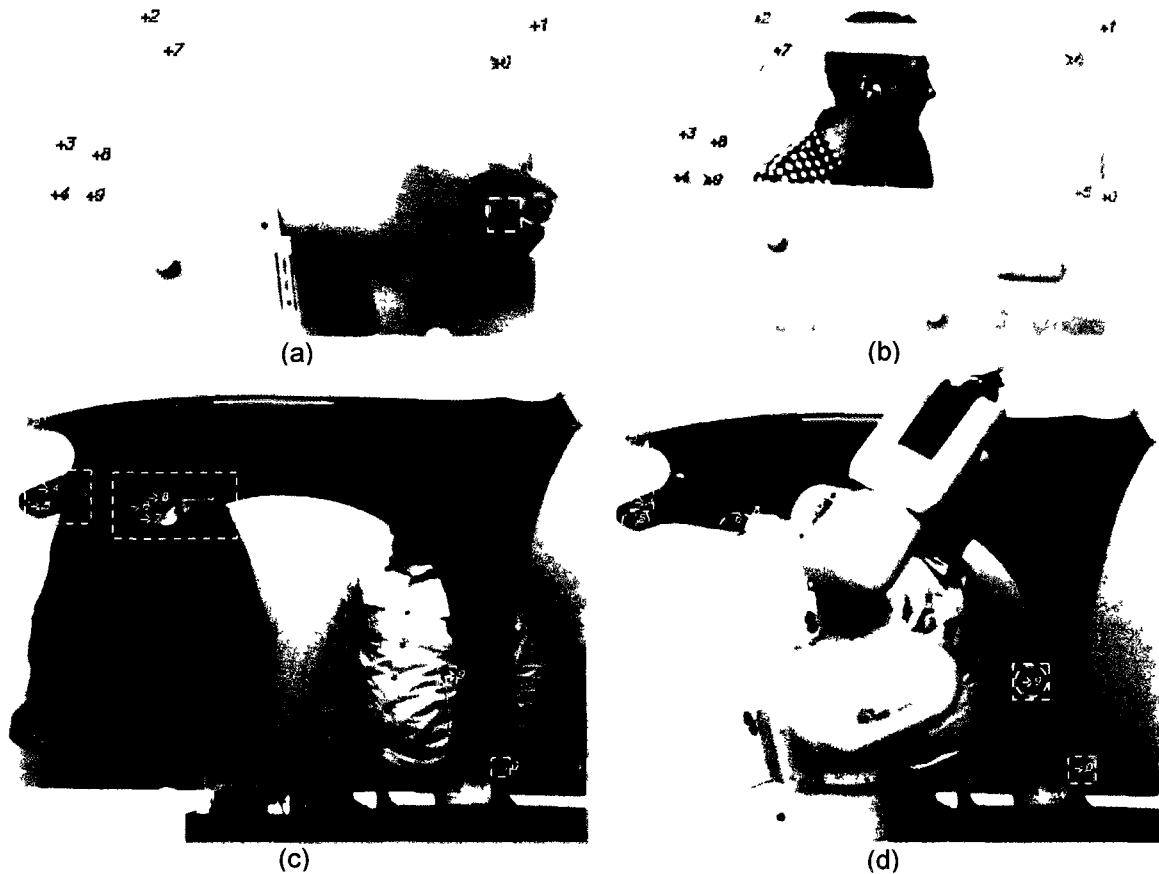


Fig 5.10. Processed frame segments grabbed by CamL during the tracking sequence, in the cases in which (a) the manipulator robot or, (b, c) people, are present in the inspected scene, (d) the manipulator robot and people are present in the inspected view.

In order to highlight the impact of the supervisory system, the MFs (or group of MFs) corrected or replenished by the coarse-level supervisory layer are tagged with a rectangle, while those refined with the fine-level supervisory sub-system are encircled.

Additionally, MF_6 in Fig. 5.10b, as well as MF_3 and MF_9 in Fig. 5.10d, are refined by both supervisory layers. By inspecting Fig. 5.10 it can be noticed that all the refined MFs point to the regions highlighted in Fig. 4.1, independently of the increase in the complexity level of the monitored scene, or the selected type of panel.

5.4.3. Impact of the Supervisory Layer in MFs Detection

The supervisory layer also monitors the MFs detection block, specifically the approximate MF selection process and the re-initialization procedure, which have been detailed in Section 4.2.1 and 4.2.2.

5.4.3.1. Impact of the Supervisory Layer in the Approximate MFs Selection

As part of the approximate MFs selection procedure, the supervisory system monitors the correspondences computed with the pyramidal LK tracker [20, 22] between the two initialization frames. Therefore, the search for a winning MFs pair, illustrated in Fig. 5.3, is performed by analyzing the set of correspondences given by the pyramidal LK tracker, by relying on the buffers of 2D distances and relative distances populated with the MFs' data in the CamL's initialization frame. The use of CamL's 2D topological structure buffers in the proposed coarse-level validation gate is feasible given the characteristics of the acquisition system, in which the vision sensors are approximately parallel.

Nevertheless, a threshold of ± 4 pixels is selected for the first two tests of the validation gate in Fig. 5.3, while the ± 2 pixels bound is set for the last test. As a result, the winning MFs pair is found in CamR, and all the other disparities are validated according to the buffer of relative distances in CamL's initialization frame, and the obtained distribution and magnitude of the disparities associated to the winning MFs.

In the case of the fender model, Fig. 5.11 illustrates the disparity vectors computed with the pyramidal LK tracker and represented in the initialization frame acquired by CamR. By inspecting Fig. 5.11, it can be noticed that the disparity vector associated to MF_2 (as indexed in Fig. 4.1b) could not be found, whereas the disparity vector related to MF_9 points to a wrong location. The regions of the two MFs (MF_2 and MF_9) are surrounded by light red ellipses. Additionally, it can also be observed that the orientation of the disparity vector related to MF_9 is different when compared to the other vectors. More specifically, the disparity associated to the first winning MF is applied to all the MFs belonging to CamL's initialization frame, in order to form a matrix of estimated

MFs' correspondences defined in CamR's initialization frame. In the case of the disparity vectors, shown in Fig. 5.11, the winning pair is (MF_1, MF_3) and the location of the correspondence related to MF_2 , which is not found by the pyramidal LK tracker, is estimated according to the disparity vector associated to MF_1 . Then a 7x7 pixels window is centered on all the MFs computed by the pyramidal LK tracker in CamR's frame and in the cases in which the estimated MFs lie within these extracted patches, their location is refined by the data given by the tracker.

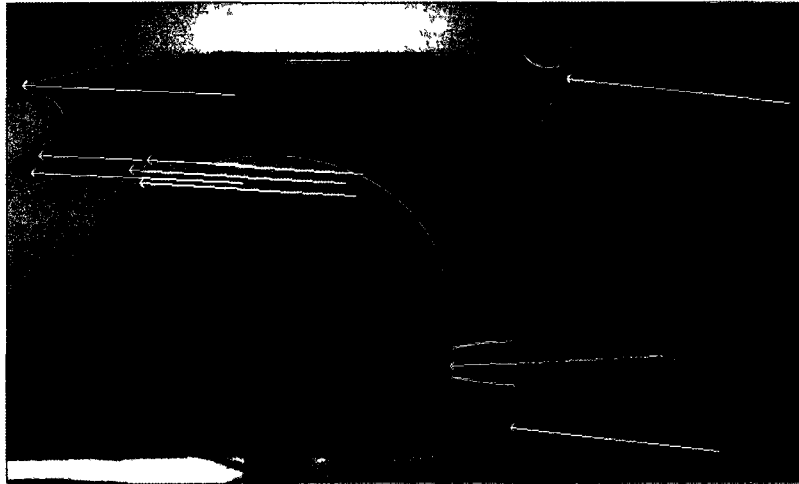


Fig. 5.11. Disparity vectors associated to fenders' MFs in CamR's initialization frame.

In the cases in which the search is not successful, or the pyramidal LK tracker is not able to compute all the MFs' disparity vectors or the validation gate applied on the computed matches (as discussed in Section 4.2.1) discards some of the motion vectors, the set of MFs is refined based on the data stored in the matrix of estimated correspondences. In this way, the coarse-level supervisory layer corrects the erroneous disparity vectors, and also replenishes the set of MFs such that the two initialization frames share the same number of MFs. The results for the fender's initialization frame, as corrected by the coarse-level supervisory layer are illustrated in Fig. 5.12. Here, the winning MFs are surrounded by squares with blue borders, whereas the estimated disparity vector for MF_2 is drawn in green, and the corrected correspondence vector for MF_9 is represented in red. Thus, from Fig. 5.12 it can be noticed that all the disparity vectors in CamR's initialization frame point to their associated MFs, as displayed in Fig. 4.1b.

Subsequently, the obtained MFs in CamR's initialization frame are refined by relying on the selective Shi and Tomasi corner detector [6], as described in Section 4.2.1. The coarse-level validation gate is applied before the refinement with the Shi and

Tomasi corner detector [6] in order to provide consistent information regarding the ROI (containing the MFs' area) extraction in CamR's initialization frame.

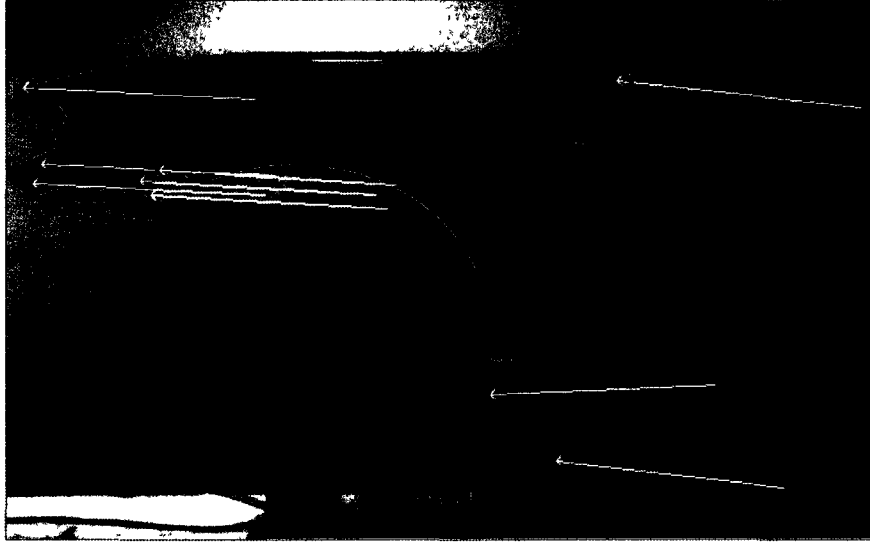


Fig. 5.12. Disparity vectors associated to fenders' MFs in CamR's initialization frame after the coarse-level supervision.

Finally, the implications of the fine-level supervisory system in the approximate MFs selection process is related to the application of Sampson's first-order correction to the refined set of matches, such that they consistently validate the epipolar constraint. As opposed to the conclusions of Hartley and Zisserman [45], and similar to Torr's statements [49], experimentation showed that Sampson's first-order correction can also correct drifts greater than one pixel.

5.4.3.2. Impact of the Supervisory Layer in the MFs Re-initialization Process

The re-initialization of the MFs' set builds upon the registration of the initialization frames with the images acquired by the stereoscopic sensor (re-initialization frames), as triggered by the object detection module, described in Section 4.2.4. This image registration technique relies on the pyramidal LK tracker [20, 22] and the Shi and Tomasi corner detector [6].

As for the refinements of the coarse-level supervisory system on the approximate MFs detection process, discussed in the previous section, the first monitoring stage of the image registration process is applied on the frames acquired by CamL. As a result, the topological structure buffers related to the distribution of the MFs in CamL's initialization frame are used within the same coarse validation gate, highlighted in Fig. 5.3, and with the thresholds mentioned in the previous section. Once the winning pair

has been found, the motion vectors returned by the pyramidal LK tracker are validated, based on the magnitude of the optical flow vector associated to the first winning MF and the data stored in the initialization folder. As a result, at the end of this stage, the set of motion vectors linking the initialization and re-initialization frames are refined and replenished such that the sets of MFs share the same cardinal in the two images. Subsequently, the Shi and Tomasi corner detector [6] is applied on the ROI containing the MFs' area and the corners extracted in the MFs' patches are used to further refine the motion vectors, as described in Section 4.2.2. Then, the newly obtained MFs are used to populate the topological structure buffers associated to the currently tracked object.

Once the refinement process has finished with CamL's frames, the coarse-level monitoring phase is also applied on the images acquired by CamR. By taking into account the possible slight changes in the pose of the rigid body with respect to the initialization pose, and the appearance of the object in the scene, the coarse-level supervision gate for the registration process in CamR's frame makes use of the topological structure buffers populated with the refined MFs in CamL's re-initialization frame. This strategy is supported by the spatial distribution of the cameras, according to which CamL's view of a slightly displaced object (with respect to the initialization pose) is more similar to CamR's view of the same object, rather than the appearance of the object in CamR's initialization frame. However, CamR's initialization frame is still used in the registration process with the current re-initialization image.

The coarse-level monitoring process of CamR's registration results is very similar to the approach presented in the previous section. Therefore, the locations of the MFs in CamR's re-initialization frames are analyzed with the purpose of finding the winning MFs pair. As already discussed, the buffer of 2D distances and relative distances populated with the refined results of CamL's re-initialization frame are used to guide this process, along with the same thresholds as in the refinement of CamL's registration procedure. Once the MFs have been corrected and replenished, a further refinement is performed by applying the Shi and Tomasi corner detector and searching for corners in the MFs' patches.

Nevertheless, the influence of the fine-level supervisory system in the MFs' re-initialization process is linked to the application of Sampson's first-order correction on the set of refined matches. Since this correction affects the location of the MFs in both views, the 2D topological structure buffers need to be re-updated.

5.5. A Metrology System for Validating the Pose and Motion Estimations

One important limitation with the current solutions for robotic interaction with moving objects [30, 64, 65, 67] is the lack of robust and objective methods for empirically evaluating the pose and motion estimations of the rigid body. In order to solve this problem, Chang *et al.* [66] used a 6 DOF NIST ("National Institute of Standards and Technology") laser tracker for the inter-calibration of multiple systems in the robotic cell and the validation of the visual servoing data. However, as mentioned in Section 2.3, the difficulty of calibrating the laser tracker with the visual servoing system preempted the establishment of a full set of rigid transformations between these physical entities. As a result, the final comparison was performed on the measured speed of the rigid body in each of its separate coordinates.

A metrology system for evaluating the pose and motion estimations was designed for the setup in Fig. 3.7a which uses the car door model illustrated in Fig. 3.1b. The general objective was to record the trajectory of the mobile part during motion, by tracing a line over a 156x140cm grid, composed of 1x1cm squares, that was fixed on the floor [72]. A third camera was attached on the mobile robot and synchronized with the calibrated stereo-vision sensor. This supplementary vision sensor, illustrated in Fig. 5.13a, provided information about the timing of the movement by detecting the position of a pencil marker over the grid during the recorded sequence.

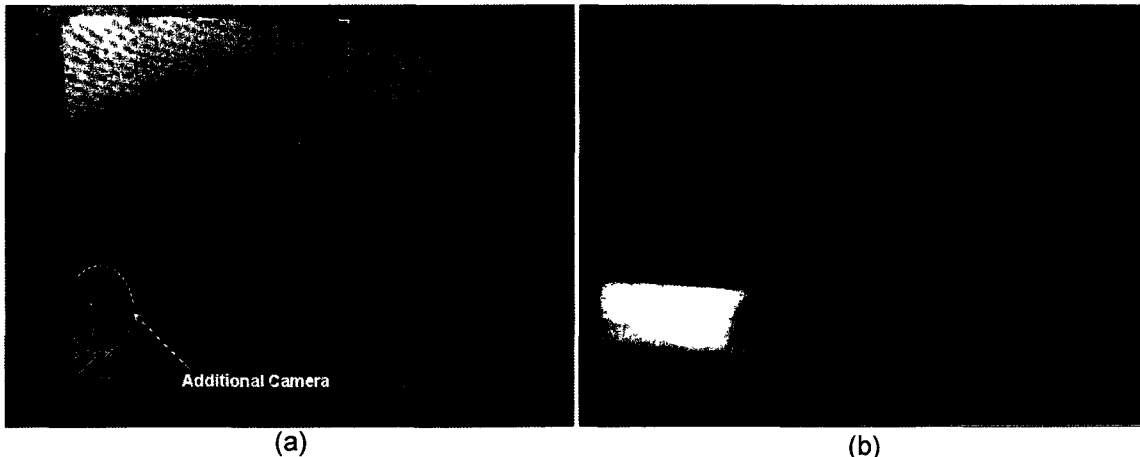


Fig. 5.13. Metrology system for validating the pose and motion estimations: (a) grid with reference frame, markers and additional camera for validation, (b) segment of a frame extracted by the additional camera.

The pencil was attached to the base of the mobile robot with a bracket and a spring system, as shown in Fig. 5.13b, that provided the marker with the sensitivity to

react to the changes in orientation of the mobile robot during the movement. Moreover, Fig. 5.13b illustrates a segment of a frame grabbed by the additional camera during the recording.

In this first prototype of the pose and motion estimation system [72], the videos were recorded offline, whereas the rigid body, manually driven by a remote control, moved with slightly variable speed. The frame extraction rate was also imposed by the processing of the ground-truth data recorded by the additional camera. The frame extraction process was triggered every time the pencil intersected with successive grid lines separated by 2cm. Thus, the average frame extraction frequency $\bar{f}_{\text{extr}}$ was 1.5Hz, and a total number of $N_{\text{frames}}=48$ extracted frames were employed in the processing sequence. The total displacement of the mock-up car door was about 100cm throughout the entire sequence, whereas the duration of the entire processed video sequence was about 40s.

In the testing stage of this pose and motion estimation system, only the rigid body structure shown in Fig. 3.1b appeared in the view of the stereoscopic system represented in Fig. 3.7. Additionally, the rigid body exhibited relatively straight and smooth motion, mainly a translation along the X axis in the first part, coupled with a minor translation along the Y axis in the second part, as defined with respect to the reference frame of the stereoscopic system, CamR, shown in Fig. 3.7a. When compared to the complete pose and motion estimation solution shown in Fig. 3.13, the early version did not include the supervisory system which monitors the object detection, feature extraction, matching and tracking processes.

The analysis of the reconstructed 3D data was based on a set of two quality tests. The first quality measure was related to the comparison between the estimated MFs results and the knowledge available about the geometrical structure of the mock-up door. In the case of the mock-up car door, illustrated in Fig. 3.1b, the MFs were represented by the four corners of the door window and the two corners of the door knob, as shown in Fig. 5.14a, along with their index.

The reference parameters were accurately measured in the real world and compared to the corresponding computed values based on the 3D positions of the MFs. The set of parameters, shown in Fig. 5.14b, and related to the rectangle representing the door window, consisted of the widths between MF_2 and MF_3 (width_1), and MF_4 and MF_5 (width_2), respectively, followed by the heights between MF_3 and MF_4 (height_1), and MF_2

and MF_5 ($height_2$), respectively. As well, the lateral size of the door knob (between MF_0 and MF_1) was monitored.

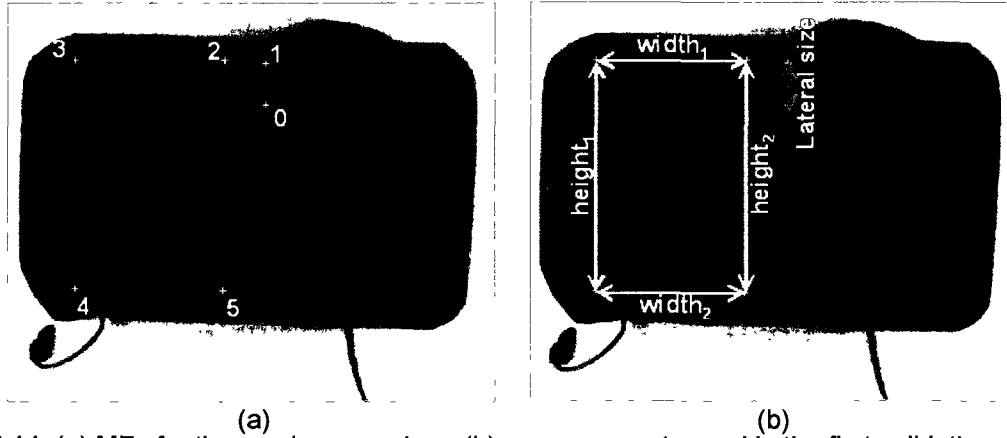


Fig. 5.14. (a) MFs for the mock-up car door, (b) measurements used in the first validation phase.

The absolute relative error between the real and the estimated data was computed over the entire processed sequence. The values reported in Table 5.2, represent the absolute relative errors computed over the sequence, with and without the monitoring stage that re-extracted the corners.

Table 5.2. Quality measure I – ground truth data comparison

Corners re-extraction	Error with respect to ground truth values				
	Width ₁ Error (cm)	Width ₂ Error (cm)	Height ₁ Error (cm)	Height ₂ Error (cm)	Size of knob Error (cm)
Without	0.82	0.93	1.55	1.7	0.79
With	0.76	0.9	1.63	1.73	0.42

Since the robot or people were not present in the scene during this stage of experimentation, the MFs monitoring stage built upon the Shi and Tomasi corner detector with sub-pixel accuracy [6, 21], which was re-applied every 5 frames. Therefore the returned corners found in 5x5 pixels windows centered on the MFs were used in the refinement process. From Table 5.2, it can be observed that the parameters, in which MF_4 and MF_5 were involved ($height_1$, $height_2$, $width_2$), were more erroneous than the measurements which included the other MFs ($width_1$ and the knob segment). This was caused by the relatively higher back-projection error, in the vertical direction, obtained for the latter MFs. Moreover, considering the group containing the first four MFs, the mean absolute error was inferior to 1cm. Thus, acceptable accuracy was achieved from the perspective of robotic marking of deformed areas over an automotive part, where the

goal is to highlight the region of the deformation defects, which will be repaired by factory associates. Interestingly, there was no consistent improvement observed in the data when the MFs were monitored through corner re-extraction, as only the rectangle widths and the lateral size of the knob tended to get slightly closer to their real values.

The second quality measure was related to the comparison of the pose and motion estimations with the ground-truth data recorded by tracing the trajectory of the rigid body over the grid fixed on the floor. In order to compare the two different data sets, one that was measured with respect to the world reference frame (CamR), and the other one relative to the grid (O_G , as shown in Fig. 5.13a), a two-step transformation procedure was employed. First, ten corner-shaped markers placed over the grid, and displayed in Fig. 5.13a, were recovered with the Shi and Tomasi corner detector [6, 21]. The method developed by Arun *et al.* [57] was used to compute the transformation matrix between the world reference frame and the grid reference frame. With the use of the estimated transformation matrix, the 3D locations of the MFs were then expressed with respect to the grid. In the second step, the displacements in the X , Y , Z directions (relative to O_G) between the MFs and the tip of the pencil tracing the trajectory were measured manually to allow a comparison of trajectories defined with respect to the grid reference frame.

The validation procedure included two comparisons for validating the accuracy of the pose and motion estimations. Figure 5.15 illustrates the real and estimated 3D trajectories of the rigid body throughout the sequence. For computing the estimated trajectory, the transformation procedure was applied on the first MF since it exhibited the smallest back-projection error during the entire motion sequence. For the initial part of the processed video, where the rigid body was smoothly translating along the X axis, the two trajectories were very close to each-other. In the second part (for $X > 40\text{cm}$), a slight divergence started to appear between the two motion paths, corresponding to the segment in which the rigid body also exhibited small displacements in the Y direction.

The maximum value of the obtained divergence was 3.5cm along the X axis, 2.8cm along the Y axis, and 2cm with respect to the Z axis. One of the most important causes of this error is related to the fact that the pencil which marked the grid was mounted on a spring which tended to slightly bias the trajectory when changing direction. The least-squares method used in computing the transformation matrix between the world reference frame and the grid reference frame, together with the manual

measurements made in the second step of the transformation procedure, also introduced a small amount of error in the system.

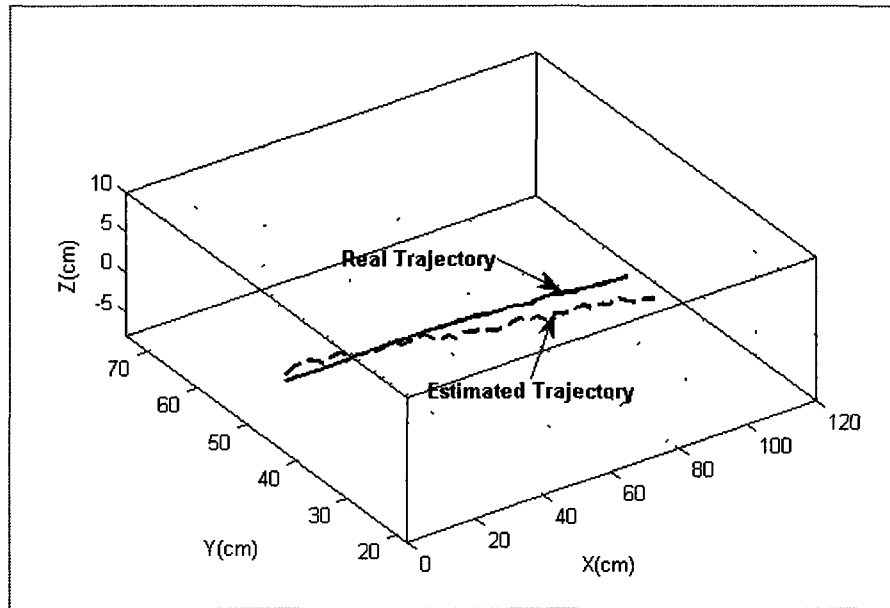


Fig. 5.15. Real and estimated 3D trajectories comparison for the rigid body throughout the motion sequence.

Nevertheless, the slight turning right effect, caused by the minor translations along Y (relative to O_G) present in the motion pattern of the rigid body in the second part of the experiment, had a noticeable effect on the accuracy of the second phase of the rigid transformation computation procedure. Therefore, the positions of the MFs with respect to the tip of the pencil were affected, due to the rudimentary spring system used for marking the grid. The latter effect explains the magnitude of the obtained divergence.

For the second comparison, the velocities along the X and Y axes were analyzed for the ground-truth and estimated data sets. Figure 5.16a shows that the real and estimated velocities along the X axis were very close to each other. Also, the rudimentary trajectory marking system was the cause of the spikes present in the last part of the sequence, in both Fig. 5.16a and 5.16b. Moreover, the velocity in the X direction was not constant as the mock-up door was manually moved with a mobile robot during experimentation, which resulted in slight variations around the mean value of 3cm/s.

The velocity along the Y axis was very small in the first part of the experiment and was mainly caused by the merged effect of noise, imperfect orthogonality of the stereo-cameras with the horizontal surface over which the motion occurred, and the small variations during the motion. Then, in the second part of the motion sequence, the

absolute value of the velocity along the Y axis increased, as the rigid body slightly turned right from the initial direction of motion.

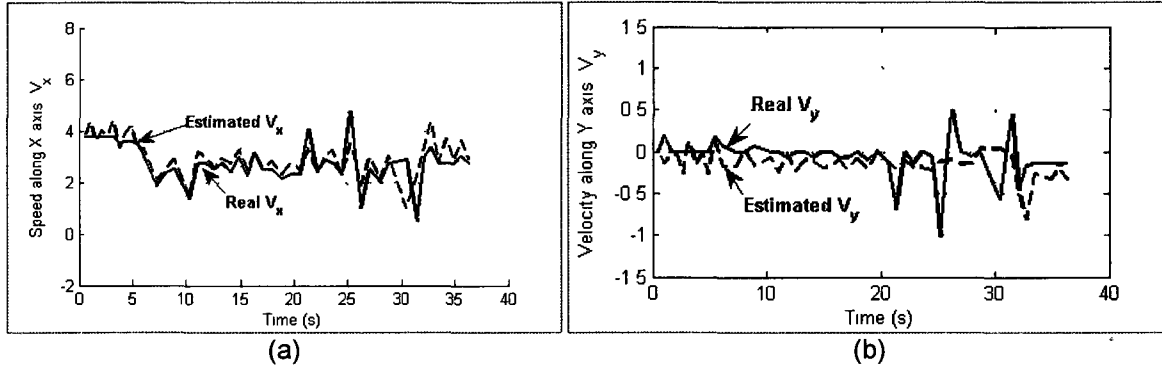


Fig. 5.16. Real and estimated velocities (cm/s) throughout the motion cycle, along: (a) the X axis, (b) the Y axis

Although the proposed validation system was not as highly accurate as the ground-truth data accumulation presented in [66], the two-step inter-calibration procedure mentioned above allowed a complete comparison in position and velocities, rather than differential motion as performed in [66]. As a result, the recorded trajectory was fairly accurate and provided a good basis of comparison. Nevertheless, it demonstrated the suitability of the early pose and motion estimation prototype, for general application in robotic marking of surface deformation regions, in the context of quality control in industrial manufacturing.

This initial metrology system was further developed for the final configuration that relies on a different positioning of the acquisition system, as shown in Fig. 3.8, and which is integrated in a complete robotic arm work cell for autonomous 3D deformation defects detection and marking, illustrated in Fig. 3.10. More details will be provided along with discussion of the validation of the pose and motion estimation using this infrastructure in Sections 6.3 and 6.4.

5.6. Chapter Summary

This chapter completes the description of the proposed pose and motion estimation solution, whose high-level diagram is illustrated in Fig. 3.13. The first part of this chapter detailed the methodologies of 3D reconstruction and motion estimation that are embedded in the proposed framework and play a fundamental role for the robotic navigation.

Subsequently, the innovative layer of the pose and motion estimation architecture, given by the supervisory level, was described in the second part of this

chapter. This additional block continuously monitors and validates the feature tracking and matching processes, which have the strongest impact on the accuracy of the pose and motion estimations. The two sub-systems embedded in the supervisory layer, that is the coarse-level and fine-level supervision, are developed from a software perspective. Therefore they do not require any increase in the complexity of the hardware architecture [30, 64, 65], or the mathematical formulation of the feature tracking [24, 26, 27, 36] and matching processes [39, 40, 41].

The merit of the proposed supervisory layer is related to the manner in which the knowledge regarding the topological structure of the MFs in 2D and 3D is successfully integrated in the validation and monitoring processes. As a result, in spite of the little amount of information available about MFs detected on a texture-less automotive body panel, the system offers a reliable supervision alternative to the challenging problem of estimating the pose and motion of the part even though the motion happens in a complex environment with other moving entities and sporadic occlusions. The supervisory layer also provides a higher level of flexibility and complexity for the motion patterns that the object can exhibit on the assembly line, when compared to the trivial setup under which the object performs only straight motion, with no changes in orientation. The proposed pose and motion estimation system is able to function properly even in cases where the assembly line contains curved sections, as often imposed by space limitations in factory settings. The robustness exhibited by the proposed system also supports its integration with different applications in which the motion generator might not be a conveyor belt.

Finally, the last part of this chapter introduced a rudimentary metrology system for validating the accuracy of the pose and motion estimations, which was designed alongside the first prototype developed. Experimental validation showed that sufficient accuracy can be obtained, in the context of robotic marking of deformations. This early experimental setup did not allow for a complete analysis of the overall performance of the autonomous 3D deformations detection and robotic marking station that was developed later on. However, the experimental validation of the final and complete pose and motion estimator will be further discussed in the next chapter, along with the final setting involving the robotic marking of surface deformation defects for quality control in industrial manufacturing.

Chapter 6. Robotic Integration and Experimental Validation

6.1. Introduction

This chapter details the incorporation of the pose and motion estimation system within the robotic station for marking undesired surface deformation defects over the surface of automotive body panels, without human intervention beyond initialization. Section 6.2 describes the complete experimental cell for autonomous detection and marking of deformation defects, developed in partnership with Yogeswaran [80], and initially introduced in Section 3.5. The two major components needed for the integration of the robotic marking system within the proposed quality control application are provided by two inter-calibrations. The former relates the stereoscopic sensor used for pose and motion estimation and the robotic base. The latter links the defects detection system with the acquisition sensor used for pose and motion estimation. The second phase of calibration provides the necessary generality level for extrapolating the proposed solution to the case in which the defects detection system and the panel's pose and motion estimator, along with the robot marking stage, are positioned in different stations. The incorporation of the two inter-calibrations in the procedure of robotic pointing of surface deformation defects is described in the last part of Section 6.2.

Section 6.3 introduces the first prototype for the autonomous marking of deformation defects in a static environment. Therefore, the accuracy of the robotic stamping of deformations is monitored at different locations and orientations of the automotive panel on the assembly line, which also allows for the validation of the pose and motion estimations, throughout the motion cycles.

Finally, Section 6.4 describes the integration of the pose and motion estimator and the two inter-calibrations within a more comprehensive system where the robot marks defects on moving panels. Specifically, the first part of Section 6.4 is dedicated to the analysis of the robotic gripper's attitude with respect to the moving panel throughout the entire monitored motion cycle. This analysis represents the preliminary phase for the development of the final prototype for on-line robotic interaction, which relies on a spray gun end-effector. In this way the marking of deformations can be successfully performed, without the need for the manipulator robot to physically touch the moving panel.

6.2. Integrated Autonomous System for Marking of Surface Deformation Defects

The description of the complete robotic work cell for the combined application of 3D deformation defects detection [80] and marking on automotive body panels was introduced in Section 3.5, and will be briefly discussed here. First, a 3D surface model of the automotive part, as shown in Fig. 6.1a, is acquired by the structured light sensor (SLS) which is positioned in the same work station as the stereoscopic sensor used for the pose and motion estimation (SSPME) of the rigid body. As it can be noticed from Fig. 6.1b, three deformations (dings) have been attached to the car door model. These dings vary from a maximum size of 2.9cmx2.2cmx1.4cm (Ding₁), to a minimum size of 1.6cmx1.3cmx1cm (Ding₃). The coloured and dense 3D model acquired with the SLS contains 32040 points and constitutes the input to the surface deformation defects detection station, developed by Yogeswaran [80], as illustrated in Fig. 3.9.

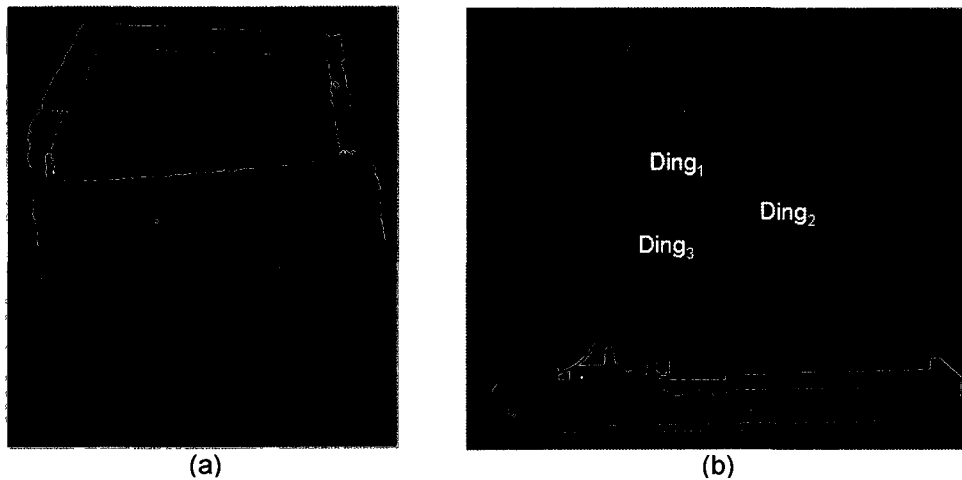


Fig. 6.1. (a) Textured point set surface map of the scanned car door, (b) car door model with attached dings.

Taking into account the blocks embedded in the system for 3D deformation defects detection, as shown in Fig. 3.9, Fig. 6.2a illustrates the results obtained by Yogeswaran [80] after applying the feature extraction algorithm, whose objective is to detect local features which contain sharp variations of their normal orientation, indicating potential deformation regions. A feature grouping module performs the clustering of the collections of local feature pieces, in order to properly represent larger deformations, such that information about size, shape and other characteristics can be determined, to aid in the classification of deformations. The patches in Fig. 6.2b are related to the grouped deformation regions belonging to the car door panel. The objective of the

feature classification sub-system is to extract feature groups that exhibit the characteristics of the deformation features to be detected, such as dings, dents or extra spots of welding. Figure 6.2c illustrates the regions selected by the feature classification module. For each of the three identified dings, their associated 3D coordinates, expressed with respect to the reference frame of the SLS, attached to its left camera, $CamL_{SL}$, are provided to the robotic marking station.

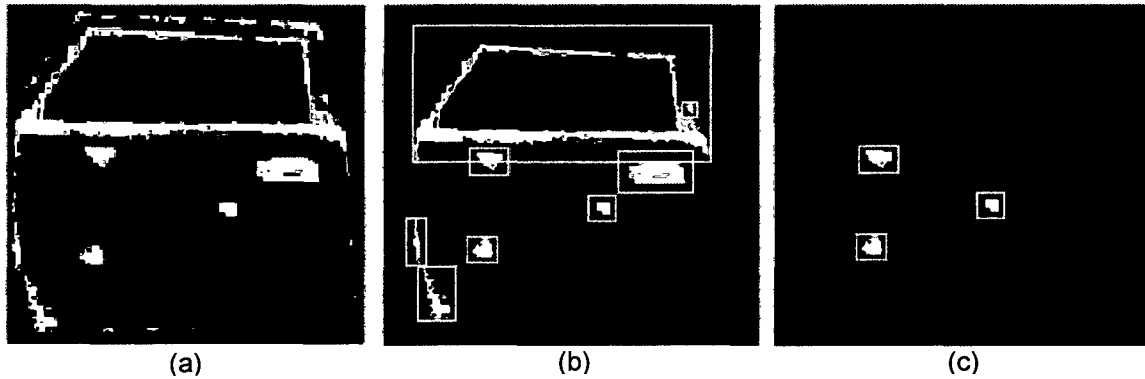


Fig. 6.2. Results of the 3D surface deformation defects detection system: (a) feature extraction, (b) feature grouping, and (c) feature classification.

The current configuration of the detection system, which runs off-line, allows the detection of deformations at a minimum size of approximately $1\text{cm} \times 1\text{cm} \times 1\text{cm}$, due to the relatively low resolution of the SLS. Nevertheless, using a 3D scanner of higher resolution allows detecting finer deformations, without any change to the detection algorithm [73, 75, 80].

As specified in Section 3.5, in order for the robot to perform the marking operation on the automotive panel, two inter-calibrations are needed. The first one relates the SSPME with the robotic arm, whose reference frame, O_B , is attached to its base, as shown in Fig. 3.11a. The second inter-calibration involves the SLS and the SSPME, whose reference frame is given by $CamR$. With the help of the two inter-calibrations, which will be discussed in the next two sections, the positions of the surface deformations can be transferred into the robot's reference frame, to guide the marking task.

6.2.1. Inter-calibration between the Stereo-vision Sensor and the Robot Base

In order to inter-calibrate the SSPME and the robot's base, a checkerboard pattern, which can be attached to the robot's gripper, was designed. The calibration pattern, composed of $3\text{cm} \times 3\text{cm}$ black and white squares, is mounted in colinearity with

the reference frame of the tool, O_T , as shown in Fig. 6.3. Given that the only transformation between the tool of the robot and the calibration pattern is a constant translation along the Z axis of the tool's reference frame (Fig. 3.11a), the location of the checkerboard corners can be uniquely identified with respect to O_T , and eventually, to the robot base, O_B , knowing the robot's kinematics. As a result, independently of the pose of the robot with respect to the stereo-vision system, the location of the chessboard features relative to the robot's tool remains constant.

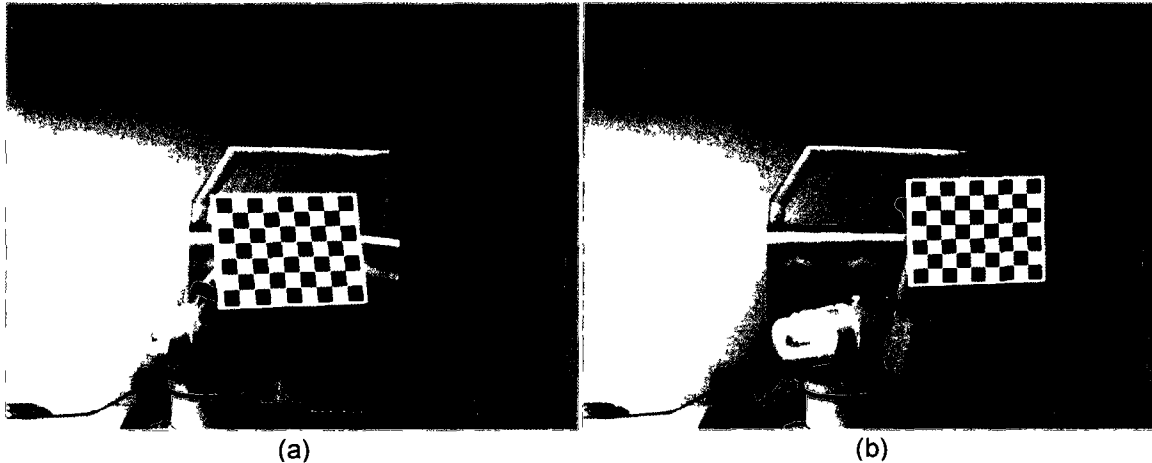


Fig. 6.3. (a), (b) Images taken by CamR during the inter-calibration procedure

To acquire a set of calibration feature points, the robot is successively driven to 15 different configurations, such that the region of the workspace containing the automotive panel over the visible section of the assembly line is covered. To exemplify, Fig. 6.3 illustrates two segments of the frames grabbed by CamR during the inter-calibration procedure. The block diagram of the inter-calibration process is shown in Fig. 6.4.

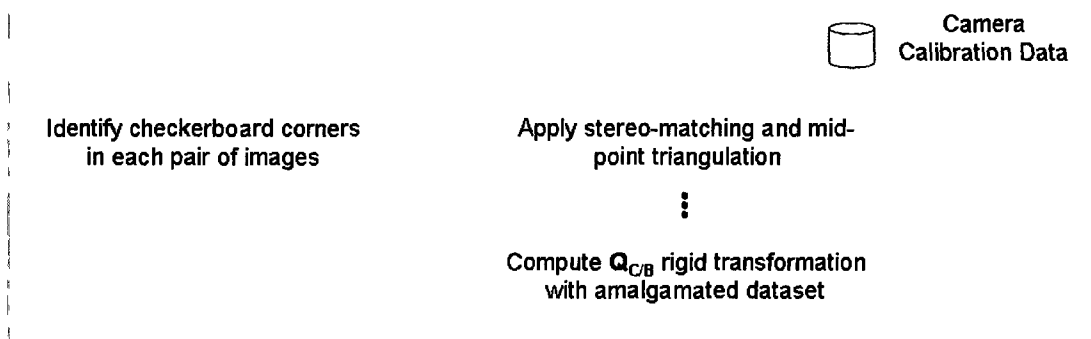


Fig. 6.4. Inter-calibration of CamR and robot's base procedure.

For each different robotic configuration, a synchronized set of images is acquired by both cameras, CamL and CamR. Use is made of OpenCV's functionality [21] for detecting checkerboard corners in both views. To illustrate this procedure, Fig. 6.5

shows two segments from a synchronized pair of frames grabbed by the stereoscopic sensor, on which the checkerboard corners identified with OpenCV [21] are identified. Since the location of the checkerboard corners is only approximate, the precision of the detected features is improved with the sub-pixel accuracy refinement method [21].

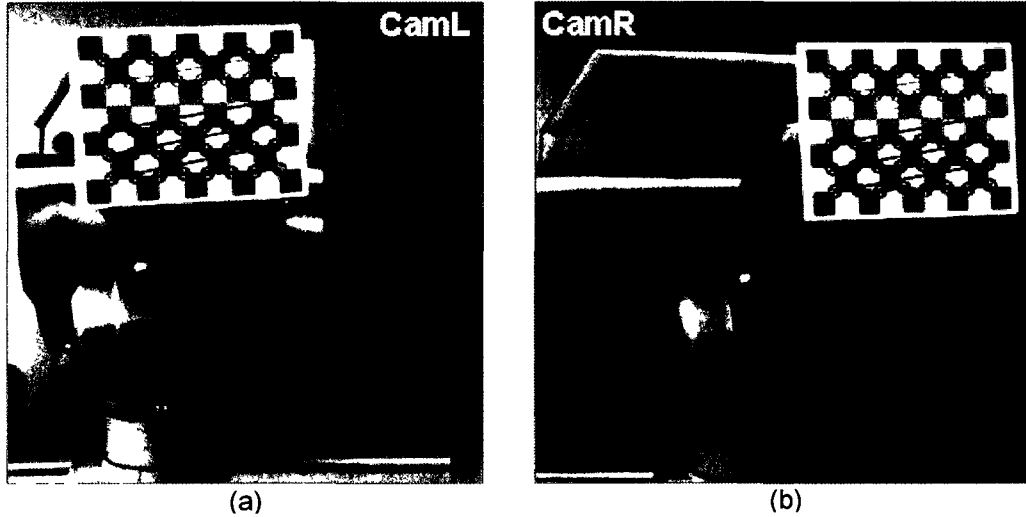


Fig. 6.5. Image segments, with identified checkerboard corners, from two synchronized frames grabbed by (a) CamL, and (b) CamR during the inter-calibration procedure.

Once the chessboard corners are refined in both stereo-views, the feature correspondence problem becomes straightforward, since the matrix structures associated to the corners' location are populated in the same order for both views. With the purpose of further improving the precision of the feature correspondences, Sampson's first-order correction [45], described in Section 5.4.2.1, is also applied twice on the initial set of matches, which results in a considerable minimization of the epipolar residuals, from an order of magnitude of 10^{-2} to 10^{-15} . The magnitude of the change in the checkerboard corners' location, after applying Sampson's first-order correction, is approximately 10^{-3} pixels, besides the considerable decrease in the values of the epipolar residuals.

Subsequently, the full calibration data of the stereo-vision sensor, together with the refined set of feature matches are used in the linear triangulation process, discussed in Section 5.2, in order to recover the 3D sparse structure of the checkerboard, with respect to CamR. Each time the checkerboard corners, belonging to a configuration in which the robot has been driven, are reconstructed, their corresponding 3D locations with respect to the robot's base are also computed, using the robot's forward kinematics. After the checkerboard corners are reconstructed in the two reference frames, CamR and O_B , the rigid transformation between these systems, $Q_{CamR/Base}$, can be computed

with the least-squares procedure proposed by Arun *et al.* [57]. After performing an analysis similar to that introduced in Section 5.3, which relies on the maximum of the absolute error and the root mean squared error (RMSE), a better accuracy was obtained for the case in which the rigid transformation matrix, $Q_{CamR/Base}$, was computed with the amalgamated set of data acquired after 15 successive 3D recoveries of the chessboard corners rather than using an averaging procedure over 15 different transformation matrices, estimated at every recorded location of the checkerboard corners.

In practice, the inter-calibration between the robot's base and the SSPME needs to be computed only once, that is when configuring the robotic marking system. As a result, the 4x4 rigid transformation matrix, $Q_{CamR/Base}$, is obtained, which embeds the rotation $R_{CamR/Base}$, and translation $T_{CamR/Base}$ between the CamR and O_B :

$$Q_{CamR/Base} = \begin{bmatrix} R_{CamR/Base} & T_{CamR/Base} \\ \tilde{0}_{1 \times 3} & 1 \end{bmatrix} \quad (6.1)$$

where $\tilde{0}_{1 \times 3}$ represents the null vector having one line and three columns. The matrix $Q_{CamR/Base}$ follows the convention according to which, a point $\tilde{P}_{O_B}$, defined in the robot's base reference frame, can be recovered from the point $\tilde{P}_{CamR}$, expressed with respect to CamR, according to the equation:

$$\tilde{P}_{O_B} = Q_{CamR/Base} \cdot \tilde{P}_{CamR} \quad (6.2)$$

in which both $\tilde{P}_{O_B}$, $\tilde{P}_{CamR}$ are expressed in homogenous coordinates.

6.2.2. Inter-calibration between the Stereoscopic Sensor and the 3D Surface Imager

For transferring the 3D locations of the detected deformation defects from the SLS reference frame to that of the SSPME, a second inter-calibration procedure needs to be performed. In one category of applications, advantage can be taken of the fact that the SLS is installed in the same work cell as the SSPME. Therefore, a similar inter-calibration approach, based on the checkerboard pattern can be applied for the calibration of these two systems [75]. In a more general scenario, as discussed in Section 3.5, the complete architecture for deformations detection and robotic marking can be divided in two different stations, the first one dealing with the 3D scanning and deformation detection and the second one performing robotic tracking and marking of deformation areas. In these conditions, the marking operation can be performed at a

subsequent stage from the defects detection procedure, resulting in substantial accelerations of the quality control process.

Under the general scenario, the previous target-based calibration procedure is not feasible anymore, since it will be impossible for the SLS and the SSPME to view the same calibration object at the same time. In addition to this, another difficulty is associated to the fact that there are high chances for the pose of the automotive part with respect to the SLS to exhibit considerable changes when compared to its attitude relative to the SSPME. These variations might originate from the 3D surface imager performing the scanning operation from a shorter distance to the object, in order to optimize the accuracy of the model, which is critical for identifying tiny deformations over automotive panels. Conversely, as discussed in Section 3.4, the SSPME needs to be positioned in the scene in a way that maximizes the duration over which the moving panel appears in both views.

In order to accommodate these constraints imposed by the installation of the SLS and the SSPME in distinct stations, an original solution is proposed that makes use of the MFs pre-selected over the structure of the automotive parts. Once again, use is made of the *a priori* selection of features as they prove to be the most stable points over the surface of the inspected panel. Therefore, the methodology proposed for the approximate MFs selection and refinement, introduced in Section 4.2.1, is extrapolated to the 3D scanning and deformations detection station, since the SLS also relies on a higher resolution and calibrated stereoscopic sensor.

During the process of 3D scanning the automotive panel remains stationary. The operator can manually pre-select the locations of the desired MFs, over the initialization frame grabbed by the left camera of the SLS, $CamL_{SL}$. Then, the locations of the MFs are further refined by applying the Shi and Tomasi corner detector [6], in the region of interest (ROI) containing the MFs' area, as explained in Section 4.2.1. Subsequently, the stereo-correspondence process is guided by the pyramidal implementation of the LK tracker [20, 22], which is reliable in the case of the SLS sensor, as well, since the two vision sensors are approximately parallel. Thus, in a similar fashion as for the supervisory layer involved in the approximate MFs detection process, as discussed in Section 5.4.3, the topological structure buffers related to the spatial distribution of the MFs in $CamL_{SL}$'s frame are used for monitoring and validating the accuracy of the disparity vectors returned by the pyramidal LK tracker. Subsequently, the MFs are refined with the Shi and Tomasi corner detector in $CamR_{SL}$'s frames, and the final set of

correspondences is enhanced by applying Sampson's first-order correction [45]. The refined location of the MFs in the initialization image grabbed by CamL_{SL} are illustrated in Fig. 6.6a, whereas Fig. 6.6b shows the MFs refined in the initialization frame acquired by CamR_{SL}. Also, in Fig. 6.6b are shown the epipolar lines corresponding to the MFs extracted in CamL_{SL}'s initialization frame.

Similar to the procedure described in Section 4.2.1, the approximate MFs selection in the frame acquired by CamL_{SL}, is performed only once when the system is configured to inspect a specific type of automotive panel. For all cases where a new panel of the same type starts its inspection cycle, the set of MFs is automatically re-initialized, as discussed in Section 4.2.2. In this way, the new panel can exhibit a slightly different pose with respect to the SLS without causing changes to the proposed MFs-based inter-calibration methodology.

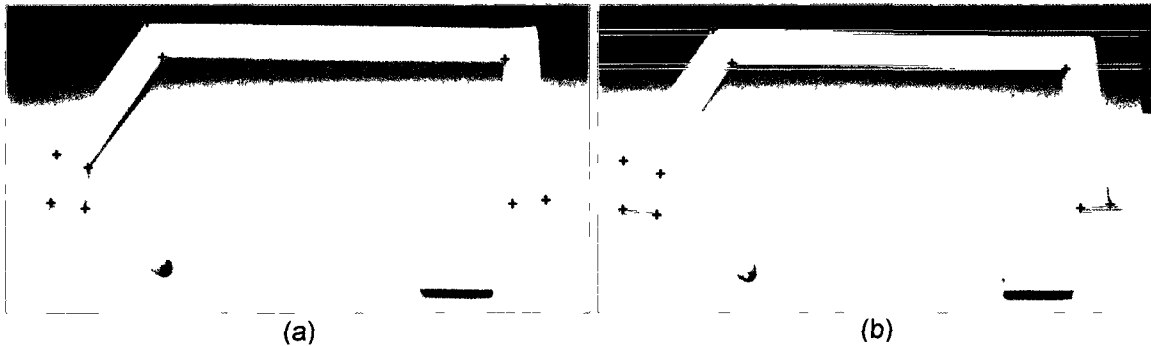


Fig. 6.6. (a) Refined MFs in the initialization frame grabbed by CamL_{SL}, (b) refined MFs and their associated epipolar lines in the frame acquired by CamR_{SL}.

The extrinsic calibration technique implemented by Boyer [81] for the same SLS, builds upon a checkerboard pattern approach, as proposed by Hartley and Zisserman [45]. However, the calibration accuracy obtained when applying the extrinsic calibration procedure, developed by Bériault [76], on the SSPME, is superior to the precision acquired by Boyer. The back-projection error, for both approaches, was computed by using the extrinsic and intrinsic calibration data to triangulate the points of the calibration target in the 3D space, and then project them back in the image plane [81]. Specifically, the distance, expressed in pixels, between the identified calibration points and their back-projected locations, was computed and averaged. In the case of the calibration method developed by Boyer [81], an average back-projection error of $\bar{e}_{bp}^{SLS} \cong 2.58$ pixels, and a maximum back-projection error of $M_{bp_error}^{SLS} \cong 4.31$ pixels were obtained. Conversely, the approach provided by Bériault [76], is characterized by an average back-projection

error of $\bar{e}_{bp}^{SSPME} \cong 0.15$ pixels and a maximum back-projection error of $M_{bp_error}^{SSPME} \cong 0.52$ pixels.

Thus, although the SLS embeds a stereoscopic sensor whose cameras have a resolution of 1392x1040 pixels, when compared to the 640x480 pixels resolution of the vision sensors composing the SSPME, the calibration performed on the SSPME is more precise. The accuracy associated to these two calibrations contributes to the precision obtained for the inter-calibration between the SLS and the SSPME, which will be reported in the next paragraphs.

As opposed to the inter-calibration performed between the SSPME and the robot base, the inter-calibration between $CamL_{SL}$ and $CamR$ is executed on-line, for every inspected object, once the MFs set is re-initialized in the views of the SSPME, after the completion of the deformations detection cycle for the specific automotive part. Only a limited number of MFs ($N_{MF}=10$, in the case of the car door and fender model) are used by the proposed inter-calibration, which is extremely fast (a few milliseconds). The 3D MFs' point cloud reconstructed in the SLS's reference frame ($CamL_{SL}$), together with the 3D MFs' set recovered at re-initialization with respect to $CamR$, are used as inputs to the least-squares rigid motion estimation approach proposed by Arun *et al.* [57], which was introduced in Section 5.3. Thus, the transformation matrix $Q_{CamL_{SL}/CamR}$ is computed and the 3D MFs, recovered in the SLS's reference frame, are transferred in SSPME's reference frame, for the analysis of the estimation error, as performed in Section 5.3. A MF defined in $CamL_{SL}$'s reference frame, $\tilde{MF}_i^{CamL_{SL}}$, $i = 0, \dots, N_{MF} - 1$ can be transferred in $CamR$'s reference frame, according to the equation:

$$\tilde{MF}_{i_{est}}^{CamR} = Q_{CamL_{SL}/CamR} \cdot \tilde{MF}_i^{CamL_{SL}} \quad (6.3)$$

in which $\tilde{MF}_i^{CamL_{SL}}$, $\tilde{MF}_{i_{est}}^{CamR}$, are expressed in homogenous coordinates. Figure 6.7 illustrates the 3D MFs' area, as reconstructed in $CamR$'s reference frame, at re-initialization, as well as the estimated MFs' region, computed with eq. (6.3). Additionally, Table 6.1 shows the results obtained for the two measures used in the error analysis of the rigid transformation estimation. As discussed in Section 5.3, these measures are related to the maximum absolute error and the root mean squared error, which characterize the differences between the reconstructed and estimated 3D MFs' region. One can observe that the maximum magnitude of the two error measures is obtained along the Z component, followed by a slightly lower error in the X component. The cause

of these displacements is three-fold. First, the 3D positions of the MFs are not extremely accurate, as they are affected by the precision of the calibration data, stereo correspondences and triangulation processes. Second, the considerable difference in the resolution of the cameras used for the SLS and the SSPME respectively also introduces a potential drift between the exact location of the MFs within the inner and outer frame of the door's opening. Finally, the procedure used for estimating the rigid transformation, $Q_{\text{CamL}_{\text{SL}}/\text{CamR}}$ builds upon a least-squares approach, thus actively contributing to the error measures shown in Table 6.1. Therefore, the residuals shown in Table 6.1 can only be regarded as approximations of the real displacements between the two 3D rigid structures.

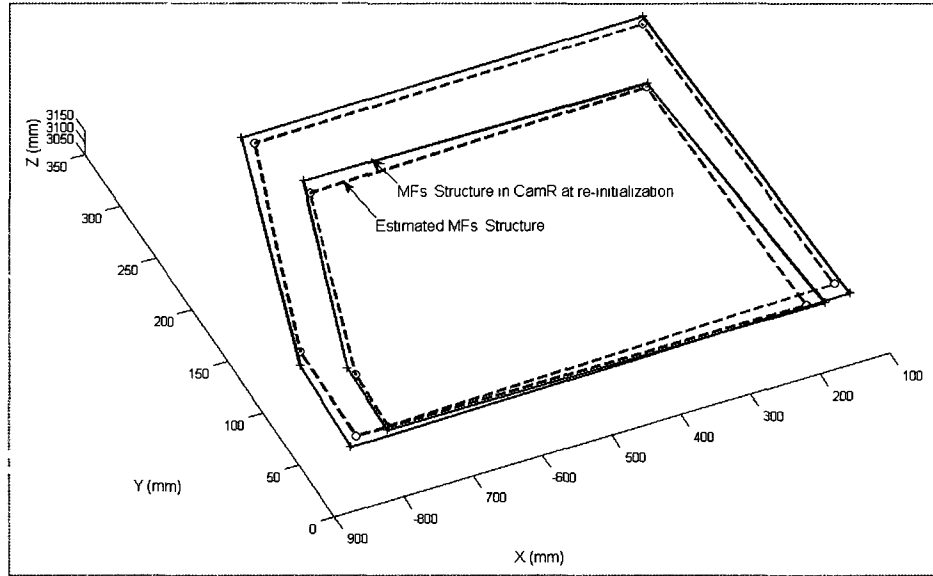


Fig. 6.7. Recovered/estimated 3D MFs' area at re-initialization, defined in CamR's reference frame

Table 6.1. Error analysis of the SLS/SSPME inter-calibration

Maximum of the Absolute Error $M_{ \epsilon }$ (mm)			Root Mean Squared Error ϵ_{RMS} (mm)		
$M_{ \epsilon }^X$	$M_{ \epsilon }^Y$	$M_{ \epsilon }^Z$	ϵ_{RMS}^X	ϵ_{RMS}^Y	ϵ_{RMS}^Z
19.29	8.86	24.11	10.64	6.52	13.56

The 3D surface model of the inspected car door, acquired with the SLS and shown in Fig. 6.1a, was also used in two preliminary tests, whose objectives were to refine the locations of the 3D MFs, recovered with respect to CamL_{SL} , and used in the inter-calibration procedure. A first test involved the detection of the 3D points belonging

to the surface model, located at the smallest Euclidean distance from the 3D points recovered with the proposed MFs-based procedure. Using the camera calibration data these points have been back-projected onto the image plane. Figure 6.8a illustrates the location of their associated 2D points in CamL_{SL} 's frame. Taking into account the fact that the 3D points belonging to the surface model have also been triangulated with the same calibration data used in the MFs-based approach, the back-projection operation does not introduce any bias in the 2D mapping of the these 3D points. By comparing with the location of the pre-selected and refined MFs, shown in Fig. 6.6a, the new MFs are drifted from the exact corners of the inner and outer frame of the door window.

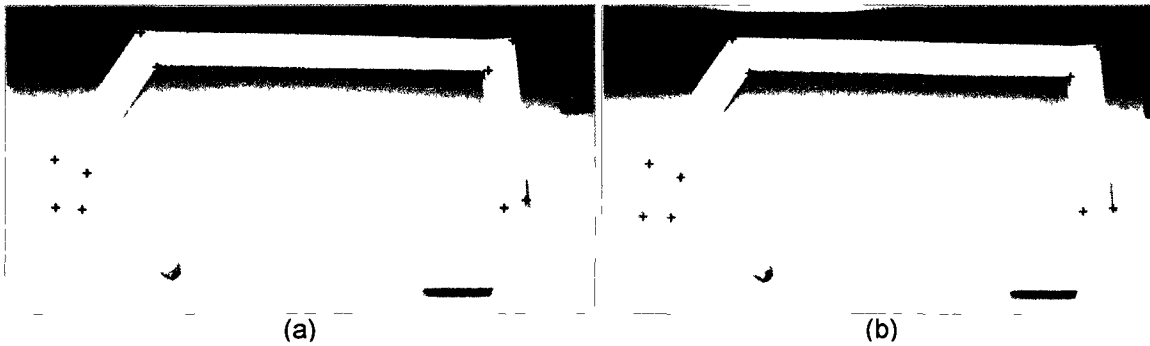


Fig. 6.8. Back-projected MFs in CamL_{SL} 's frame after the refinement based on: (a) Euclidean distance in 3D, (b) Euclidean distance in the X and Y components.

A second test was also performed in which the MFs were replaced with the 3D points belonging to the surface model, located at the smallest Euclidean distance, computed only with the X and Y components, from the 3D points calculated with the MFs-based approach. This assumption was motivated by the fact that a higher error might be present in the depth component of the 3D points belonging to the surface model, especially in the border region, which corresponds to the MFs' area. The back-projections of the newly obtained 3D MFs in the image plane can be observed in Fig. 6.8b. By comparing with the MFs' locations in Fig. 6.8a, the new MFs point more precisely to their aimed location, except for MF_4 and MF_9 which exhibit a similar amount of drift as with the previous approach.

The results obtained for the two measures validating the accuracy of the rigid transformation matrix, with the "refined" 3D MFs did not offer any consistent improvement when compared to the data illustrated in Table 6.1. This is mainly due to the noise present in the data of the 3D surface model, especially on the boundary regions of the object, which are not reliably reconstructed due to the limitations of the 3D imager [81]. Therefore, the inter-calibration building upon the 3D points recovered with

the MFs-based approach was privileged for the integration of the 3D deformations detection block with the autonomous robotic marking station. This decision was also supported by the fact that the final objective is for the robot to mark the deformation area and not necessarily the precise centroid of the deformation. Subsequently, the surface deformations will have to be fixed by factory associates. Therefore an approximate marking of these defects (within a few centimeters) will still be very useful for them.

Finally, it should be noticed that the proposed inter-calibration can also be extended to the cases in which the 3D scanning is not performed with a system composed of a stereoscopic sensor. In this case, the initialization (or re-initialization) frame can be represented by a range (or binary) image generated from the acquired 3D model. Thus, the same feature extraction approach can be extrapolated in the case of this image, and upon the detection of the MFs in 2D, their associated 3D points can be extracted from the 3D model, based on the same mapping that was used to generate this image. With the use of higher resolution scanning methodologies, the accuracy obtained over the boundary regions of the inspected automotive part should be improved, while still making use of the proposed MFs-based inter-calibration procedure.

6.2.3. Robotic Pointing of Surface Deformation Defects

The robotic interaction directly relies on the results provided by the 3D deformations detector and the panel's pose and motion estimator. The fact that the locations of the surface deformations are readily provided in the base reference frame of the robot, O_B , after inter-calibrations, simplifies the path planning for the robot to mark the deformations. This generic solution is adequate independently of the marking strategy, including the use of a marker tip or chalk, a stamping sponge or a spray gun.

For each of the identified dings, the 3D "contact" point with the smallest depth with respect to CamR is extracted, by using the inter-calibration between the SLS and the SSPME performed at re-initialization. This point corresponds to the location where the deformation shows maximal protrusion and defines where the tool's center point (TCP) needs be positioned in order to make contact with the deformation region. Figure 6.9 illustrates the deformation surfaces obtained by applying a Delaunay triangulation [89] on the 3D points of the three dings transferred with respect to the reference frame of the SSPME, that is CamR. Additionally, for each of the deformation surfaces in Fig. 6.9, the line passing through the 3D "contact" point is also represented in red. This line is formed by a set of points whose X and Y coordinates are extracted from the X , Y

components of the “contact” point and their Z components are varied within a small range of the Z coordinate of the “contact” point. As illustrated in Fig. 6.9, the “contact” point is not necessarily the centroid of the deformation surface, given that the “hand-made” defects appended on the door panel do not exhibit a uniformly smooth curvature.

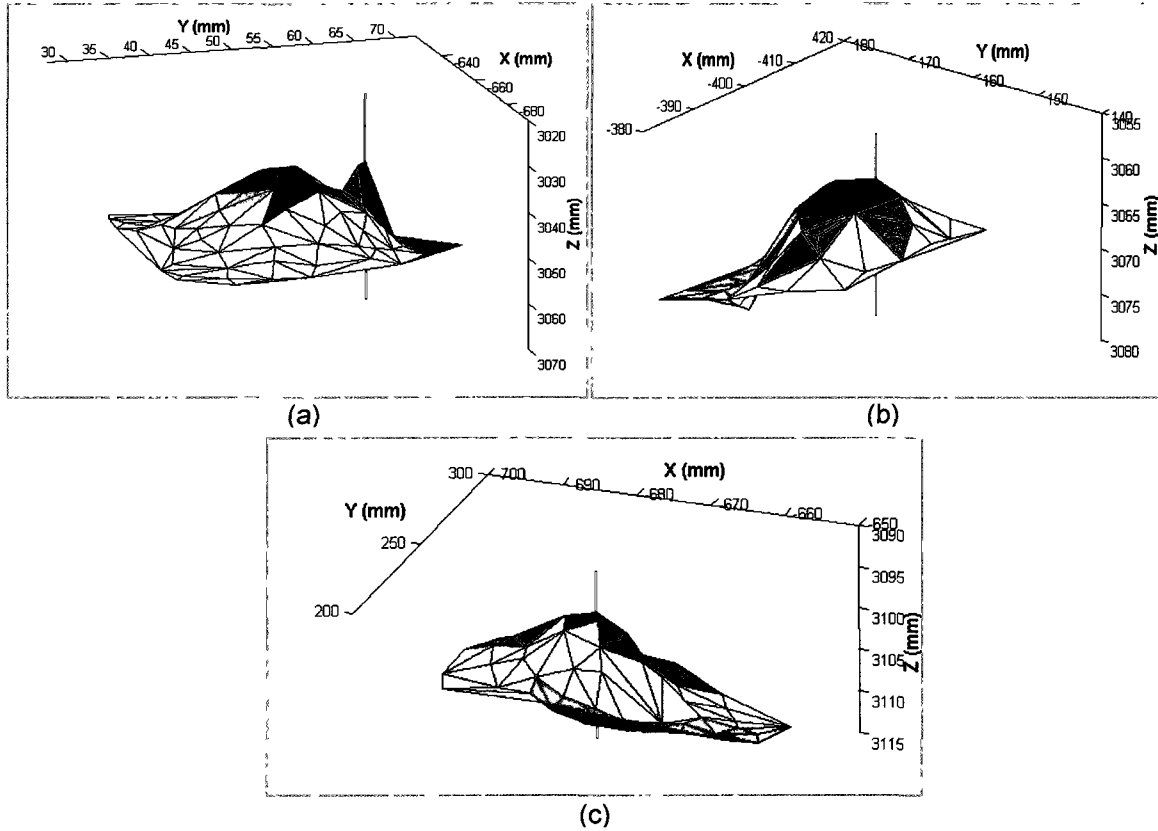


Fig. 6.9. 3D deformation meshes defined with respect to CamR, at re-initialization for: (a) Ding₁, (b) Ding₂ and (c) Ding₃.

Beyond the locations of the detected deformations, the orientation of the marking tool, with respect to the area that the robot needs to point to, must also be specified. For that matter, a least-squares interpolation of a plane is computed from the set of 3D points, expressed with respect to CamR, which belong to each deformation. The obtained interpolated plane for Ding₂, which is translated on the “contact” point of this deformation defect, is illustrated in Fig. 6.10. Similar to the examples in Fig. 6.9, the surface of the deformation, as well as the interpolated plane are defined with respect to CamR. A supplementary reference frame, O_v , is attached to the computed plane, as shown in Fig. 6.10. Its origin is defined by the “contact” point of the deformation area, with the X and Y axes parallel to the interpolated plane vectors and the Z axis pointing out of the plane, perpendicularly to the local surface patch.

The 3D vectors representing the axes of O_v are normalized to form a rotation

matrix that defines the rigid transformation from CamR to the robot's tool, $Q_{Tool/CamR}$. The rotation matrix is estimated as:

$$R_{Tool/CamR} = [\overrightarrow{Y_v} | -\overrightarrow{X_v} | -\overrightarrow{Z_v}] \quad (6.4)$$

where the three linear independent columns are selected such that the tool reference frame, O_T shown in Fig. 6.11a, becomes collinear with O_v , except for the Z axes that point in opposite directions.

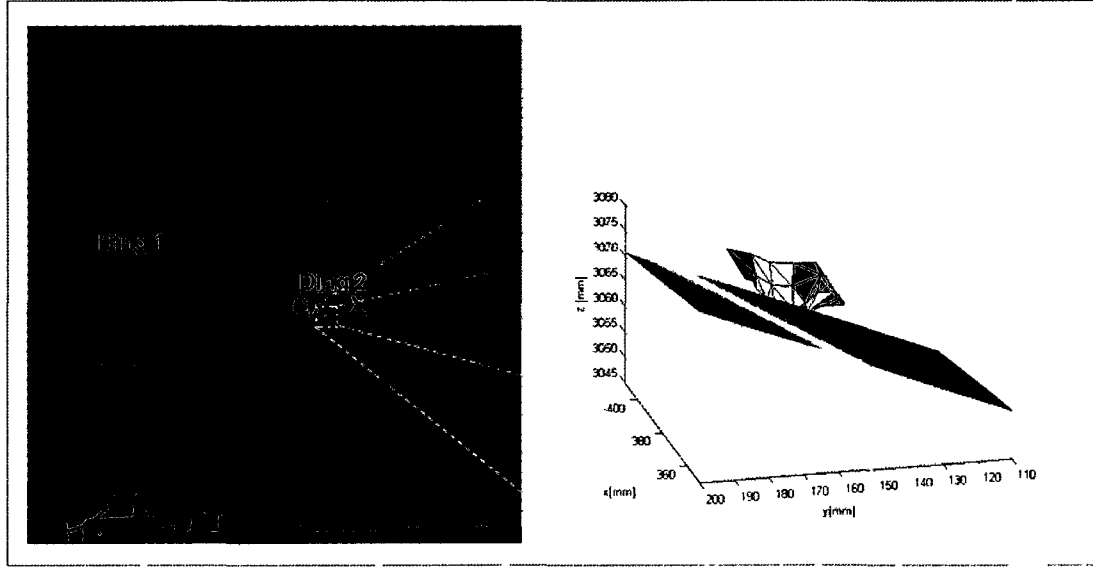


Fig. 6.10. 3D mesh of Ding₂ along with the interpolated plane translated on the "contact" point.

The translation component of $Q_{Tool/CamR}$ is estimated by the position of the "contact" point with respect to CamR. Finally, the transformation defining the pointing pose of the tool with respect to the base of the robot, $Q_{Tool/Base}$, such that the contact operation can be accomplished, is defined by:

$$Q_{Tool/Base} = Q_{CamR/Base} \cdot Q_{Tool/CamR} \quad (6.5)$$

6.3. Robotic Stamping of Deformations in a Static Environment

The final operation related to the robotic marking of deformations was initially tested on a static automotive panel [75]. In these circumstances, the robotic tool was a stamping instrument, described in Section 3.5, and illustrated in Fig. 6.11a and Fig. 6.11b, along with its associated reference frame, O_T . The area of the stamping surface is approximately 8.2cmx8.2cm, which gives sufficient flexibility regarding the exact positioning of the marking device with respect to the location of the deformation.

In order to evaluate the performance of this stamping system, nine scenarios were considered, based on the pose of the car door panel on the assembly line. For

experimentation, a constant speed of $v_{ss} \approx 1.4$ cm/s was set for the sled system, whereas a frame capture rate of $f_{extr}=0.5$ Hz was used by the supervised pose and motion estimation system. In addition, the performance was analyzed with three different orientations of the sled. These orientations were obtained by manually rotating the sled platform around the Y_o axis, as shown in Fig. 6.11c, with different angles $(\theta_0, \theta_1, \theta_2) = (0^\circ, 10^\circ, -15^\circ)$. Moreover, the stamping operation was performed when the panel was located at the beginning (Pos_A), the middle (Pos_B) and the end (Pos_C) of the sled, each location being separated by 27cm, as shown in Fig. 6.11c.

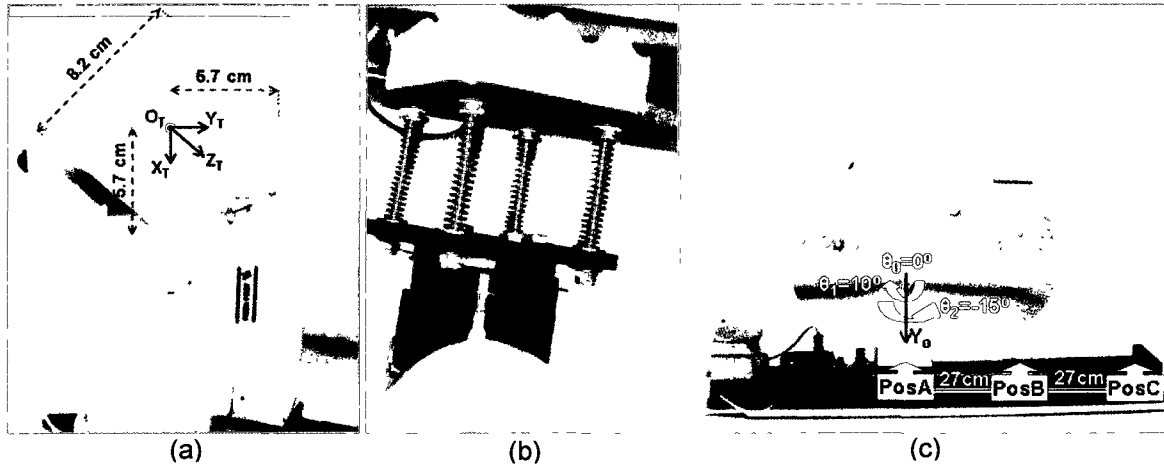


Fig. 6.11. (a) Top view of the stamping tool with assigned reference frame and dimensions, (b) lateral view of the stamping tool, (c) sled system with car door panel and data regarding the performance evaluation of the robotic stamping operation.

In the case of the experimentation performed in Pos_A , for the three different orientations, use was made of the inter-calibration between the SLS and the SSPME, which was computed on-line, as soon as the 3D recovery of the MFs in this initial position was completed. Then, for the performance evaluation in Pos_B and Pos_C two different approaches were tested.

In the first case, use was made of the inter-calibration between the SLS and the SSPME to transfer the 3D points of the deformations, expressed with respect to $CamL_{SL}$, in the reference frame of the SSPME, $CamR$. Then, the location of these points during the subsequent processing cycles of the pose and motion estimator, was updated based on the estimated rigid transformation exhibited by the automotive part during two subsequent frames, as explained in Section 5.3. Therefore, the precision of the pose and motion estimation system had a direct impact on the accuracy of the 3D points associated to the identified deformations, when evaluating the stamping operation in Pos_B and Pos_C , regardless of the car door's orientation relative to the acquisition

system. Thus, although the stamping operation was performed on the stationary panel, the integration of the proposed pose and motion estimator in the robotic system was still possible.

In the second case, the inter-calibration between the SLS and the SSPME was computed at every processing cycle of the pose and motion estimator, as soon as the 3D locations of the MFs had been recovered. Therefore, under these settings, the motion estimation component was substituted with the inter-calibration procedure, which shares the same complexity, as it relies on the same number of MFs.

In each of the nine scenarios considered, the robot followed the state flow shown in Fig. 6.12. Starting from the “home” position, the manipulator robot was guided in the “pre-location” (“pre-contact” point) of the first defect. In this “pre-location” the stamping tool had the same attitude with respect to the deformation area, as described in Section 6.2.3, but a reserve of 20cm was subtracted from the X (depth axis of O_B , as shown in Fig. 3.11a) component of the “contact” point, as defined with respect to the robot’s base reference frame, O_B , by using eq. (6.5). Then, the robot was driven to the “contact” point of the deformation in order to perform the stamping operation. Subsequently, the robot arm was guided back to the “pre-location” of the first defect. Then it started a new cycle by going to the “pre-location” of the second defect, and so on.

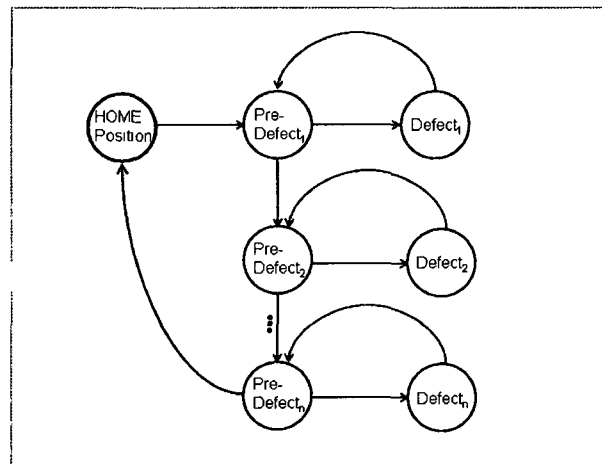


Fig. 6.12. State-flow diagram of the manipulator arm during the stamping procedure on the stationary automotive panel.

After stamping the last identified surface deformation, the robot returned in the “pre-location” of the last defect, and then it completed its motion cycle by returning to the “home” position. With the purpose of accurately measuring the stamping errors, a reserve of 2cm was subtracted from the magnitude of the X component of the “contact” point of the defects, as recovered with respect to O_B , when the robot was programmed

to stamp the respective surface deformation (i.e. reaching the “Defect_{1,2, n}” states as shown in Fig. 6.12). This reserve was introduced with the purpose of accounting for a possible slight “over-determination” of the X (depth) component of the “contact” point of the defects, which would affect the physical stability of the panel over the sled system. Figures 6.13a, 6.13b and 6.13c illustrate the robotic stamping operation in the scenarios $(Ding_1, Pos_A, \theta_0)$, $(Ding_2, Pos_B, \theta_1)$ and $(Ding_3, Pos_C, \theta_2)$.

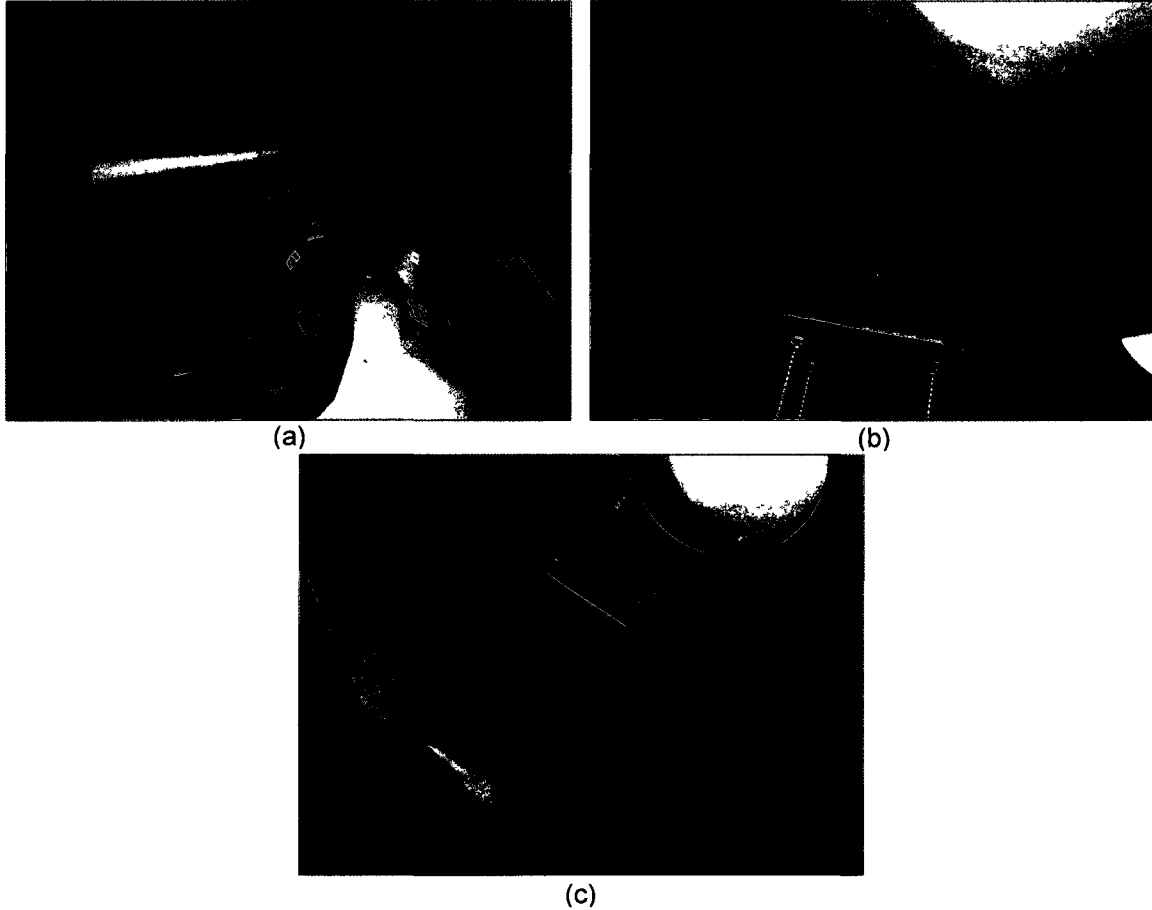


Fig. 6.13. Deformation defects stamping results in the scenarios: (a) $(Ding_1, Pos_A, \theta_0)$, (b) $(Ding_2, Pos_B, \theta_1)$, (c) $(Ding_3, Pos_C, \theta_2)$.

In order to monitor the accuracy of the stamping operation, the mean absolute errors, $|\bar{e}_X|, |\bar{e}_Y|, |\bar{e}_Z|$, characterizing the average displacement error of the tool center point (TCP) to the “contact” point of the three deformation regions, were measured, with respect to the axes of the robot’s base reference frame, O_B . Table 6.2 presents the mean absolute errors for the nine scenarios considered. Additionally, the subscript “Met₁” is related to the first mentioned approach in which the inter-calibration between the SLS and the SSPME was performed only once, when the car door model was located in Pos_A , for all the different orientation cases. Subsequently, the subscript “Met₂”

is linked to the second methodology in which the inter-calibration between the SLS and the SSPME was computed at every processing cycle of the pose and motion estimator. The errors are fairly uniform in all directions and remain stable, independently from the position or orientation of the panel along the track. Also, as it can be noticed from Table 6.2, the obtained mean absolute errors vary within the same range for both considered approaches, and no obvious benefits are provided by either method.

Table 6.2. Accuracy of the defects stamping operation.

Mean Absolute Error	Pos _A			Pos _B			Pos _C		
	θ_0	θ_1	θ_2	θ_0	θ_1	θ_2	θ_0	θ_1	θ_2
$ \bar{e}_X _{Met_1}$ (cm)	1.2	1.4	1.3	1.5	1.3	1.4	1.4	1.3	1.4
$ \bar{e}_X _{Met_2}$ (cm)				1.3	1.4	1.3	1.2	1.5	1.5
$ \bar{e}_Y _{Met_1}$ (cm)	1.1	0.9	1.2	1.2	1.2	1.3	1.3	1.2	1.4
$ \bar{e}_Y _{Met_2}$ (cm)				1.1	1.4	1.4	1.4	1.1	1.3
$ \bar{e}_Z _{Met_1}$ (cm)	1.2	1.3	1.4	1.5	1.6	1.5	1.7	1.5	1.6
$ \bar{e}_Z _{Met_2}$ (cm)				1.7	1.5	1.6	1.7	1.6	1.4

The relatively small changes in the magnitude of the mean absolute errors for the cases in which the sled system is rotated with respect to the SSPME, validate the integration of the proposed solution in an industrial application in which the assembly line contains curved regions.

The principal sources of error originate from the limited resolution of the SLS that slightly biases the exact locations of the deformations over the panel, and from the accuracy of the pose and motion estimator, together with the rigid transformations estimated via the inter-calibration procedures [75]. The latter two are considerably influenced by the fact that the stereoscopic system must remain at a relatively large distance (about 3m) from the assembly line in order to provide a sufficiently wide field of view to track the panel over the entire inspection work cell. In addition to this, the inter-calibration between the SLS and the SSPME relies only on 10 MFs, as discussed in Section 6.2.2, thus influencing the precision of the robotic stamping operation.

Regarding the inter-calibration between the SLS and the SSPME, a different methodology inspired from the fine-level supervisory layer, described in Section 5.4.2, was also tested. In this approach, the two sets of 3D MFs are transferred to the normalized space S^N [45], and then the rotation component of the rigid transformation, computed with the full set of points, is used to de-rotate the normalized set belonging to CamR. Subsequently, the displacement residuals, defined in Section 5.4.2.2, are computed and three points that do not form a collinear configuration and have the smallest displacement residuals are extracted and form the data sets for the computation of a new rigid transformation between the SLS and the SSPME. Although this new inter-calibration resulted in smaller magnitudes for the maximum of the absolute error and the root mean squared error (when computed with only the three points forming the extracted set), the obtained stamping accuracy was inferior to the one reported in Table 6.2. Therefore, the uncertainty present in the 3D locations of the MFs (with respect to both SLS and SSPME), has a stronger impact on the precision of the inter-calibration than the least-squares method used for computing the rigid transformation.

By following the state-flow diagram shown in Fig. 6.12, the total operation of robotic stamping for the three identified dings over the surface of the door panel, took $t_{\text{stamp}} \approx 40\text{s}$, when the speed of the robot was set to 25% of its maximum value. According to [82], the maximum linear speed for the joint-interpolated motion is 4m/s for the CRS F3T manipulator. Additionally, the MFC application “RoboStaticStamp” associated to the robotic stamping operation on stationary objects is implemented in Visual C++, together with the open-source library for computer vision, OpenCV [21]. Moreover, “RoboStaticStamp” runs in real-time on a station having a 5130 Intel Xeon CPU at 2 GHz and 2.75GB of RAM.

The precision achieved in this validation gate makes the proposed approach suitable for marking with the stamping sponge, shown in Fig. 6.11a, or a spray gun, given that the final objective is to mark the region that contains the deformation within a few centimeters accuracy. The exact location of the deformation within the marked region is easily determined by human workers who will perform the repair in a separate station, using the marks as a guide. For this matter, the proposed stamping solution represents a viable alternative to perform fully automated region marking of deformations over large surfaces and for substantial volumes of production.

6.4. Robotic Marking of Deformations on Moving Panels

This section describes the extension to the on-line operation of robotic marking of deformation defects on an assembly line. Therefore, the current goal is to perform the marking operation on the moving panel. As explained in Section 3.5, the three major components of the autonomous robotic system, which works in interaction with the moving object, are represented by the supervised pose and motion estimator and the two inter-calibrations, described in Sections 6.2.1 and 6.2.2.

6.4.1. On-line Robotic Stamping of Surface Deformation Defects

In the first stage of the robotic interaction with a moving object, the manipulator robot was programmed to point the “pre-location” of the “contact” point (“pre-contact” point) of one of the deformations, for the entire motion sequence of the automotive panel. Thus, when compared to the scenarios described in Section 6.3, when the stamping operation was performed on a static object, now the robot is all the time present in the scene, in order to continuously point the “pre-contact” location of the ding, within a “look-and-move” architecture [63]. As a result, with these supplementary tests that get closer to the real application on an assembly line, the robustness of the proposed pose and motion estimator to situations where the robot and people are present in the scene will be extensively evaluated. In addition to this, in order for the robot to continuously point the deformation area, the transformation $Q_{Tool/Base}$, discussed in Section 6.2.3, is computed at every processing cycle of the pose and motion estimator. A velocity of approximately $v_{ss}=1.4\text{cm/s}$ was set for the sled system, whereas a frame extraction rate of $f_{extr}=0.5\text{Hz}$ was selected for the pose and motion estimator. In this way, a number of $N_{frames}=20$ frames have been extracted and processed, for a total period of time, $t_{exp}=40\text{s}$.

The results related to the accuracy of the robotic stamping operation in the static environment, shown in Table 6.2, represent the pre-requisite for the analysis that will be performed in the next paragraphs, related to the attitude of the robotic tool with respect to the deformation area, during the continuous pointing experiment. Thus, the 3D location of the TCP (Tool's Center Point), which corresponds to the “pre-contact” point of the selected location, is monitored throughout the entire motion cycle of the object.

In order to simplify the analysis, a scenario in which the sled system is located on a plane approximately perpendicular to the principal axis of the acquisition sensor, is considered, similar to the scenario characterized by the angle $\theta_0=0^\circ$, in Section 6.3. As a result, the Z-component of the 3D MFs, throughout the motion cycle, should contain very

small variations between the initial and final magnitudes. Moreover, in order to start from the core component of the proposed visual servoing architecture, in order to inspect the effects of each of its embedded sub-systems on the attitude of the robotic gripper relative to the moving panel, results related with the pose and motion estimation system will first be presented. Therefore, Fig. 6.14a illustrates the 3D trajectory (with respect to CamR) associated to MF_6 , as indexed in Fig. 4.1a, for the cases in which the robot did not appear in the view of the cameras (labeled as “no robot in the view” in Fig. 6.14a and 6.14b), and for the cases where the manipulator robot was programmed to continuously point the pre-location of $Ding_2$ (labeled as “robot in the view” in Fig. 6.14a and 6.14b).

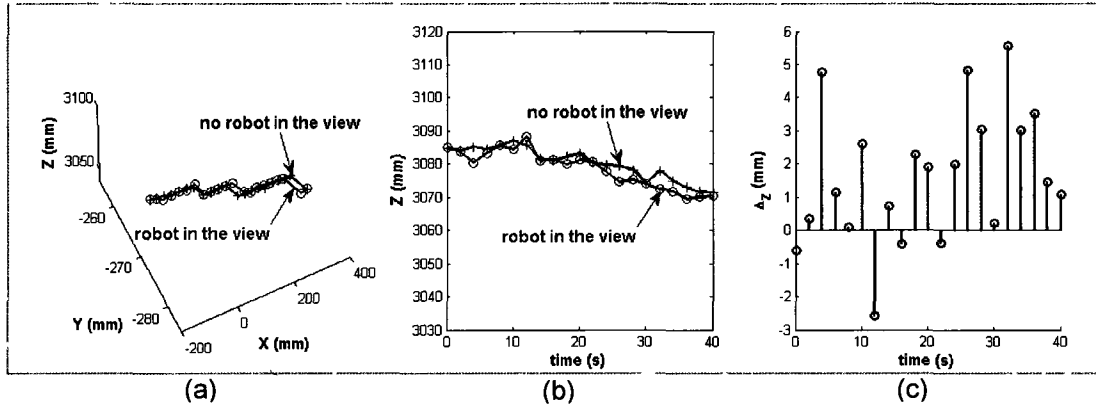


Fig. 6.14. Results related to MF_6 (relative to CamR) in the first scenario with/without the robot in the view of the acquisition system: (a) 3D trajectories of MF_6 , (b) Depth (Z) component of MF_6 coordinates during the motion cycle, (c) Z-differences between the two MF_6 trajectories.

The Z-component of the recovered 3D locations for MF_6 , during the two motion cycles (the first one without the manipulator robot in the view of the cameras, and the second one with the robot in the view) are shown in Fig. 6.14b. The differences between these two depth components (Δ_z) are computed at every processing cycle, by subtracting from the Z component of MF_6 , in the case with “no robot in the view”, the depth component of MF_6 when the robot appeared in the view of the acquisition sensor. The purpose of this calculation, which quantizes how the depth trajectories differ for the two similar motion sequences, is to inspect the robustness of the pose and motion estimator to the partial occlusions caused by the manipulator robot, during the continuous pointing operation. These differences, which should be zero, ideally, are represented in Fig. 6.14c.

The difference (maximum depth variation) between the maximum and minimum depth of MF_6 throughout the individual motion cycles, was $\delta_{Z_{CamR}}^{Max_1} = 15.5\text{mm}$ for the case in which the robot was not present in the view of the cameras, whereas a maximum depth

variation of $\delta_{Z_{CumR}}^{Max_{II}} = 18.7\text{mm}$ was obtained for the case in which the robot was continuously pointing the “pre-contact” point of the second ding. Therefore, the difference between the two maximum depth variations is very small, due to the contributions of the supervisory layer, introduced in Section 5.4, in all the processes embedded in the pose and motion estimator. Nevertheless, the imperfect orthogonality between the principal axis of the SSPME and the plane in which the panel exhibits its motion, also contributes the small depth variations, which exhibit a decreasing magnitude throughout the end of the motion sequence.

Additionally, in Fig. 6.14c it can be noticed that the highest difference (Δ_Z) between the depth-components of MF_6 (for the two motion cycles) is approximately 5.57mm, for the entire sequence. Overall, the variations in the pose and motion estimations for the cases in which the robot is present or absent from the view of the acquisition system remain very small. This behavior can be attributed to the introduction of the supervisory layer. This experience also validates the robustness of the pose and motion estimation system to occlusions, “highlights” or specular reflections present in the images, or to drifting effects manifested by the MFs during tracking over long-sequences.

Another advantage of the testing scenario is related to the fact that the main motion exhibited by the automotive panel is given by a translation along the Z axis, when expressed with respect to the reference frame of the robot’s base, O_B , shown in Fig. 3.11a. Therefore, after embedding the inter-calibration between the SSPME and the robot’s base, the variations in the X-component of MF_6 , should also be contained within small values, since the panel’s motion takes place on a plane approximately perpendicular to the X axis of O_B , which is almost parallel with the optical (Z) axis of the SSPME. As a result, the proposed experimental validation allows for the analysis of the impact of the inter-calibration between the SSPME and the robot’s base, which is the second major component of the robotic tracking and marking station, on the depth component of MF_6 , throughout the considered motion cycles, similar to the previous experiment.

Figure 6.15a illustrates the 3D trajectory of MF_6 , with respect to O_B , for two successive cases where the robot was absent or present, respectively, in the view of the SSPME. Similar to the previous experimental tests, in the latter case, the “pre-contact” point of $Ding_2$ is continuously pointed by the robot. The X-component of the 3D MF_6 during the motion cycles is represented in Fig. 6.15b. For these cases, the maximum

variation in the magnitude of the X -component was $\delta_{X_{O_B}}^{\text{Max}_I} = 9.3\text{mm}$ for the setting in which the robot was not present in the stereo-views, while a maximum variation of $\delta_{X_{O_B}}^{\text{Max}_{II}} = 15.1\text{mm}$ was observed in the case in which the manipulator robot performed the continuous pointing operation and therefore partially occluded the views. These magnitudes are very close to the results obtained with respect to CamR, therefore the inter-calibration between the SSPME and the robot's base is not considerably affecting the depth variations of MF_6 , throughout the motion cycles.

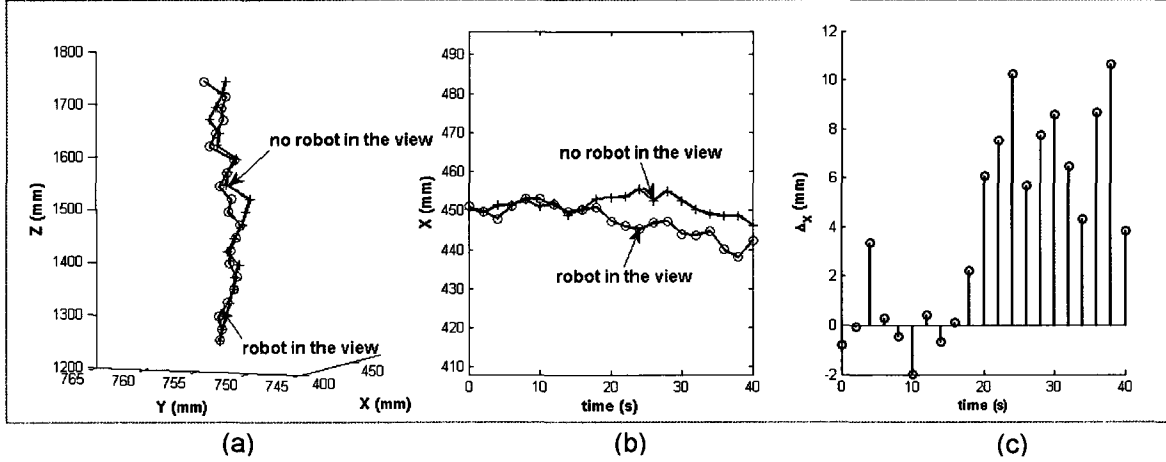


Fig. 6.15. Results related to MF_6 (relative to O_B) in the first scenario with/without the robot in the view of the acquisition system: (a) 3D trajectories of MF_6 , (b) Depth (X) component of MF_6 coordinates during the motion cycles, (c) X -differences between the two MF_6 trajectories.

The difference (Δ_X) between the X -components of MF_6 , during the currently considered motion cycles, is illustrated in Fig. 6.15c. Similar to the previous case, Δ_X is computed by subtracting the X -component of MF_6 , in the case when the robot was in the view of the SSPME, from the depth component of MF_6 (relative to O_B) when the robotic arm did not appear in the view of the acquisition sensor. One can notice that the divergence that occurs between the two X -trajectories, from approximately the middle part of the processing sequence, remains between bounded limits. Moreover, the maximum displacement between the X -components is approximately 10.7mm. When compared to the maximum displacement between the Z -components, obtained with respect to CamR, there is a slight increase (less than 6mm) in this residual, which is mainly caused by the rigid transformation matrix computed between the robot's base and CamR, at inter-calibration, coupled with the imperfect colinearity between the X axis of O_B and the Z axis of CamR, as well as the imperfect orthogonality between X axis of O_B and the plane in which the automotive body exerts its motion.

In order to test the impact of the inter-calibration between the SLS and the SSPME, as the final major component of the proposed quality control application, on the attitude of the stamping tool relative to the panel's surface, the trajectory of the "pre-contact" point of Ding₂ was also inspected. Therefore the same scenario, extensively tested in the last two experiments (the former with respect to CamR, the latter relative to O_B) in which the object's motion is mainly a translation along the Z axis of the robot's base, was considered. Since the current objective is to inspect the location of the "pre-contact" point of Ding₂ (by continuously pointing it), the manipulator will appear in the view of the acquisition sensor, for the entire motion sequence of the car door.

Similar to the previously tested cases involving MF₆, the depth variations of the "pre-contact" point of Ding₂, evaluated with respect to O_B, should also remain within small bounds. Moreover, they should manifest a similar behavior as in Fig. 6.15b (the case with robot in the view), as the contact point of the second ding belongs to the same rigid structure as MF₆. This constraint is essential from the perspective of the robotic acting on the moving panel, as these depth variations can result in two faulty robotic interactions, under a framework involving the physical touching of the moving part. The first one is linked to the situation in which the "under-estimation" of depth results in the impossibility of physically marking the defect, whereas the "over-estimation" of depth might affect the physical stability and integrity of the automotive panel on the assembly line. As a result, the proposed on-line robotic tracking and marking system should exhibit fault tolerance to these two situations.

Figure 6.16a illustrates the trajectory of the "pre-contact" point of Ding₂ (pre-Ding₂), with respect to the robot's base O_B, for the two cases in which the inter-calibration between the SLS and the SSPME was performed at the beginning or at every processing cycle of the pose and motion estimator. Figure 6.16b shows the X-component of pre-Ding₂, for the two considered inter-calibration methodologies, during the motion cycle. In these cases, the maximum variation in the X component for the first inter-calibration approach was $\delta_{Xd_{O_B}}^{Max_I} = 20.7\text{mm}$, whereas a maximum variation of $\delta_{Xd_{O_B}}^{Max_{II}} = 28.9\text{mm}$ was obtained for the motion cycle in which the SLS/SSPME inter-calibration was calculated at every processing step. As mentioned in Section 6.2.2, in the cases in which the inter-calibration SLS/SSPME is computed at every processing cycle of the pose and motion estimator, the visual servoing data does not rely anymore on the motion estimation data (characterizing the motion exhibited by the panel between

the previous and the current frame extraction). As a result, the motion estimation procedure is substituted by the SLS/SSPME inter-calibration technique, which shares the same computational complexity, as it relies on the same least-squares approach [57], applied on the same number of MFs.

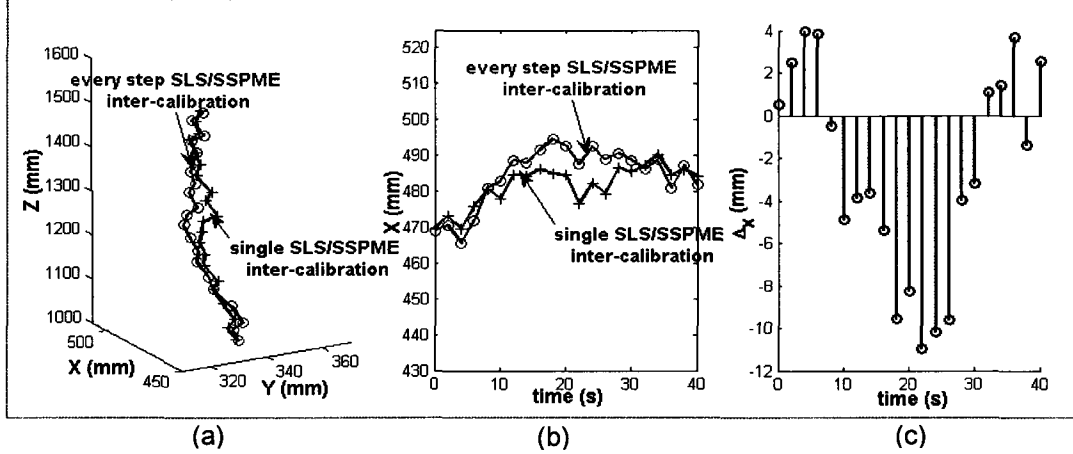


Fig. 6.16. Results related to pre-Ding₂ (relative to O_B) with the two SLS/SSPME inter-calibration methodologies: (a) 3D trajectories of pre-Ding₂, (b) Depth (X) component of the 3D pre-Ding₂ during the motion cycles, (c) X-differences between the two pre-Ding₂ trajectories.

When compared to the previous results, in which the trajectory of MF₆ (with respect to O_B) was monitored, the maximum depth variation continued to increase with another 5.6mm, and 13.8mm, respectively, for the two considered inter-calibration approaches. Additionally, Fig. 6.16c illustrates the differences between the X-components of pre-Ding₂ for the two motion cycles (in the former, the inter-calibration between the SLS/SSPME is performed only once, whereas in the latter, the inter-calibration SLS/SSPME is computed at every processing cycle of the pose and motion estimator). Under these settings, the maximum acquired depth difference had an absolute value of 10.9mm. Since the increase in the maximum depth variation was smaller for the first inter-calibration methodology, the decision was made to proceed with inter-calibration only at the beginning of the tracking sequence.

Moreover, the trajectory of the depth component of pre-Ding₂ in the case of the first inter-calibration methodology is more similar to the trajectory obtained for MF₆ (relative to O_B , as in Fig. 6.15b with “robot in the view”) than in the case of the second inter-calibration approach. However, although the maximum depth variations are approximately within the same range ($\delta_{X_{O_B}}^{\text{Max}_1} = 15.1\text{mm}$ when expressed to O_B in Fig. 6.15b with “robot in the view”, and $\delta_{X_{d_{O_B}}}^{\text{Max}_1} = 20.7\text{mm}$, relative to O_B in Fig. 6.16b with the first inter-calibration approach), it can be noticed that the depth components exhibit a

contrasting behavior, during the motion sequence of the object. Indeed, in Fig. 6.15b (with “robot in the view”) it seems that the object is getting closer to the robot’s base, whereas in Fig. 6.16b (with the first inter-calibration approach) the depth of the panel with respect to the robot’s base (given by the X -component of the “pre-contact” point of $Ding_2$) is increasing during the monitored motion cycle. These contradictory behaviors, coupled with the increase obtained in the maximum depth variation throughout the motion sequence are the effects of the limited accuracy obtained for the inter-calibration between the SLS and the SSPME. The pose and motion estimator and the inter-calibration between the SSPME and the robot’s base also contribute to these depth variations, but with a lighter weight, as demonstrated by the previously two experiments, in which the trajectory of MF_6 was inspected with respect to $CamR$, and O_B , respectively.

Although in Section 3.5 it was mentioned that the compliance of the stamping sponge allows a spring compression of approximately 1.5cm, which supported its integration in the prototype for robotic stamping on a stationary object, the depth variations shown in Fig. 6.16b demonstrate the need for eventually complementing the proposed passive vision system with an additional proximity sensing device, especially when performing the robotic stamping operation on a moving object. Such an extra sensing device can take the form of an infrared sensor or a mini-laser scanner that can be connected to the robotic gripper in order to better supervise the close interaction with the moving panel. In this way, the physical stability of the panel during the motion cycle can be reliably maintained, by extensively monitoring for the two faulty robotic interactions (related to the “under-estimation” or “over-estimation” of the depth components associated to the contact points of the defects). As a result, the redundancy that this complement to the sensing architecture would add to the defects marking system would have beneficial effects in providing safer and fault-tolerant navigation data to the robotic system. However, as specified in Section 1.2, the cost and time constraints imposed by the proposed quality control application preempt the incorporation of an additional sensing device within the visual servoing architecture, in the context of this work.

On the other hand, the limited ranges in which the depth components vary throughout the motion cycle (Fig. 6.16b) can accommodate the use of the passive stereo-vision sensing system within an on-line robotic marking architecture that relies on a spray gun methodology rather than a stamping approach. With a spraying approach, the robot does not need to physically touch the moving object since a specific distance

needs to be maintained between the spray gun and the deformation area. Such an approach appears to be more adequate and fault-tolerant for the application considered given the depth variations observed throughout the motion cycles. This strategy also satisfies the final objective of the proposed robotic application, which is to mark the region that contains the deformation within a few centimeters accuracy, given that the final repairs will be performed by factory associates who can use the markers to quickly locate the damages. By selecting a spraying gun approach rather than a stamping solution, the complexity level of the hardware architecture remains the same, and no modifications are required to the proposed pose and motion estimator. The following section proposes a closer examination of this alternative solution.

6.4.2. Robotic Interaction with Motion Prediction using a Spray Gun

In the experimentations discussed in Section 6.4.1, where the manipulator robot was programmed to continuously indicate the “pre-contact” point of $Ding_2$, the visual servoing data did not include any motion prediction estimations. Specifically, upon the extraction of a new set of stereo-frames, at time t_{ext} , for instance, and the completion of the structure and motion estimation cycle, followed by the computation of the transformation $Q_{Tool/Base}$, as explained in Section 6.2.3, the robot was driven to the “pre-contact” point of $Ding_2$. Obviously, the indicated position belonged to the estimated location of the “pre-contact” point of $Ding_2$, at time t_{ext} . However, during the entire computational time of the visual processing followed by the robotic movement cycle, the automotive panel continued to exert its motion on the assembly line. Therefore, the manipulator robot was slightly diverged from the “pre-contact” point of $Ding_2$, for the entire motion sequence, with one exception happening at the end of the motion cycle, when the final pair of frames processed by the pose and motion estimator were extracted very close to the completion time of the panel’s motion. As a result, to ensure the accuracy for the operation of the on-line robotic marking station for autonomous spraying of deformation defect locations, a motion prediction methodology needs to be embedded in the pose and motion estimator process.

In order to reproduce in the laboratory the on-line defects marking process, performed by a robotic system equipped with a spray gun, the end-effector tool, shown in Fig. 3.11a, was redesigned. The base of the pointing tool was maintained, and an LED light pointer was attached to it, as illustrated in Fig. 6.17a, in order to both indicate the deformation area and to provide a visual insight into the region that will be sprayed.

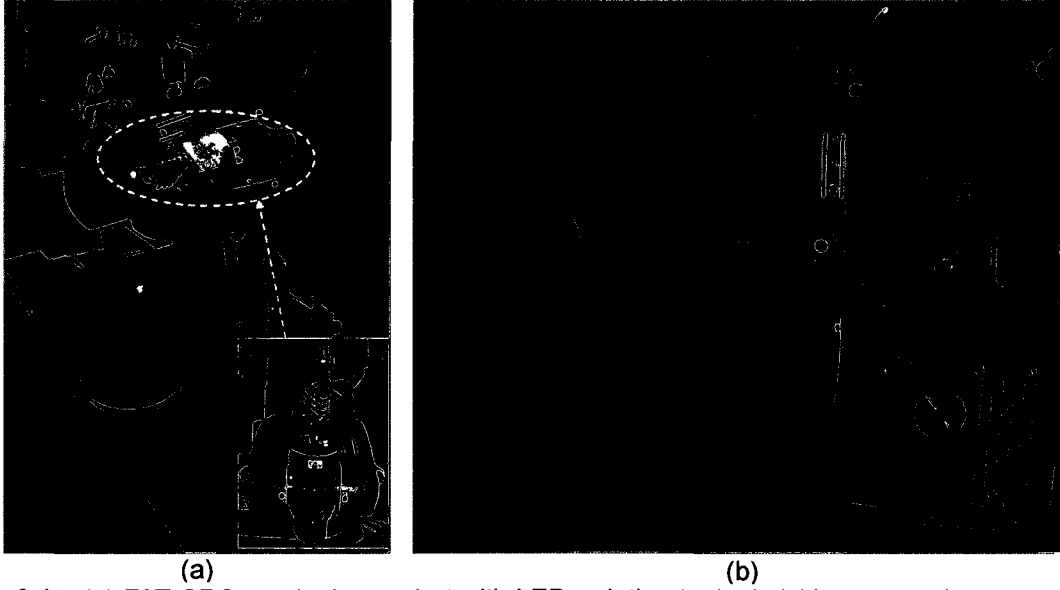


Fig. 6.17. (a) F3T CRS manipulator robot with LED-pointing tool mimicking a spraying operation, (b) manipulator robot pointing on the first ding.

For instance, Fig. 6.17b was extracted from an experimental test in which the manipulator robot, equipped with the LED-pointing tool, marked the location of the first ding (seen as a slightly brighter spot on the door panel), while maintaining a 20cm distance in the X direction away from the “contact” point of Ding₁.

For the development of the motion prediction sub-system, some knowledge about the motion generator (sled system) was used. As mentioned in Section 4.4.2, the sled system exhibits a velocity response similar to a “ramp” function. This behavior is apparent at the beginning and the end of the motion sequence. Therefore, the proposed motion prediction procedure is applied during the interval $[t_{initial}+4s, t_{final}-4s]$, where $t_{initial}$ is the starting time of the motion, and t_{final} represents the time of motion completion. The 4s delay at the beginning and the end of the sequence, was selected in accordance to the 5% settling time of the sled’s velocity control system, when a magnitude of $v_{ss} \approx 1.4\text{cm/s}$ is chosen. Taking into account the relatively constant motion of the sled system, after acceleration and before deceleration, which resembles that of a real conveyor, the prediction sub-system builds upon a “one-step” motion extrapolation methodology. Thus, the central component of the motion prediction process is related to the rigid transformation characterizing the motion exhibited by the panel between two extracted frames, for instance at time t_i and t_{i+1} , which is applied to the 3D points of the deformations, in order to update their location corresponding to the time t_{i+1} . This rigid transformation is actually applied twice, in order to predict the position of the deformation points at time t_{i+2} , when the spraying operation can happen. As a result, the location of

the defects at time t_{i+2} (“one-step (period)” in front of t_{i+1}), is made available to the robotic station. In this way, the LED-pointing operation is more accurate, as the robot is part of a “look, predict and move” approach. The only requirement associated to the proposed “one-step” motion extrapolation prediction approach is related to the processing times of the pose and motion estimator and the robotic movement procedure. As a result, the sum of the computational time of the pose and motion estimator and the time required by the manipulator robot to reach the imposed position needs to be smaller or equal to $1/f_{extr}$, where f_{extr} is the frame extraction rate. Under the current configuration, the pose and motion estimator can operate in real-time, up to an average update rate of $\bar{f}_u \cong 5\text{Hz}$. As a result, the maximum time available for the robotic station to arrive at the imposed location, computed as the difference between the frame extraction period and the average processing time of the pose and motion estimator, is approximately $\bar{t}_{arrive}^{max} = \frac{1}{f_{extr}} - \frac{1}{\bar{f}_u} \cong 1.8\text{s}$, for the cases in which $f_{extr}=0.5\text{Hz}$.

Following an experimental procedure similar to that described in Section 6.4.1, the precision analysis of the motion prediction process is initially performed with respect to the pose and motion estimator, which is the first major component of the robotic tracking and marking station. Therefore, the estimated rigid transformation, characterizing the motion exhibited by the panel between the previous and the currently extracted frames, is applied to the MFs, recovered by the pose estimation procedure applied on the current frames, in order to anticipate their 3D position in the next frame, which will be extracted *a posteriori*. The scenario considered in Section 6.4.1 is selected for these tests. Thus the motion of the automotive panel is performed over a plane approximately perpendicular to the principal axis of the stereoscopic sensor. Figure 6.18a shows the estimated and predicted trajectory of MF₇, during the prediction interval [6s, 38s], as expressed with respect to CamR. Also, Fig. 6.18b illustrates the X, Y, Z components of the prediction error computed as the difference between the estimated and the predicted 3D positions of MF₇, during the prediction motion cycle.

As can be noticed from Fig. 6.18b, the X, Y components of the prediction error are inferior to $\pm 1\text{mm}$, while the maximum absolute value of the prediction error on the Z component is 6.35mm. As a result, the relative precision of the proposed prediction methodology validates its integration in the final prototype for robotic spraying of deformations given that the accuracy on the depth (distance to the panel) is less critical than in the other directions. Moreover, during the entire processed sequence examined

here, the manipulator robot was present in the scene, continuously aiming at the “pre-contact” point of the second ding, as in Section 6.4.1.

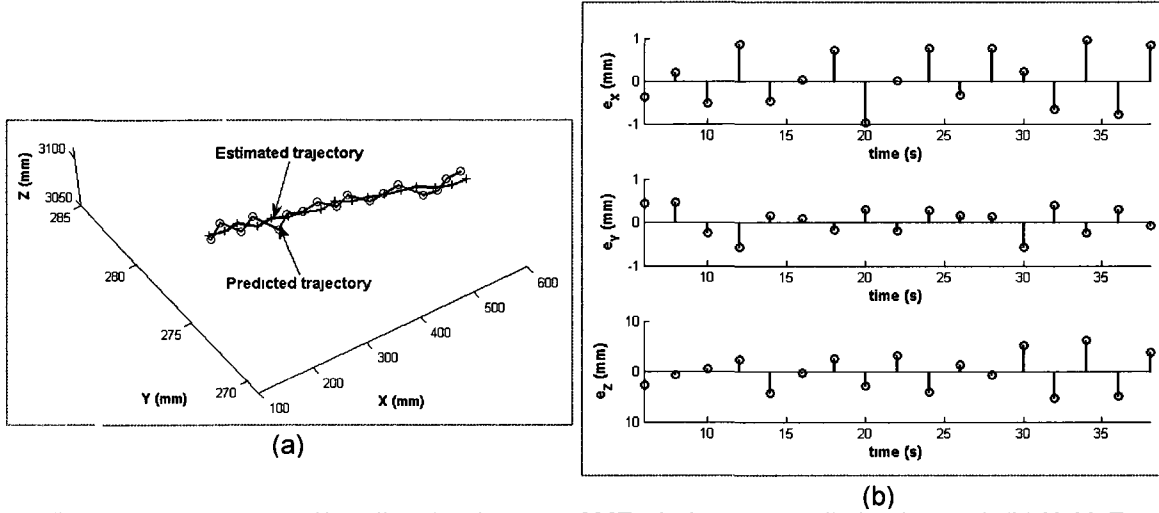


Fig. 6.18. (a) Estimated/predicted trajectory of MF₇ during the prediction interval, (b) X, Y, Z components of the prediction error with respect to CamR.

The precision of the prediction method was also tested with respect to the robotic station, in order to evaluate the effects of the two inter-calibration procedures (the former between the SSPME and the robot’s base, and the latter relating the SLS and the SSPME) on the prediction error. Therefore, the predicted and estimated locations of the “pre-contact” point of Ding₂, were also inspected during the same prediction interval [6s, 38s], resulting in the trajectories shown in Fig. 6.19a, expressed with respect to the robot’s base, O_B.

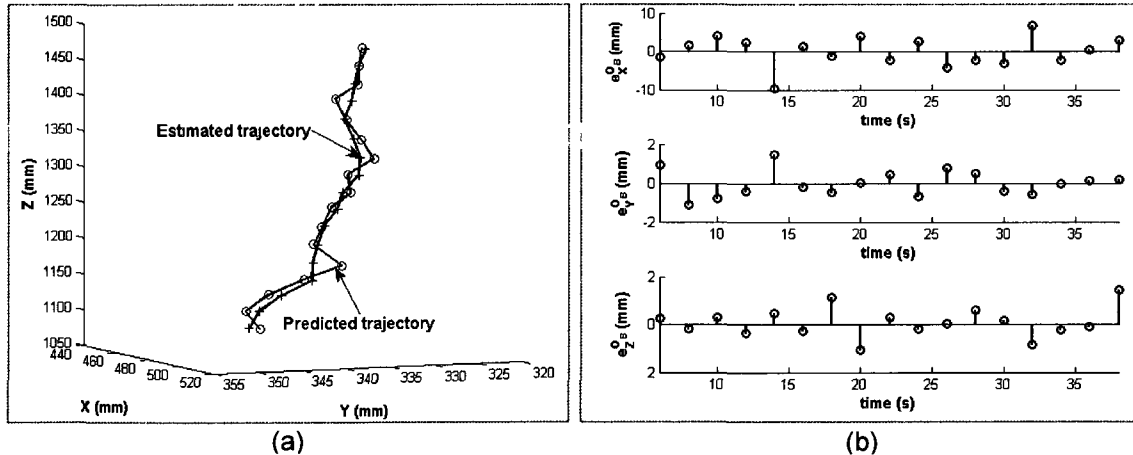


Fig. 6.19. (a) Estimated/predicted trajectory of the “pre-contact” point of Ding₂ during the prediction interval, (b) X, Y, Z components of the prediction error, with respect to O_B.

Additionally, Fig. 6.19b illustrates the X, Y and Z components of the prediction error, relative to O_B. It can be noticed from Fig. 6.19b that the Y and Z components of the prediction error are inferior to ± 2 mm, whereas the maximum absolute value of the X

(depth, relative to the robot's base reference frame, shown in Fig. 3.11a) component of the displacement error is 9.67mm. As expected, the relatively higher error of the prediction error obtained in the Z-component of MF_7 , as expressed with CamR, transfers to the X-component of the "pre-contact" point of $Ding_2$, relative to O_B , due to the approximately parallelism between the Z axis of CamR and the X axis of O_B .

As a result, the increase in the prediction error is very small, when compared to the accuracy obtained with respect to CamR, supporting the proposed prediction approach. Moreover, the approximately 10mm magnitude of the depth component of the prediction error, with respect to O_B , can easily be accommodated by the robotic system that uses an LED-pointer or spray gun for marking, since the latter does not need to physically touch the moving panel.

Figure 6.20 illustrates the block diagram of the on-line robotic operation for spraying deformation defects. Starting from the "home" position, the robot is guided to the "pre-contact" point of the first defect, within a "look-and-move" procedure [63]. At each frame extraction of the pose and motion estimator, the transformation $Q_{Tool/Base}$ is computed and appended to a data buffer, which will contain all the $Q_{Tool/Base}$ transformations, computed throughout the entire motion cycle of the automotive panel.

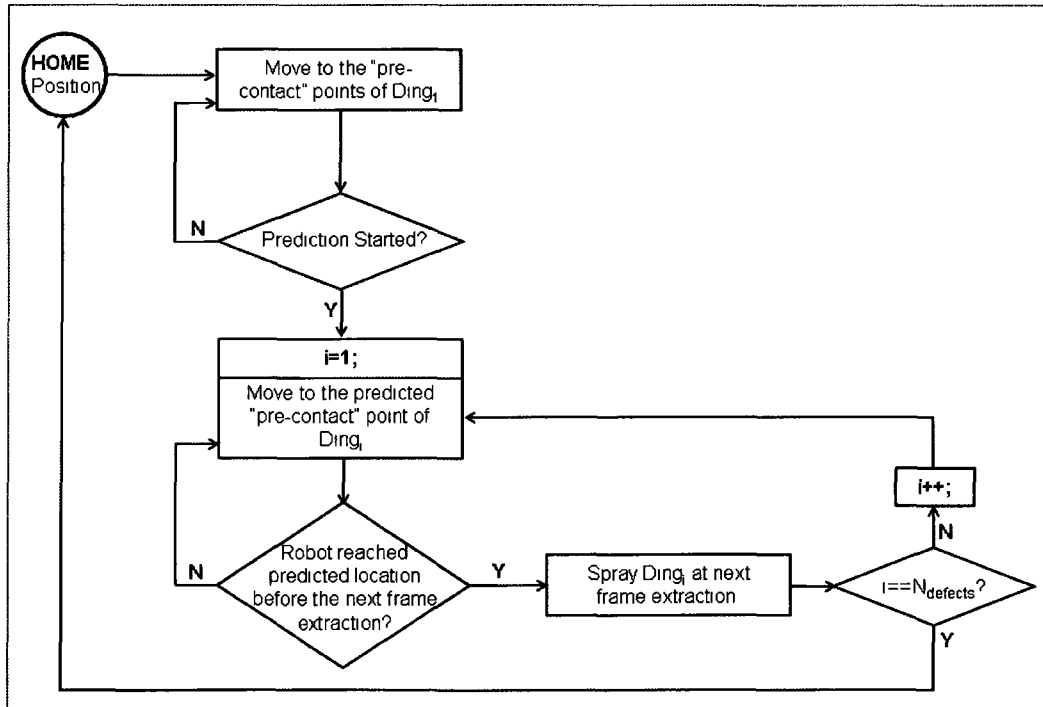


Fig. 6.20. Flowchart for the on-line defects spraying operation.

As soon as the robot reached the "pre-contact" point of the first defect, as given by the first transformation stored in the data buffer, the visual servoing data for its next

movement is also extracted from the same buffer. In the cases in which two new transformations have been appended to the buffer, during the interval in which the robot reached the previous location, the most recent one is extracted for the next robotic movement, for time efficiency purposes. As shown in Fig. 6.20, this procedure is repeated until the “one-step” prediction process is started.

As soon as the prediction procedure is triggered (according to the prediction interval, which is [6s, 38s] for the current settings involving the selected speed for the sled system), the manipulator robot is driven to the predicted location of the “pre-contact” point of the first defect, as opposed to the visual servoing data prior to the prediction process, in which the robot remained slightly shifted with respect to the “pre-contact” point of the first defect, due to the continuous motion of the panel, and the lack of motion prediction.

In the situation in which the robot reaches the imposed location before the next frame extraction, it remains in that position in order to perform the spraying operation. Thus, the triggering of the next frame extraction coincides with the moment in which the spraying occurs, and the defects’ index (i , in Fig. 6.20) is incremented. Conversely, if in the interval in which the robot reaches the predicted location, a new set of frames is grabbed by the stereoscopic sensor, the robot waits for the next prediction data (for the current defect) and the same course of actions is performed, as illustrated in Fig. 6.20. This procedure is executed until all the deformation defects have been sprayed. Then, the manipulator robot is driven to its “home” position and the pose and motion estimation process is stopped. Obviously, it is assumed that the prediction interval is long enough such that the entire set of deformations can be marked before the panels reaches out of the camera’s field of view.

Similar to the settings introduced in Section 6.3, three scenarios are considered, for the final tests, based on the pose of the sled system with respect to the acquisition sensor. Some samples of images grabbed during these scenarios are shown in Fig. 6.21.

One can see that the projected light falls over the dings’ region in all cases. The angles triplet $(\theta_0, \theta_1, \theta_2)$, shown in Fig.6.11b, quantize the rotation applied to the sled system around the Y_0 axis, in order to obtain these three final testing scenarios. The same velocity for the sled system was maintained, while the speed of the manipulator robot was set to 40% of its maximum magnitude. Under these settings the total duration of the robotic spraying procedure, as measured from the beginning of the prediction cycle, took $t_{\text{mark_door}}=10\text{s}$, for each of the three final scenarios. Subtracting $t_{\text{mark_door}}$ from

the total prediction duration $t_{\text{prediction}}=32\text{s}$, which is only determined by the length and speed of the conveyor used, and the field of view of the cameras, a reserve of $t_{\text{extra}}=22\text{s}$ remains available for performing extra deformation spraying steps, until the end of the complete motion prediction cycle. Therefore a larger set of deformations can be sprayed, during the entire motion sequence of the panel on the relatively short length (54 cm) of the sled.

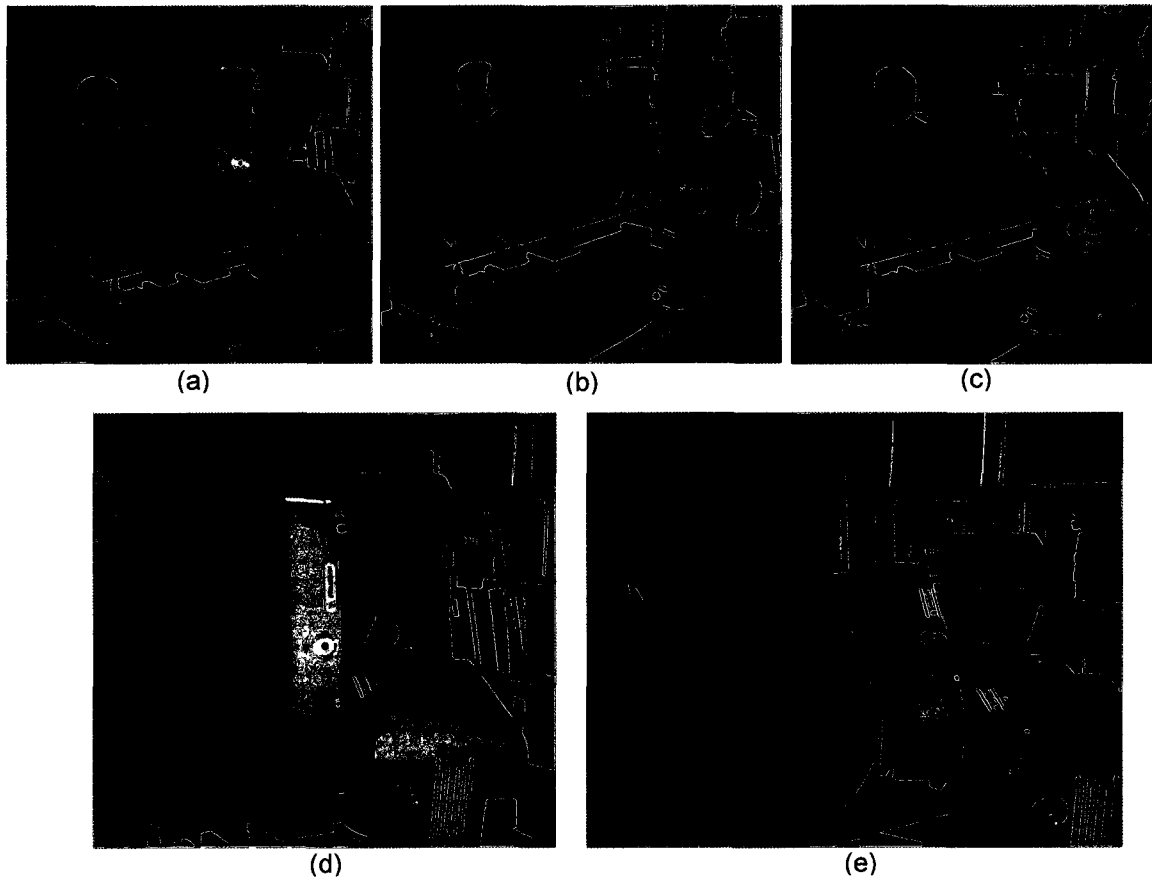


Fig. 6.21. Images acquired during experimentation with the on-line robotic spraying operation: (a) (Scenario₁, Ding₁), (b) (Scenario₁, Ding₂), (c) (Scenario₁, Ding₃), (d) (Scenario₂, Ding₁), (e) (Scenario₂, Ding₂).

In order to validate the generality level of the proposed on-line robotic spraying operation, the procedure characterized by the flowchart illustrated in Fig. 6.20, was also applied on the fender model, shown in Fig. 3.1d. Figure 6.22a displays the deformation regions, which were identified from a visual inspection of the fender. The relative distances between the “contact” points of the deformations and the closest MF, were measured off-line, according to the reference frame of the SSPME, CamR. Then, during the first 3D reconstruction procedure, performed by the pose and motion estimator, the locations of the “contact” points were determined, with respect to CamR, by using the

off-line relative measurements. Subsequently, the location of these “contact” points during the tracking of the fender, was updated based on the rigid transformation, characterizing the motion exerted by the panel between the previously and currently extracted frames.

Without loss of generality, only the scenario characterized by a motion pattern exhibited in a plane approximately perpendicular to the principal axis of the acquisition sensor, was considered. Figures 6.22b-d show some samples acquired during the on-line robotic spraying operation on the fender. Due to the elevated light absorbance exhibited by the dark-blue fender model, only a reserve of 15cm was subtracted from the “contact” points of the deformations, for better visualization during the robotic spraying operation.

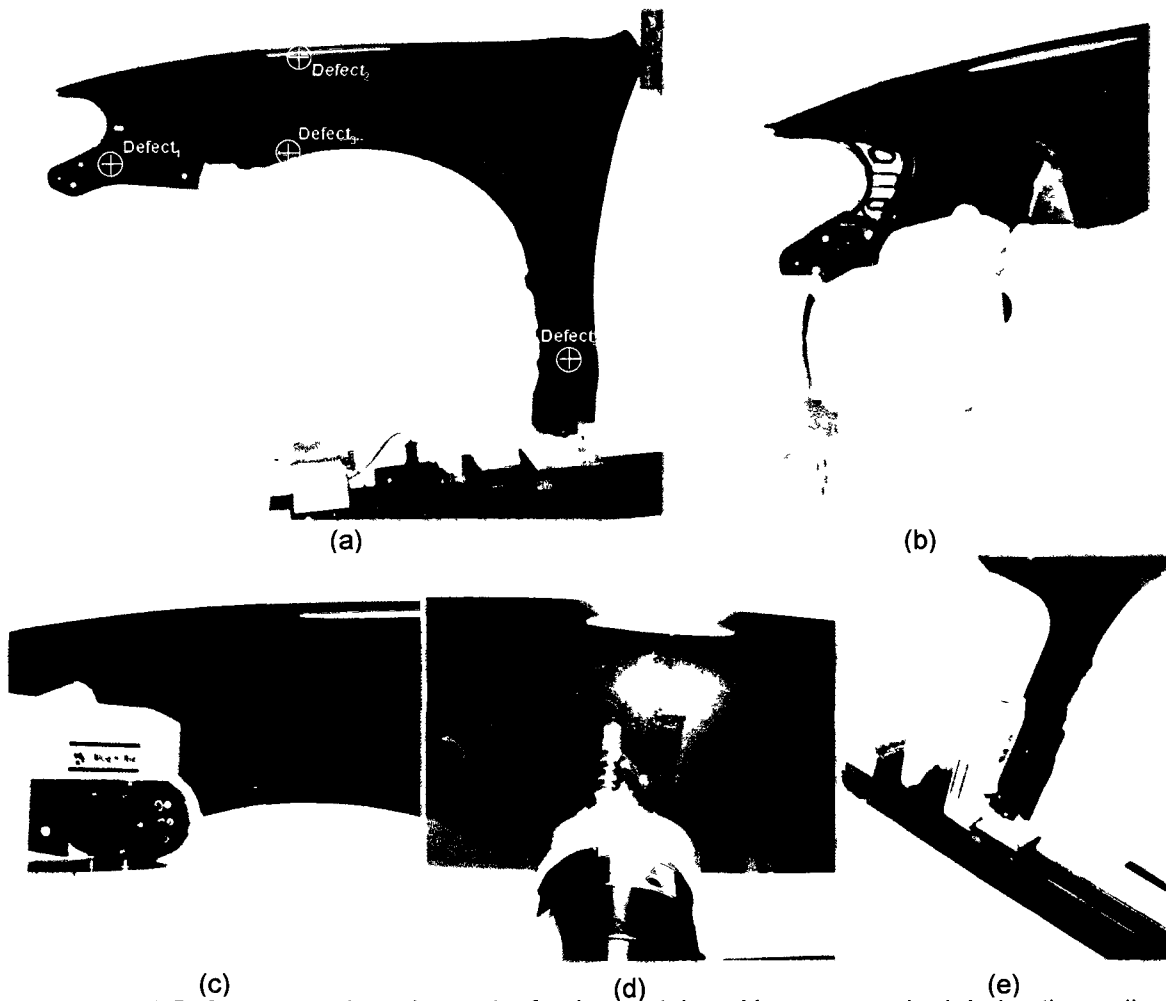


Fig. 6.22. (a) Defects pre-selected over the fender model, and images acquired during the on-line robotic spraying operation: (b) Defect₁, (c) Defect₂, (d) Defect₃, (e) Defect₄.

From Fig. 6.22b-d, one can see that the projected light falls over all the four defects' regions. By using the same magnitude for the speed of the robotic arm, the total

marking operation took $t_{\text{mark_fender}}=15\text{s}$, which left a reserve of $t_{\text{extra}}=17\text{s}$ available for performing extra deformation spraying steps, until the end of the complete motion prediction cycle.

The maximum number of deformation defects that can be marked is inversely proportional to the relative distance between the defects over the surface of the automotive panel. For example, in cases where the defects share a similar relative distance as for the current dings of the car door model ($\text{distance}(\text{Ding}_1, \text{Ding}_2) \approx 29\text{cm}$, $\text{distance}(\text{Ding}_2, \text{Ding}_3) \approx 31\text{cm}$, as computed between the “contact” points), a maximum number of $N_{\text{max}}^{\text{defects}} = 9$ deformations can be sprayed, with the current prototype. However, if a different speed is set for the manipulator robot or a “two-step” motion prediction procedure is chosen, or a higher frame extraction rate is selected, $N_{\text{max}}^{\text{defects}}$ can be increased. As mentioned in the first part of this section, the pose and motion estimator can operate up to an update rate of $\bar{f}_u \cong 5\text{Hz}$. Therefore, the increase in the frame extraction rate coupled with a “two (three)-step” motion prediction can substantially augment $N_{\text{max}}^{\text{defects}}$, if needed.

Defining the maximum number of defects that can be marked must take into account several factors, such as the length and speed of the assembly line, the computational complexity of the pose and motion estimator and the inter-calibration procedure between the SSPME and the SLS, the selected velocity of the manipulator robot, the distance between the robot’s “home” position and the first defect, and finally, the relative distances between the defects. Therefore, as soon as an upper bound is defined for the maximum number of possibly detectable deformation defects over a single panel, a trade-off is to be performed between all the above mentioned factors in order to obtain the most efficient solution.

6.5. Chapter Summary

This chapter concludes the description of the proposed autonomous robotic system for marking undesired deformations over moving automotive panels characterized by very few apparent visual features. The first part of this chapter discussed the integration of the robotic marking system within the autonomous defects detection and marking station, which is performed by means of two inter-calibration procedures. These inter-calibrations relate the pose and motion estimator with the

robotic system and the structured light sensor employed in the deformation defects detection phase.

The second part of this chapter is dedicated to the problem of robotic stamping of deformation defects in a static environment. However, with the purpose of validating the pose and motion estimations, this procedure was repeated at different positions and orientations of the automotive panel with respect to the acquisition sensor.

Finally, in the last part of this chapter, the challenging task of robotic interaction on moving panels for marking of deformations is described. In order to account for the small depth variations present in the attitude of the robotic tool with respect to the moving panel, while preserving the same visual servoing architecture, a modified robotic prototype was proposed, that is equipped with a spray gun, for the final experimental validation. The spraying procedure is then imitated in the laboratory by using an LED-pointer as an end-effector, in order to provide a reliable insight into the surface region that will be sprayed during the on-line robotic marking process.

The accuracy achieved in the two validation phases (on the static and moving panel) converts the proposed robotic solution into a viable alternative to perform fully automated region marking of deformations over large surfaces and for substantial volumes of production.

Chapter 7. Conclusions

7.1. Summary

This thesis addressed the challenging problem of robotic interaction with moving objects that exhibit few visual features on their surface in the context of an application for marking of surface deformation defects under quality control settings in the automotive industry. Chapter 2 reviewed state-of-the-art solutions for the two major components of the proposed robotic application, respectively related to the pose and motion estimation problem, and to the actual robotic interaction with moving bodies.

The complete design process for a quality control application involving robotic interaction with moving objects under passive visual guidance was examined in Chapter 3. Starting with the hardware analysis of the proposed quality control system, a standard stereo-vision configuration, appropriately positioned in the scene, was selected, in order to provide a consistent insight into all the motion patterns that the rigid body can exhibit on an automotive assembly line. The complete experimental cell for automated deformation defects detection and marking, also introduced in Chapter 3, defined the prerequisites for the high-level description of the software components of the application.

The initial part of Chapter 4 was dedicated to an innovative strategy for macro-features selection and rigid body detection. The macro-features pre-selection by a process engineer, that needs to be performed only during system's configuration, represents the central element of the proposed feature-based pose and motion estimation solution, which overcomes the lack of strong visual features. The knowledge provided to the system by these macro-features was successfully integrated in the object detection module, allowing for faster inspection of manufactured goods.

In the second part of Chapter 4, the selected methodologies for the feature extraction, tracking and matching processes were detailed. For feature extraction, several techniques, including the classical SIFT, Shi and Tomasi, Harris, Noble and SUSAN feature detectors were experimentally evaluated. A correlated stability-robustness empirical measure was proposed to compare the results provided by all the selected feature extractors. The Shi and Tomasi corner detector obtained the highest score and was therefore selected for integration in the final pose and motion estimation prototype. For feature tracking, the integration of the pyramidal Lucas-Kanade (LK) tracker into the pose and motion estimation solution was introduced. The severe tracking errors exhibited by the

LK tracker in the cases in which the brightness constancy assumption is not validated, were also discussed. Situations such as occlusions, photometric variations, or temporary appearance of different entities (workers) in the inspected scene were explicitly considered as they represent prerequisites for the robust tracking of an object in a typical industrial setting.

Chapter 5 detailed the methodologies for 3D reconstruction, motion estimation and robust supervision, completing the description of the proposed pose and motion estimation solution. The original component of the proposed approach consists of an innovative module for the pose and motion estimator that takes the form of a supervisory layer. Its objective is to ensure that the robotic marking station receives “timely-mannered”, accurate and fault-tolerant visual servoing data. The developed supervisory system embeds two sub-systems, related respectively to the coarse and fine-level supervision processes. The former provides an alternative to the sensitivity of the pyramidal LK tracker to occlusions, changes in illumination or the temporary appearance of different entities in the monitored scene. The latter addresses the problem of drift accumulation, which is affecting the pyramidal LK tracker when long sequences are processed.

The merit of the proposed supervisory layer is related to the manner in which the knowledge provided by the topological structure of the macro-features in 2D/3D is successfully embedded in the monitoring and validation processes. As a result, despite the limited amount of information available about the macro-features extracted from a texture-less automotive body panel, the system provides a robust supervisory alternative to the problem of estimating the pose and motion of an automotive panel, although the motion happens in a complex environment with other moving entities and sporadic occlusions. As demonstrated by the conducted experimentations, the pose and motion estimation system is able to operate properly even in cases where the assembly line contains curved sections, as frequently imposed by the limited space in factory settings. Moreover, the proposed approach handles the supervision process from a software perspective, which results in eliminating the need for accurate 3D CAD models of the inspected parts, controlled backgrounds or extra vision sensors.

The first part of Chapter 6 was dedicated to the integration of the defects detection system with the robotic tracking and marking station, which was accomplished by means of two inter-calibrations. The former related the stereoscopic sensor used for pose and motion estimation with the robot’s base. The latter linked the surface imager, which is the central component of the defects detection system developed separately from this thesis, with the

stereo-vision sensor used for estimating the pose and motion of the automotive panel. The proposed calibration strategy was designed to ensure the necessary flexibility for implementing the defects detection system and the robotic tracking and marking stage either into a single work cell or in different stations located subsequently along the assembly line.

The last part of Chapter 6 described the problem of robotic interaction with the automotive panel from two different perspectives. The first prototype provided a solution for autonomous marking of deformation defects using a stamping tool in a static environment. The second prototype handled the task of robotic interaction with a moving panel. An *analysis of the robotic gripper's attitude with respect to the moving part throughout the considered motion cycles* was performed. Based on this analysis, a robotic marking prototype relying on a spray gun end-effector was privileged for the final experiments. In this way, the marking of deformations was successfully performed, without the need for the manipulator robot to physically touch the moving automotive part, which is perfectly adequate for the application considered in this work.

The accuracy achieved with the two marking validation approaches (on static and moving panels) demonstrated the suitability and viability of the proposed visually-guided robotic solution for performing fully automated region marking of deformations over large surfaces and for substantial volumes of production in the automotive industry.

7.2. Contributions

This thesis proposes an integrated autonomous robotic system for tracking and marking surface deformation defects on moving panels, for quality control in the automotive industry. The main contributions of this work are:

- The design of a viable solution to the problem of estimating pose and motion on industrial objects that exhibit very few contrasting features over their surface, and its integration into a real robotic work cell. The only prerequisite to the proposed solution is a minimum amount of knowledge about a limited set of macro-features, which are manually pre-selected by the installation engineer when configuring the robotic tracking and marking station. Such conditions are easily met in quality control applications happening in industrial settings.

- The development of an original solution for object detection, which is more computationally efficient than the approaches relying on segmentation and color histograms. The central component of the object detection module resides in the manner in which the knowledge provided by the macro-features is integrated in the designed validation gates.
- The development of an innovative supervisory layer, integrated in the pose and motion estimation solution, with the objective of providing robustness, rapidity and fault-tolerance to visual servoing in the robotic station. Through the interventions of the supervisory layer, the limitations of classic feature extraction, tracking and matching processes are solved from a software perspective, without the requirement of exact 3D CAD models of the inspected parts, or additional proximity sensing technologies.
- The integration of the passive visual acquisition system with the robotic arm, coupled with the development of an original inter-calibration technique between the defects detection stage and the stereo-vision system used for estimating the pose and motion of the automotive panel. While relying only on the limited amount of knowledge provided by the macro-features, the latter inter-calibration technique embeds the generality level needed for the implementation of the defects detection stage, and the robotic tracking and marking system, into two different stations. Moreover, this inter-calibration approach is performed on-line and does not require a specific target object, as it adapts itself to the macro-features selected over the surface of the inspected automotive panels.
- The development of two defects marking prototypes, the former related to a stamping operation on a static object, and the second handling the marking procedure on a moving panel, by using an LED-pointer, to mimick a spraying operation. In both cases, the extensive experimental validation demonstrated that acceptable accuracy is achieved for marking of deformation areas over static and moving automotive panels, while relying only on the visual servoing data provided by a simple passive stereo-vision sensor.
- The design of a correlated stability-robustness empirical measure, tested on a real setting, for determining the most suitable corner detector, out of five of

the most popular feature detectors, for the feature extraction component of the developed pose and motion estimator of moving panels exhibiting few distinctive features.

Parts of this work have been published in [72] and [75].

7.3. Future Work

Although the accuracy achieved by the robotic tracking and marking system is acceptable for the integration in a real industrial setting for quality control in the automotive industry, a series of future work directions can be identified for all the components of the proposed application.

Regarding the pose and motion estimator, the time reserve, computed as the difference between the frame extraction period and the necessary visual processing time, allows for the addition of supplementary enhancement stages in the supervisory layer, in order to further refine the pose and motion estimations. Therefore, the procedure embedded in the fine-level supervisory system, that is responsible for attaching a signature to the 3D macro-features' region, can be supplemented with a methodology under which a corner model would be attached to each individual macro-feature. The addition of different validation gates in the structure of the supervisory layer would result in a more robust supervision of the feature tracking and matching processes, which play an essential role in the precision of the pose and motion estimations.

Another improvement to the pose and motion estimation would be to complement the limited number of macro-features with other keypoints that can be extracted from the structure of the rigid body. The increased set of tracked features can be easily integrated into the fine and coarse-level supervisory procedures, for robust tracking and matching. In this way, the least-squares procedure selected for estimating the motion that the rigid body have exhibited between two subsequent frames would be applied on a larger set of data, which would have beneficial effects on the accuracy of the computed estimations.

Regarding the integration between the defects detection station and the robotic tracking and marking system, different feature matching methodologies can be explored for computing the macro-features' correspondences between the two stations. This could represent an interesting extension to eliminate the need for the installation engineer to perform a manual pre-selection of keypoints during configuration of the defects detection station.

Finally, the prototype of the on-line robotic marking station, which relies on a spray gun end-effector, can also embed a more robust motion prediction procedure, building upon a more robust kinematic model. For example, by selecting a Kalman filter for the prediction methodology, a higher generality level can be achieved by the station responsible for the robotic interaction with the moving object, allowing for a more varied set of motion patterns exhibited by the automotive panel.

References

- [1] C. Harris and M. Stephens, "A combined corner and edge detector", *Proceedings of the 4th Alvey Vision Conference*, UK, 1988, pp. 147-151.
- [2] Computer Vision CITS4240, Course Notes, "Edge detection", *University of Western Australia*, [on-line], <http://undergraduate.csse.uwa.edu.au/units/CITS4240/Lectures>
- [3] A. Noble, "Description of image surfaces", PhD Thesis, *Department of Engineering Science, Oxford University*, 1989.
- [4] M. Brady and H. Wang, "Vision for mobile robots", *Philosophical Transactions: Biological Sciences*, vol. 337, no. 1281, 1992, pp. 341-350.
- [5] J. Canny, "A computational approach to edge detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, 1986, pp. 679-714.
- [6] J. Shi and C. Tomasi, "Good features to track", *Proceedings of the 9th IEEE Conference on Computer Vision and Pattern Recognition*, USA, 1994, pp. 593-600.
- [7] Y. -P. Tan, S. R. Kulkarni and P. J. Ramadge, "Extracting good features for motion estimation", *Proceedings of the IEEE International Conference on Image Processing*, Switzerland, 1996, pp. 117-120.
- [8] S. Smith and J. M. Brady, "SUSAN – A new approach to low level image processing", *International Journal of Computer Vision*, vol. 23, no. 1, 1997, pp. 45-78.
- [9] D. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, vol. 60, no. 2, 2004, pp. 91-110.
- [10] S. Wei and G. Bao-long, "A robust object detecting and tracking method", *Proceedings of the 5th International Conference on Fuzzy Systems and Knowledge Discovery*, China, 2008, pp. 121-125.
- [11] H. Bay, T. Tuytelaars and L. Van Gool, "SURF: Speeded Up Robust Features", *Proceedings of the 9th European Conference on Computer Vision*, Austria, 2006, pp. 404-471.

- [12] É. Vincent, "On feature point matching, in the calibrated and uncalibrated contexts, between widely and narrowly separated images", *PhD Thesis*, University of Ottawa, Canada, 2004.
- [13] X. C. He and N. H. C. Yung, "Curvature scale space corner detector with adaptive threshold and dynamic region of support", *Proceedings of the 17th International Conference on Pattern Recognition*, UK, 2004, pp. 791-794.
- [14] T. Kaneko and O. Hori, "Feature selection for reliable tracking using template matching", *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, USA, 2003, vol. 1, pp. 796-802.
- [15] J. -H. Park, K. -W. Park, S. -H. Baeg and M. -H. Baeg, "Reliable feature point detection and object pose estimation using photometric quasi-invariant SIFT", *Proceedings of the International Conference on Control, Automation and Systems*, Korea, 2008, pp. 2142-2147.
- [16] E. Delp and O. Mitchell, "Image compression using block truncation coding", *IEEE Transactions on Communication*, vol. 27, no. 9, 1979, pp. 1335-1342.
- [17] M. Liu, C. Wu and Y. Zhang, "Multi-resolution optical flow tracking algorithm based on multi-scale Harris corner points feature", *Proceedings of the 2008 Chinese Control and Decision Conference*, China, 2008, pp. 5287-5291.
- [18] Z. Živković and F. van der Heijden, "Improving the selection of feature points for tracking", *Pattern Analysis and Applications*, vol. 7, no.2, 2004, pp. 144-150.
- [19] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision", *Proceedings of the 1981 DARPA Image Understanding Workshop*, 1981, pp. 121-130.
- [20] C. Tomasi and T. Kanade, "Detection and tracking of point features", *Carnegie Melon University Technical Report*, CMU-CS-91-132, USA, 1991.
- [21] G. Bradsky and A. Kaehler, "Learning OpenCV – Computer Vision with the OpenCV Library", *O'Reilly*, USA, 2008.
- [22] J. -Y. Bouguet, "Pyramidal implementation of the Lucas Kanade feature tracker – description of the algorithm", [on-line], http://robots.stanford.edu/cs223b04/algo_tracking.pdf

- [23] H. Jeon, J. Jeong, J. Bang and C. Hwang, "The efficient features for tracking", *Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence*, USA, 2008, pp. 241-244.
- [24] J. Weber and J. Malik, "Robust computation of optical flow in a multi-scale differential framework", *International Journal of Computer Vision*, vol. 14, 1995, pp. 67-81.
- [25] H. V. Le, G. Seetharaman and B. Zavidovique, "A multiscale technique for optical flow computation using piecewise affine approximation", *Proceedings of the 2003 IEEE International Workshop on Computer Architectures for Machine Perception*, USA, 2003, pp. 221-230.
- [26] A. Fusiello, E. Truco, T. Tommasini, and V. Roberto, "Improving feature tracking with robust statistics", *Pattern Analysis & Applications Journal*, vol. 2, 1999, pp. 312-320.
- [27] U. Uenohara and T. Kanade, "Geometric invariants for verification in 3-D object tracking", *Proceedings of the 1996 IEEE International Conference on Intelligent Robots and Systems*, Japan, 1996, pp. 785-790.
- [28] E. Barrett and P. Payton, "General methods for determining projective invariants in imagery", *CVGIP: Image Understanding*, vol. 53, 1991, pp. 46-65.
- [29] H. T. Tsui, Z. Y. Zhang and S. H. Kong, "Feature tracking from an image sequence using geometric invariants", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Puerto Rico, 1997, pp. 244-249.
- [30] Y. Yoon, G. N. DeSouza and A. C. Kak, "Real-time tracking and pose estimation for industrial objects using geometric features", *Proceedings of the 2003 IEEE Conference on Robotics & Automation*, Taiwan, 2003, pp. 3473-3478.
- [31] A. Shokurov, A. Khropov and D. Ivanov, "Feature tracking in images and videos", *Proceedings of the 2003 Graphicon International Conference*, Russia, 2003, pp. 177-179.
- [32] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography", *Communications of the ACM*, vol. 24, 1981, pp. 381-395.

- [33] A. Milella, B. Nardelli, D. Di Paola and G. Ciciirelli, "Robust feature detection and matching for vehicle localization in uncharted environments", *Proceedings of the 3rd International Workshop on Intelligent Robots and Systems*, USA, 2009, pp. 11-16.
- [34] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces", *International Journal of Computer Vision*, vol. 13, no. 2, 1994, pp. 119-152.
- [35] C. Olson, L. H. Matthies, M. Schoppers and M. W. Maimone, "Robust stereo ego-motion for long distance navigation", *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, USA, 2000, pp. 453-458.
- [36] H. Jin, P. Favaro and S. Soatto, "Real-time feature tracking and outlier rejection with changes in illumination", *Proceedings of the 8th IEEE International Conference on Computer Vision*, Canada, 2001, pp. 684-689.
- [37] Y.-S. Yao and R. Chellappa, "Tracking a dynamic set of feature points", *IEEE Transactions on Image Processing*, vol. 4, no. 10, 1995, pp. 1382-1395.
- [38] P. Fua, "Combining stereo and monocular information to compute dense depth maps and preserve depth discontinuities", *Proceedings of the International Joint Conference on Artificial Intelligence*, 1991, pp. 1292-1298.
- [39] R. Deriche, Z. Zhang, Q. -T. Luong and O. Faugeras, "Robust recovery of the epipolar geometry for an uncalibrated stereo rig", *Lecture Notes in Computer Science*, vol. 800, 1994, pp. 567-576.
- [40] T. Darrell, "A radial cumulative similarity transform for robust image correspondence", *Proceedings of the Conference on Computer Vision and Pattern Recognition*, USA, 1998, pp. 656-662.
- [41] Z. -N. Li and G. Hu, "Analysis of disparity gradient based cooperative stereo", *IEEE Transactions on Image Processing*, vol. 5, no. 11, 1996, pp. 1493-1506.
- [42] R. Laganière, S. Gilbert and G. Roth, "Robust object pose estimation from feature-based stereo", *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 4, 2006, pp. 1270-1280.
- [43] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two perspectives", *Nature Publishing Group Journal*, vol. 293, 1981, pp. 133-135.

- [44] R. Hartley, "In defence of the 8-point algorithm", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, 1997, pp. 580-593.
- [45] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision", Cambridge University Press, UK, 2000.
- [46] K. Kanatani and Y. Sugaya, "Compact fundamental matrix computation", *Proceedings of the 3rd Pacific-Rim Symposium on Image and Video Technology*, Japan, 2009, pp. 179-190.
- [47] P. Beardsley, P. Torr and A. Zisserman, "3D Model acquisition from extended image sequences", *Proceedings of the 4th European Conference on Computer Vision*, UK, 1996, pp. 683-695.
- [48] A. Whitehead and G. Roth, "The Projective Vision Toolkit", *Proceedings of the Modeling and Simulation Conference*, USA, 2000, pp. 204-209.
- [49] P. H. S. Torr, "A structure and motion toolkit in Matlab", *Microsoft Research Technical Report MSR-TR-2002-56*, 2002.
- [50] J. Mulligan, "Fast calibrated stereo vision for manipulation", *International Journal of Real-Time Imaging*, vol. 3, no. 5, 1997, pp. 331-341.
- [51] J. Weng, T. S. Huang and N. Ahuja, "Motion and structure from two perspective views: Algorithms, error analysis and error estimation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 5, 1989, pp. 451-476.
- [52] M. Pollefeys, "Visual 3D modeling from images", *Tutorial Notes, University of North Carolina*, 2002, [on-line], <http://www.cs.unc.edu/~marc/tutorial/>
- [53] P. H. S. Torr and A. Zisserman, "Feature based methods for structure and motion estimation", *Springer-Verlag Lecture Notes on Computer Science*, 2000, pp. 278-294.
- [54] T. Huang and A. Netravali, "Motion and structure from feature correspondences: A review", *Proceedings of the IEEE*, vol. 82, no.2, 1994, pp. 244-249.
- [55] R. Holt and A. Netravali, "Number of solutions for motion and structure from multiple frame correspondence", *International Journal of Computer Vision*, vol. 23, no. 1, 1997, pp. 5-15.

- [56] H. Shariat and K. Price, "Motion estimation with more than two frames", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 5, 1990, pp. 417-434.
- [57] K. S. Arun, T. S. Huang and S. D. Blostein, "Least-squares fitting of two 3-D point sets", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, 1987, pp. 698-700.
- [58] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, 1991, pp. 376-380.
- [59] J. M. Kuang and M. Liu, "A novel explicit pose estimation algorithm based on Euclidean geometry", *Proceedings of the 2007 International Conference on Information Acquisition*, Korea, 2007, pp. 247-252.
- [60] A. Milella and R. Siegwart, "Stereo-based ego-motion estimation using pixel tracking and iterative closest point", *IEEE International Conference on Computer Vision Systems*, USA, 2006, pp. 21-25.
- [61] R. Y. Tsai, "A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses", *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, 1987, pp. 323-344.
- [62] Q. V. Le and A. Y. Ng, "Joint calibration of multiple sensors", *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, USA, 2009, pp. 3651-3658.
- [63] S. Hutchinson, G. D. Hager and P. I. Corke, "A tutorial on visual servo control", *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, 1996, pp. 651-670.
- [64] G. N. DeSouza and A. Kak, "A subsumptive, hierarchical and distributed vision-based architecture for smart robotics", *IEEE Transactions on Systems, Man and Cybernetics*, Part B, vol. 34, no. 5, 2004, pp. 1988-2002.
- [65] Y. Yoon, J. Park and A. C. Kak, "A heterogeneous distributed visual servoing system for real-time robotic assembly applications", *Proceedings of the International Conference on Robotics and Automation*, USA, 2006, pp. 4422-4424.

- [66] T. Chang, T. Hong, M. Shneier, G. Holguin, J. Park and R. D. Eastman, "Dynamic 6DOF metrology for evaluating a visual servoing system", *Proceedings of the 2008 Performance Metrics for Intelligent Systems Workshop*, USA, 2008, pp. 173-180.
- [67] L. Davis, D. DeMenthon, T. Bestul, S. Ziavras, H. V. Srinivasan, M. Siddalingaiah and D. Harwood, "Robot acting on moving bodies (RAMBO): Interaction with tumbling objects", *Technical Report – Center for Automation Research – University of Maryland*, [on-line], http://www.cfar.umd.edu/~daniel/daniel_papersfordownload/RAMBOPaper.pdf
- [68] A. Namiki and M. Ishikawa, "Robotic catching using a direct mapping from visual information to motor command", *Proceedings of the 2003 IEEE International Conference on Robotics & Automation*, Taiwan, 2003, pp. 2400–2405.
- [69] R. C. Luo, R. E. Mullen Jr. and D. E. Wessell, "An adaptive robotic tracking system using optical flow", *Proceedings of the IEEE International Conference on Robotics and Automation*, 1988, pp. 568-573.
- [70] A. Vázquez, A. Adán and F. Molina, "3D Vision system for robotic interaction in moving scenes", *Proceedings of the IEEE International Symposium on Intelligent Signal Processing*, China, 2007, pp. 1-6.
- [71] A. Adán, F. Molina, A. S, Vázquez and L. Morena, "3D Feature tracking using a dynamic structured light system", *Proceedings of the 2nd Canadian Conference on Computer and Robot Vision*, Canada, 2005, pp. 168-175.
- [72] V. Borsu and P. Payeur, "Pose and motion estimation of a moving rigid body with few features", *Proceedings of the IEEE International Workshop on Robotic and Sensor Environments*, Italy, 2009, pp. 116-121.
- [73] A. Yogeswaran and P. Payeur, "Feature extraction from point clouds for automated detection of deformations on automotive body parts", *Proceedings of the IEEE International Workshop on Robotic and Sensor Environments*, Italy, 2009, pp. 122-127.
- [74] A. Boyer, P. Curtis and P. Payeur, "3D Modeling from multiple views with integrated registration and data fusion", *Proceedings of the 6th Canadian Conference on Computer and Robot Vision*, Canada, 2009, pp. 252-259.

- [75] V. Borsu, A. Yogeswaran and P. Payeur, "Automated surface deformations detection and marking on automotive body panels", *Proceedings of the 6th IEEE International Conference on Automation Science and Engineering*, Canada, 2010, pp. 551-556.
- [76] S. Bériault, "Multi-camera system design, calibration and 3D reconstruction for markerless motion capture", *Master Thesis*, University of Ottawa, 2008.
- [77] Z. Zhang, "A flexible new technique for camera calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, 2000, pp. 1330-1334.
- [78] X. Chen, J. Davis and P. Slusallek, "Wide area camera calibration using virtual calibration objects", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, USA, 2000, pp. 520-527.
- [79] I. Ihrke, L. Ahrengerg and M. Magnor, "External camera calibration for synchronized multi-video systems", *Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, Czech Republic, 2004, pp. 537-544.
- [80] A. Yogeswaran, "3D Surface maps analysis for automated detection and classification of deformations", *Master Thesis*, University of Ottawa, 2010.
- [81] A. Boyer, "Adaptive structured light imaging for 3D reconstruction and autonomous robotic exploration", *Master Thesis*, University of Ottawa, 2009.
- [82] CRS Robotics Corporation, *F3 Robot System User Guide*, Burlington, ON, 2000.
- [83] CRS Robotics Corporation, *Track User Guide*, Burlington, ON, 1999.
- [84] CRS Robotics Corporation, *C500C Controller User Guide*, Burlington, ON, 2000.
- [85] CRS Robotics Corporation, *RAPL-3 Language Reference Guide*, Burlington, ON, 1999.
- [86] T. Brox, B. Rosenhahn, J. Gall and D. Cremers, "Combined region and motion-based 3D tracking of rigid and articulated objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, 2010, pp. 402-415.

- [87] M. E. Munich, P. Pirjanian, E. Di Bernardo, L. Goncalves, N. Karlsson and D. Lowe, "SIFT-ing through features with ViPR", *IEEE Robotics & Automation Magazine*, 2006, pp. 72-77.
- [88] SIFT Demo Program, Version 4, July 2005, [on-line], <http://www.cs.ubc.ca/~lowe/keypoints>
- [89] MathWorks Inc., [on-line], www.mathworks.com
- [90] V. Garcia, "Keypoint extraction", 2007, [on-line], <http://www.mathworks.com/matlabcentral/fileexchange/17894-keypoint-extraction>
- [91] H. Bulthoff, J. Little and T. Poggio, "A parallel algorithm for real-time computation of optical flow", *Nature Journal*, vol. 337, no. 6207, 1989, pp. 549-553.