Utilisation de Moments Photometriques en Asservissement visuel

Manikandan Bakthavatchalam

2015

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

École doctorale Matisse

présentée par

Manikandan BAKTHAVATCHALAM

préparée à l'unité de recherche 6074 - IRISA
Institut de Recherche en Informatique et Systèmes Aléatoire
Université de Rennes 1

Utilisation of Photometric Moments in Visual Servoing

Thèse soutenue à Rennes
le 17 mars 2015

devant le jury composé de :

Éric MARCHAND
Professeur, Université de Rennes 1 / Président du jury

El Mustapha MOUADDIB
Professeur, Université Picardie Jules Verne / Rapporteur

Youcef MEZOUAR
Professeur, Institut Français de Méchanique Avancée / Rapporteur

Philippe MARTINET
Professeur, Ecole Centrale de Nantes / Examinateur

Omar TAHRI
Chercheur Habilité, Université de Bourgogne / Examinateur

François CHAUMETTE
Directeur de recherche Inria / Directeur de thèse

Acknowledgements

To my jury

I offer my thanks to all the thesis committee members for having accepted to be in my jury. I thank Dr.Eric Marchand for having honored me by being the President of the jury. I thank the reviewers for their expert review of the manuscript and for their useful comments and suggestions. I thank all the jury members for attending my defense and for putting forth their questions and concerns about the work. I also thank them for giving me useful pointers on the various ideas listed in future work.

First of all, I have to thank my advisor François Chaumette for having accepted me as his PhD student and for giving me this nice opportunity to work under his supervision. I developed a great admiration for his traits as a scientist, especially his razor-sharp intellect, his keen eye for details, his insistence on scientific rigor and his timely advices, for all of which I have to thank him sincerely. I thank him for his detailed and useful comments when reviewing my papers which helped me to improve myself each time. Without his advice and intervention at several important phases of this PhD, it would have been very hard to take this thesis to completion. I also thank him for his patience and giving me autonomy even at times when progress slowed down. Finally, I also thank him for his understanding and authorizing my extended stay abroad in a difficult personal situation.

I offer my immense thanks to Dr.Omar Tahri for our pleasant and fruitful collaboration. He played a very important role in changing my point of view about the subject and research in general. He warmly received me at the Institute of Systems and Robotics in Coimbra for my research internship which became a wonderful turning point for this PhD. I thank him for all his advice and encouragement. This was a great boost to my self-confidence and came at a time when it was absolutely required. I thank him for giving me the opportunity to explore topics beyond my thesis scope. I thank him sincerely for his patient exchange of ideas with me and for giving me the freedom to contact him on skype anytime.

To my teachers

I am in debt to all my teachers who taught me so far. I especially thank Prof.Wisama Khalil from Ecole Centrale de Nantes sincerely for being such a great and inspiring teacher. With his individual attention to each student taking his robotics courses, he has nurtured generations of roboticists. I am one student among the umpteen others who benefited from his teaching and guidance. I sincerely thank all the teachers and academic staff at

Nantes and Warsaw University of Technology from whom I have gathered knowledge and inspiration.

I offer my thanks to all the teachers at Pondicherry who taught during my undergraduate Engineering years. I thank Dr.Sundaramurthy for his unique teaching style, hands-on guidance in projects and investment of his personal finances and energy which inspired the whole batch of students. We have taken some precious lessons from him for life. I also sincerely extend my gratitude to all my school teachers and mentors at Petit Seminaire Higher Secondary School. The discipline and fundamentals that you instilled in me are indispensable for the pursuit of anything worth having in life.

To lagadic and INRIA people

First of all, I thank INRIA for having provided me the CORSIS grant, for the excellent infrastructure and environment to pursue my doctoral studies. I thank Eric Marchand for being kind enough to permit me to go his office without an appointment and for clearing my doubts, be it in computer vision or visual servoing. I thank Paolo Robuffo Giordano for his useful suggestions during the defense presentation rehearsals. I thank Marie for her warm smiles and for emphasizing the *bisou* culture in office. I thank Alexandre Krupa for his jokes and friendly attitude.

I thank Fabien Spindler for helping me organize my software and showing me some good practices. I completely enjoyed working with Fabien. I thank him sincerely for his help and professionalism, for teaching me things that will be useful perhaps throughout my career.

I equally thank all the team members of lagadic including but not only Suman Raj Bista, Nicolas Cazy and Noel Meriaux who attended the rehearsals and for giving their comments. I thank Nicolas Cazy for the nice defense poster. I thank Filip Novotny for offering ready help with understanding the code base and for his sarcastic humour. I thank Olivier Kermorgant for sparking interest in python with his lovely script. I thank Celine Teuliere for always having offered to help even after she left lagadic. I thank Laurent Coutard, Bertrand Delabarre, Aly Magassouba, Pierre Martin, Riccardo Spica and Hideaki Uchiyama for maintaining a cordial atmosphere that prevailed in the different offices. I thank also Riccardo Spica for his nice company and honest feedback during ICRA presentation rehearsals. I thank Aly Magassouba for his joviality and loud laughs even at my poor jokes. I have to also thank Aly for the great swimming sessions, for bringing out the best sportsman in me and also for being the first African to learn and play cricket in Rennes. I thank Pierre Chatelain, especially for his french translation of the synthesis of the thesis. I thank Giovanni, the always-cheerful Aurelien, Caroline Nadeau, Amaury Dame and Romain Tallonneau for their camaraderie. All of you were a part and reason of the wonderful period in INRIA.

I thank all the team assistants, Angelique, Nathalie, Celine and Helene who were so helpful with the tedious administrative issues. I thank Helene for her smooth organisation of activities related to the defense. I thank Celine Gharsalli for being always-ready-to-help and for her sincere follow-up during troublesome issues while abroad on the internship. She was very friendly and always passing by our offices to enquire how we are doing, shared a joke or two and tolerated all the funny jokes we pulled on her.

I thank Charlotte-Peyrat Vaganay for bringing the *Explorateur* grant to my attention

and for her help in preparing the dossier. I thank Sophie Viaud for her kindness and prompt response for administrative help. I also thank the staff at *CMI*, especially Emilie Gesnys for the clean and smooth progress of matters with the prefecture. I thank the members at the atelier (equipe SIC) including Yves Prunault, Philippe, Jerome Priour and Bruno Derien for their rapid support whenever there were some issues with the laptop or software. I thank Alain Crenn for his nice video editing work of the thesis defense.

Friends and family

I thank my friend Nalanmaharadjane Doure who stood with me from the first day I landed in France and his family for having treated me as a member of their own family. I thank Jawahar Ravee Nithianandam for his positive and reassuring words which kept me firm on the track. I thank my friends Rafal and Marta Pijewski, Wong Sheng Chao and Franciscek Szych for their friendship and care over these years. I thank Marine for attending my defense and for the cute cup with Doc embossed on it. I thank Pedro Miraldo, Maria Chelimsky, Cecilia, Diego Resende Faria and Fernanda for their kindness and for making my stay in Portugal pleasant and wonderful.

I thank Aswin Sridharan for his wonderful company typically during the late-night stays in the office, his dinner invites and his fantastic sense of humour. I thank Surya Narayanan for the great time we shared together at Rennes, in office, at home and in the gym. I thank Mihir Jain enlightening me on nutrition aspects. I thank Raghavendran Balu for sharing his experience, helping me in interviews and for our career-related discussions. I thank Arjun Suresh for accepting to do the entrepreneurship course and for doing the project with me together. I thank Deepak for his endless invites to his home in Cesson, though I could honor only a few of them and for his patient swimming lessons. I thank everyone who came for the card game saturday sessions in my apartment which was pure fun. I thank Sendhamizhan for his friendship and who was a perfect flatmate who stayed with me a couple of weeks. I also thank Mythri Alle who helped me with her advice and cooking during her very short stay with me. It is the people who make up a place and not the place itself as such. I had a great time with Yogesh Karpate, Saravana Kumar, Karthik, Arnab Sinha, Pawan Reddy, Shubra Singh, Zehra Khan, Yogesh Chandrasekaran, Dinesh, Mounir Saggai, Mangai, Premila, Apoorba, Linus Paulin, Charles Beromeo, Maria Monica, Nupur Mittal, Aruna Rajaraman, Shruti Sridhar, Ramakanth Singal Reddy and in fact several others. I will need many more pages even if i were to describe in a few words about my happy experience with each and everyone of you.

I thank my brother Arvind who was a silent pillar of support throughout and for his help in handling financial and family-related issues. I thank both my parents for having kept us as the first priority of their lives. I thank my mom for her constant encouragement and emphasis on the importance of academic as well as extracurricular pursuits, right from when I was a kid. I thank my affectionate and lovely grandmother who was a great deal of support during my preparations to leave abroad, despite her own troublesome issues.

I feel grateful that I had the chance to meet so many kind and good-natured people in this journey. I thank all those who I would have inadvertently missed out when writing this.

Contents

Introduction

The growth in robotics, computer vision and sensor technologies in the past decade has been tremendous. Robotics especially has brought an impact in several areas including industrial manufacturing, automotive systems, medical robots for surgery, for treatment of autistic children and personal assistants. The sense of vision is indispensable for almost all of the above applications, be it fully autonomous or semi-autonomous. Auto-industry majors have already currently commercial semi-autonomous driving functions integrated in modern vehicles and aspire to get fully autonomous in the future. These systems, known as advanced driver assistance systems (ADAS) invariably consist of one or several cameras which perform the following functions: they alert the driver when departing from lanes, perform adaptive cruise control and detect pedestrians, animals and children in the vicinity of the vehicle (blind-spots which humans cannot perceive anyway) and above all ensure safety [Trivedi 07]. In manufacturing, machine and robot vision enable object recognition, handling of parts, orienting them for subsequent processing in the assembly line and inspection of defects based on image measurements.

In humans, vision is one of the most dominant senses, the loss of which could be devastating. It is fascinating to just think about the number of functions that are completely based on human visual perception: visual-motor integration for eye-in-hand and eye-body coordination, visual-auditory integration, visual memory by which we remember and recall visual information so efficiently, visual closure by which we discern a person or subject of interest from an image even if major portions of it are hidden. The admiration for our sense of vision increases manifold when we see that solutions to several problems that computer scientists are trying to achieve is carried out by our visual system in such an efficient and accurate manner. These include depth perception, fixation on and smooth tracking of a moving object (pursuit), change focus on a moving object accurately when it moves farther or closer [Palmer 99]. Not only in humans but also in a simple biological organism such as the fruit fly (*Drosophila Melanogaster*), the visual system and neurons (tiny fraction in relation to a human) are responsible for color perception, motion detection and pattern recognition [Paulk 13]. In the same vein, though a robot can be endowed with different perceptual faculties, visual perception is a key factor that will enable the accomplishment of useful and in fact critical functions in the real-world.

This thesis broadly can be classified in the domain of vision-based control, also known as visual servoing. It is the application of computer vision towards enabling robotics applications. What exactly is visual servoing? It is a closed-loop feedback control method which uses feedback information provided by visual sensors. This visual sensor is most

often a monocular camera, the cheapest, easily available and widely used sensing apparatus for robotic perception. Using this feedback information, it seeks to stabilise actuated mechanical systems to a desired equilibrium state. How does a visual servoing algorithm derive this feedback information? In order to derive this information, a traditional visual servoing algorithm extracts some geometric primitives, for example points, straight lines or ellipses from the image of the scene observed by a camera [Chaumette 90]. Let us assume that these primitives can be readily parametrized by a vector $\mathbf{x}$. How does it exploit these primitives to control an actuated system? The first is the pose-based approach (PBVS) where $\mathbf{x}$ is used to reconstruct the camera pose and then compute control inputs to regulate that pose to the desired one. The drawback of this approach is that the camera pose is sensitive to calibration and measurement errors. In the second approach, the variations in $\mathbf{x}$ are linked to the camera motion through a special matrix known as the interaction matrix. Since $\mathbf{x}$ is directly obtained from the image, this approach is known as image-based visual servoing (IBVS). Unlike PBVS, IBVS is regarded attractive since it is robust to robot modelling and camera calibration errors. Despite investigation since several decades, this interdisciplinary field continues to present interesting and unsolved research problems.

Motivation and objectives

It is well-known that conventional visual servoing methods employ as visual features measurements from geometric primitives present in the image. There are some drawbacks to this approach:

- The first caveat is that the imaged scene should contain specific 3D elements (set of 3D points or environment with lines or a cylinder, as the case may be) in order that the corresponding geometric primitives can be extracted from their respective image projections. This dependence on the presence of specific geometric elements is a restriction towards their applicability. For example, mobile robots which are expected to work in outdoor or hazardous environments cannot rely on the presence of such elements in the scene.

- It is clear that image processing, visual tracking and subsequent extraction steps are mandatory in the visual servoing pipeline of such methods. Many researches pointed out this dependence on visual tracking procedures is a handicap to the expansion of visual servoing methods [Collewet 08a] [Tatsambon 13]. That is why research to develop visual servoing methods which do not require image processing and feature extraction is of high importance as well as practical relevance.

- The image gets reduced to a set of geometric primitives after the image processing and feature extraction steps. Most importantly, the intensity information that is available in the image is not utilized at all and is lost.

Differing from the geometric-primitive based approaches, the objective of this thesis is to develop visual servoing schemes that exploit the rich intensity information available in the image. That being said, the idea of using the image intensity by itself is not new. The pure photometric visual servoing method [Collewet 11] uses the intensity directly as a visual feature. In this case, the cost function is highly non-linear. This non-linearity stems from

adoption of the iterative non-linear Levenberg-Marquardt(LM) method to optimize this cost function. The complexity introduced limits the convergence domain of this method. At the same time, the redundancy introduced with the choice of the intensity feature ensured excellent accuracies upon convergence. Intensity is made use of in [Malis 00] and [Silveira 12] but not as a visual feature. The visual features in these methods are based on projective geometric measures extracted from the images. These methods are only *less-geometric* in comparison to methods based on feature tracking but they suffer from problems of similar nature inherent to those methods. In specific, they use complicated photo-geometric image registration algorithms which unfortunately is a visual tracking problem by itself. In contrast, while trying to attain the objective of exploiting intensities, the visual servoing control itself is kept separated from the influence of image processing. That is why this work refrains from using any explicit feature extraction, image matching or visual tracking steps. Visual features based on geometric moments offered decoupled control laws and better behaviour when compared to geometric features [Chaumette 04], [Tahri 05a]. However, they were restrictive in the sense that a tracked or well-segmented object be available in the image. In [Chaumette 04], it was explicitly pointed out thus : *"the problem that occurs in practice lies in parts of the image that appear or disappear from one image to the next one, which makes the moments non-differentiable on all of the image plane". In this thesis, we tackle this problem from the modelling point of view and remove the existing restrictions imposed by the moments.* The envisaged goal is to have a set of visual features based on photometric moments for visual servoing with a 6-dof fully actuated system, and one which is not affected by the appearance and disappearance of parts of the image.

Contributions of this thesis

With the above problems and objectives in perspective, the following is a list of the contributions from this thesis. The contributions are discussed with ample background details and placed in context to enable a better understanding. The publications that resulted from the contributions (also ones awaiting review) are given at the end of this manuscript in page 221.

1. **Tunable Visual Features** In IBVS, the simultaneous control of the translational and rotational motions is a difficult problem. The difficulty arises in part from the fact that the effects in the image due to a translation along x (resp. y) is difficult to distinguish from the effects caused by a rotation around y (resp. x). Design and selection of suitable visual features is an open problem and remains unsolved even for moment-based visual servoing schemes. The visual features to control the rotational motions around x and y especially has been difficult. In the state-of-the-art, visual features based on moment invariants have been presented [Chaumette 04] [Tahri 05a]. The features introduced to control the aforementioned rotations were dependent on the shape of the object present in the image. This problem was solved when [Tamtsia 13b] proposed a fresh approach by defining the moments with respect to virtual image points called shift points. The moments thus obtained were called shifted moments and invariants were later derived from them to control the x and y rotational motions. In this thesis, we propose improvements to the shifted moments method. We define the moments with respect to specially chosen points

called tunable shift points [Bakthavatchalam 14]. This is a generalization of the shifted moments and to the best of our knowledge, this is a new direction with open-ended possibilities. These possibilities can be used to ensure that the visual servoing is optimal with respect to specific criteria. Some metrics have been proposed for selecting these shift points. We have highlighted how the default choice as done in [Tamtsia 13b] is not optimal in all cases and is especially suboptimal with respect to conditioning of the system. We believe this will inspire research towards optimal visual servoing. The proposed techniques find immediate applications in moments-based visual servoing. They are applicable to moments computed from discrete points and binary moments. It should be noted that the tunable shift points are still virtual and are not tracked in the image. So, they are applicable in the case of photometric moments as well, which will be discussed next.

2. **Photometric Moments** Different from existing approaches, the novelty in this thesis lies in the manner in which intensity in the image is utilized. Instead of directly using the intensity as a visual feature or visually tracking geometric primitives as done in the state-of-the-art, we propose to capture the intensities in the form of image moments. The idea of using image moments by itself is not new however. Image moments were shown to be attractive for IBVS because of their decoupling properties and the resulting characteristics were demonstrated to be better than geometric features like points [Chaumette 04] [Tahri 05a]. The moments we propose, computed from the intensity, are termed *Photometric Moments*. An important contribution is the development of the interaction matrix for the proposed photometric moments in analytical form. The developed interaction matrix is more general than the interaction matrix for image moments in the existing state-of-the-art. It is important to note that the obtained result is consistent with respect to previously established results although a completely different derivation has been used to obtain it. This analytical form is important because it helps to analyse the invariance properties and eventually the design of decoupled control laws. Most importantly, it is essential to conduct stability analyses. A set of visual features based on photometric moments are then developed to control the 6DOF of a robot, when there is no appearance or disappearance of scene portions from the camera image. We have shown that *it is possible to perform visual servoing using photometric moments without recourse to any visual tracking, image matching or segmentation procedures* [Bakthavatchalam 13]. We show that a better behaviour and a convergence domain larger than for the pure photometric method can be obtained. Photometric moments embrace grayscale images and are not restricted by the availability of well-segmented binary images or discrete set of points in the scene. This advancement enables marching towards the goal of applying visual servoing algorithms in complex scenes.

3. **Weighted Photometric Moments** Visual servoing based on photometric data is of great interest since it does not necessitate any image processing or visual tracking steps. A vital issue in such methods is the change in the image resulting from the appearance and disappearance of portions of the scene from the camera field-of-view during the visual servoing. This is a very important practical issue that can disturb the stability and convergence of the control scheme. In this the-

sis, we propose a spatial weighting scheme to counter this problem. A general model for photometric moments that allows for inclusion of spatial weights is presented. In this work, a custom weighting function is chosen and specific properties of its derivative are exploited in order to develop the interaction matrix in analytical form [Bakthavatchalam 15]. This facilitates the analysis of the effect of the included weighting function on the invariance properties of the moments. With the findings of this study, visual features that exhibit invariance properties could be developed. Experiments to compare the weighted photometric moments with non-weighted moments and pure photometric visual servoing confirm the effectiveness of the improved model in handling the problem of extraneous regions.

Manuscript Organization

Chapter 1 begins with a fairly detailed introduction to background in visual servoing. The task function formalism which forms the basis of visual servoing and its application to vision sensors is first discussed. A thorough discussion of stability analysis is made. This chapter is a brief survey of the currently existing visual servoing methods. The traditional geometric primitives-based IBVS methods are first presented. This is followed by presentation of current state of the art in visual servoing methods that use intensity.

In Chapter 2, we begin with the state of the art in image moments and gradually move towards *tunable visual features*. Two approaches to modelling the temporal variation of image moments, the first which uses Green's theorem and the other based on the method of Gâteaux derivatives are explained. Then, visual features for the control of translational motions are introduced. The theory of moment invariants is touched upon briefly followed by a description of the general procedure to obtain moment functions invariant to planar rotations. Existing visual features for the control of x and y rotational dof are introduced and their limitations are explained. The idea of shifted moments is then discussed. The development of a set of visual features based on the shifted moments is shown. Then, the idea of tunable visual features is proposed. Several metrics for their selection are discussed and how the visual servo can be optimized according to specific criteria is discussed. We discuss aspects related to our motivation behind the selection of a visual feature from several existing ones. The proposed ideas are validated using binary moments with symmetrical and non-symmetrical objects. The results are discussed and conclusions are outlined.

Chapter 3 is dedicated to *Photometric Moments*, the central theme of this thesis. The formulation of photometric moments is introduced and detailed mathematical developments necessary for obtaining the interaction matrix are first presented. The differences with respect to Kernel-based visual servoing are outlined. Visual features are then composed using invariants derived from the proposed photometric moments. The analytical model of the interaction matrix of photometric moments is first validated. Then, visual servoing with visual features from the proposed photometric moments is compared to the pure luminance feature, when large displacements are involved. Visual servoing results with different configurations of the interaction matrix are shown. Experimental results

from robotic platforms are then presented. Finally, conclusions are drawn based on these results.

In Chapter 4, our approach to solve the practical problem of uncommon image portions that occurs in image moments-based methods is described. An improved modelling scheme which allows for inclusion of spatial weights in the photometric image moments formulation is discussed. We delve into details about selection of a weighting function which will result in closed-form expressions of the interaction matrix components. Two different choices of the weighting function are made and the resulting interaction matrix is analysed. Then, the effect of the weighting function and loss of the invariance properties of the most commonly used visual features is described. Validations are performed in simulation to validate the correctness of the modelling steps. Comparisons experiments with non-weighted moments and pure photometric visual servoing method are provided. This is followed by a presentation of the conducted experimental validations.

Finally, general conclusions and perspectives for future work are given in Chapter 5.

Chapter 1

Visual Servoing - State of the Art

1.1 Visual Servoing

Visual Servoing is a multidisciplinary subject at the intersection of computer vision, robotics and automatic control. It is a field being investigated since several decades by researchers from different perspectives. An extensive survey of the field is a non-trivial task. The first objective of this chapter is of course to inform the reader of all background knowledge and updation of the state of art in visual servoing methods. A fairly detailed introduction to visual servoing is attempted. More emphasis is laid on intensity-based visual servoing methods than traditional feature-based ones.

Visual Servoing refers to the family of closed loop control techniques to control the degrees of freedom of an actuated system with visual feedback [Weiss 87] [Feddema 89] [Hutchinson 96] [Chaumette 06]. These actuated systems include, but are not limited to industrial robots, mobile robots and underactuated systems like quadrotors. It has a wide realm of applications in many robotics domains.

Problem Statement Given a continuous stream of images $\mathbf{I}(t)$ acquired from a vision sensor mounted on a system (with n actuated DOF), how do we determine the control inputs $\mathbf{u}(t) \in \mathbb{R}^n$ to be applied to the system in order to achieve some desired task?

Visual servoing is potentially applicable to different types of actuated systems and for different control objectives. Let us consider a typical 6 DOF serial robot with a camera mounted on its end-effector (eye-in-hand configuration). A very simple task is to move the end-effector from pose A to another pose B in the workspace. Another is to move the robot end-effector along a specific trajectory in space $\mathbf{X}(t)$. The visual servoing problem consists in calculating the actuation velocities $\mathbf{u}(t) = \begin{bmatrix}\boldsymbol{\nu}(t) & \boldsymbol{\omega}(t)\end{bmatrix}$ to be applied to the robot in order to perform the above tasks. To achieve this, the relation between the robot movements and the changes of the visual features in the image needs to be established.

Next, let us consider a more complex actuated system. Assume we have a quadrotor robot equipped with a ground-facing camera and other instrumentation like inertial sensors for attitude measurement. The desired task is to make the quadrotor centre of

mass stabilize at a specific position in space and to observe some landmark on the ground from a specific attitude. This task is in $\mathrm{SE}(3)$ with 6DOF. A quadrotor system has four propeller-mounted rotors whose angular speeds are regulated to control platform movement. The control inputs are the total thrust and net moment generated in space due to the rotor aerodynamics $\mathbf{u}(t) = \left[\Gamma_\Sigma(t), \tau_x(t), \tau_y(t), \tau_z(t)\right]$. Since there are only 4 actuated DOF, the quadrotor is an underactuated system and its control is non-trivial. For the quadrotor system, the visual servoing then consists in relating the underactuated dynamics of this system to images streamed from the onboard camera. If this relation is established, the control inputs $\mathbf{u}$ needed to stabilize the machine can be calculated. Given $\mathbf{u}$ for the system, the rotor speeds can be computed by a simple linear transformation [Mahony 12].

Control Architecture For the visual control of robotic systems, several control architectures exist depending upon where the vision system intervenes in the robot control loop [Sanderson 80]. The two-layered architecture (See Fig.1.1) is commonly adopted where the vision control loop decides the desired control set-points for the joint controller [Corke 93]. The first layer is the low level control loop which runs at a much higher frequency (100 Hz) than the vision control loop (25 Hz). The visual servoing loop specifies an intermediate system state to be attained which is realised by the low level controller. This layered architecture disentangles the visual control from the low level control in the sense that they can be studied as two separate systems. The performance of the robot control system is affected by the delays introduced by the vision control loop [Corke 96]. So, it is preferable to design visual servoing control laws such that they do not introduce significant delays in the robot control system.

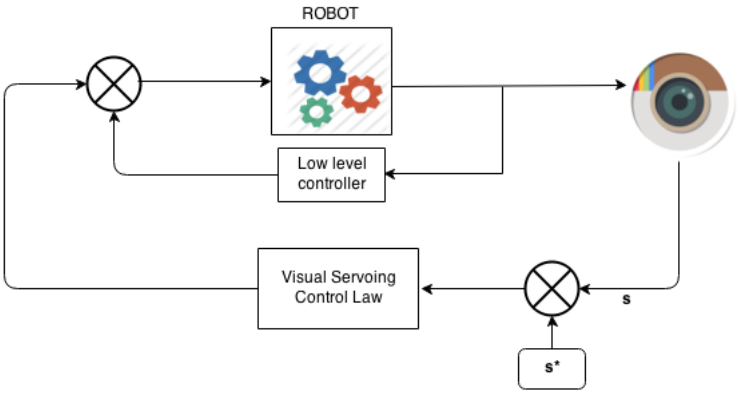


Figure 1.1 – Double layered control architecture in visual servoing

A detailed schematic of this vision control loop can be seen in Fig.1.2. It is imperative to understand the task function formalism in order to appreciate the theory and working principles behind visual servoing.

1.1.1 Principle of Task Function Regulation

The core principle behind task function regulation is that a robot can be controlled to perform useful actions by specifying those actions in the form of a function $\mathbf{e}(\mathbf{q}, t)$ to

be regulated over a temporal horizon $[0, T]$ [Samson 91b]. The task function $\mathbf{e}(\mathbf{q}, t)$ is defined in terms of some measure f of the state of the robot joints $\mathbf{q}(t)$ and an exclusive dependence on time t and importantly belongs to the differentiability class C^2.

$$\mathbf{e}(\mathbf{q}, t) = f(\mathbf{q}(t)) - f(\mathbf{q}^*(t)) \tag{1.1}$$

If $\mathbf{e}$ is C^2, we can obtain the first two derivatives of (1.1).

$$\dot{\mathbf{e}}(\mathbf{q}, t) = \frac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}} \dot{\mathbf{q}} + \frac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial t} \tag{1.2}$$

In Equation (1.2), the highlighted term $\dfrac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}}$ is known as the task Jacobian. It is this relation that links the changes in the joint space to changes in the task function space. The second term reflects all changes in $\mathbf{e}$ with respect to time, resulting from changes other than in $\mathbf{q}$. Thus, Equation (1.2) is completely general. Differentiating the same using the chain rule and applying the Schwartz theorem for partial derivatives, we get

$$\ddot{\mathbf{e}}(\mathbf{q}, t) = \frac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}} \ddot{\mathbf{q}} + \dot{\mathbf{q}} \frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}} \right) \dot{\mathbf{q}}^\top + 2 \left(\frac{\partial^2 \mathbf{e}(\mathbf{q}, t)}{\partial t \, \partial \mathbf{q}} \right) \dot{\mathbf{q}} + \frac{\partial^2 \mathbf{e}(\mathbf{q}, t)}{\partial t^2} \tag{1.3}$$

In (1.3), the Hessian of the task function $\dfrac{\partial}{\partial \mathbf{q}} \left(\dfrac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}} \right)$ is highlighted.

Task Function Admissibility A task function $\mathbf{e}(\mathbf{q}, t)$ is said to be admissible if it satisfies the following conditions:

1. There exists an ideal trajectory $\mathbf{q}_d(t)$ which is of differentiability class C^2 such that $\forall t \in \{0, T\}, \mathbf{e}(\mathbf{q}_d(t), t) = 0$ and the initial condition $\mathbf{q}_d(0) = \mathbf{q}_0$ is available.

2. A critical condition that needs to be respected is the regularity of the mapping defined by the task Jacobian till the control time horizon. In other words, the mapping defined by $\frac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}}$ should always be invertible (non-singular) $\forall t \in \{0, T\}$. This was originally defined as the regularity condition.

If these conditions are satisfied, then the control problem is well-posed and the task function is admissible. Then, a controller can be designed to regulate this admissible task function. Provided the stability condition (in Section 1.1.2) is ensured, this controller results in successful execution of the desired task. The task function can for instance be defined directly in the Cartesian space

$$\mathbf{e}(\mathbf{q}, t) = \boldsymbol{\xi}(\mathbf{q}, t) - \boldsymbol{\xi}^*(t) \tag{1.4}$$

where $\boldsymbol{\xi}^*(t)$ is the trajectory of appropriate parametrizations in Cartesian space. It belongs to the special euclidean group $\boldsymbol{\xi} \in \mathbb{SE}(3)$, consisting of a rotation matrix $\mathbf{R} \in \mathbb{SO}(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$, and is typically termed the Euclidean pose.The pose $\boldsymbol{\xi}$ depends on the nature of the actuated system. For example, for a planar 3R manipulator, the end-effector configuration can be defined by 3 quantities, (x, y) its position in the plane and an angle α

specifying its rotation in the plane. So, $\boldsymbol{\xi} = (x, y, \alpha) \in \mathbb{R}^3$. In general, $\boldsymbol{\xi} \in \mathbb{R}^6$ enabling to specify the position and orientation of a point can be with respect to a Cartesian reference frame. The term $\dfrac{\partial \mathbf{e}(\mathbf{q}, t)}{\partial \mathbf{q}} = \mathbf{J}(\mathbf{q})$ is known as the robot Jacobian. So far, task functions were expressed in the joint space or operational space of the robot. Also, the task function regulation framework is general and is applicable to sensing modalities other than vision. An application of this framework to proximity sensors can be found in [Samson 91a]. We will see next how this formalism has been applied to the case of vision sensors.

Application to Vision When exteroceptive sensors are mounted on the robot, the task function needs to be expressed in the appropriate sensor space. This formalism was applied to the case of vision sensor in [Espiau 92].

The core idea behind visual servoing is specification of robotic task as a functional of measures from the visual space.

$$\mathbf{e}(\mathbf{q}, t) = \mathbf{C}\,\mathbf{s}(\boldsymbol{\xi}(\mathbf{q}), t) - \mathbf{s}^*(\boldsymbol{\xi}^*, t) \tag{1.5}$$

where $\mathbf{s}$ is a set of visual features build from camera images obtained at specific robot pose denoted $\boldsymbol{\xi}$. Then, $\mathbf{s}^*$ denotes the set of visual features computed from the image learnt at the desired robot pose. This learnt image will be referred to as the desired image. The matrix $\mathbf{C}$ takes care of the redundant case (when the visual features are more than

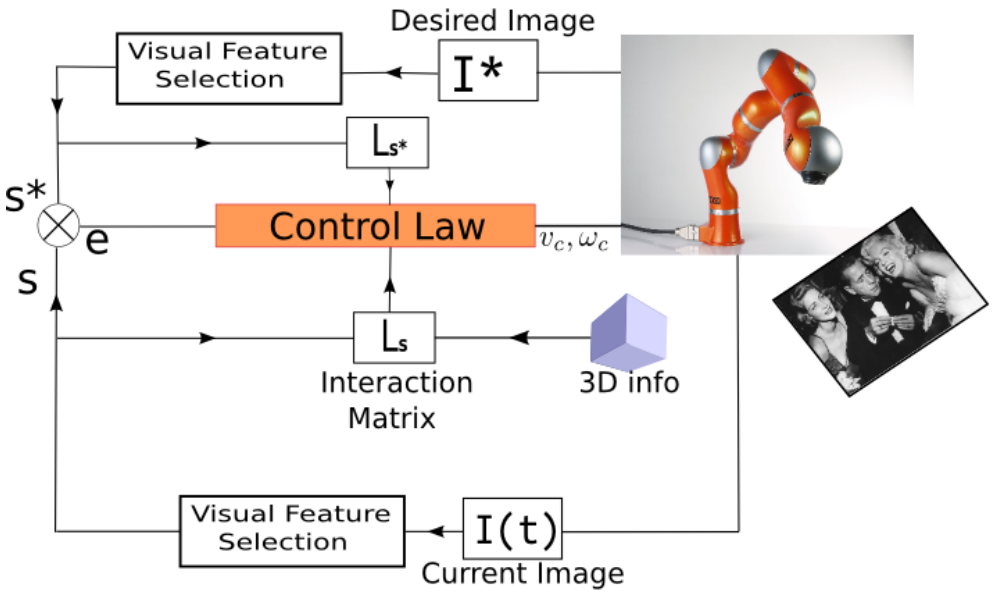


Figure 1.2 – Schematic of a robotic visual servoing system

the degrees of freedom) so that the task function regularity condition is not violated. The task function framework implicitly assumes the ability to obtain $\mathbf{e}(t)$ from the acquired

images at all times during the control. In other words, visibility of the image features is assumed. Essentially, a set of visual features $\mathbf{s}$ is the mapping

$$\mathbf{s} : \boldsymbol{\xi} \rightarrow \mathbb{R}^k \tag{1.6}$$

This mapping is composed of a sequence of mappings as explained in [Espiau 92]. The goal of a visual servoing task consists in controlling the system by regulation of the error in the visual features between the observed and desired values $\mathbf{s}(t) - \mathbf{s}^*$ to 0. In visual servoing, it is essential to link the variations $\dot{\mathbf{s}}(t)$ in the visual features to the motions of the robot. This operation is what is essentially carried out by the task function in Eq (1.2). When the camera is mounted on the end-effector, $\dot{\mathbf{s}}(t)$ is linked to the camera motion whose spatial velocity is $\mathbf{v}_c = \begin{bmatrix}\boldsymbol{\nu}_c(t) & \boldsymbol{\omega}_c(t)\end{bmatrix}$, $\boldsymbol{\nu}_c(t)$ and $\boldsymbol{\omega}_c(t)$ being the instantaneous linear and rotational velocities expressed in the camera frame. Assuming that the visual features at the desired pose are constant during the control, we can simply take $\dot{\mathbf{e}} = \dot{\mathbf{s}}$. Further, from (1.2), we can write

$$\dot{\mathbf{s}} = \mathbf{J}_\mathbf{s}\,\dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t} = \frac{\partial \mathbf{s}}{\partial \boldsymbol{\xi}}\,\dot{\boldsymbol{\xi}} + \frac{\partial \mathbf{s}}{\partial t} \tag{1.7}$$

where $\mathbf{J}_s$ is the task Jacobian. The second term $\dfrac{\partial \mathbf{s}}{\partial t}$ denotes the change in features resulting from motion of the target object and may also include changes in feature values induced by potential disturbances acting on the system.

Let us consider two Cartesian reference frames: $\mathcal{F}_c$ attached to the camera and $\mathcal{F}_n$ attached to the end-effector. Let $\boldsymbol{\xi}_n$ and $\boldsymbol{\xi}_c$ refer to the configuration of the robot expressed in the end-effector $\mathcal{F}_n$ and camera reference frames $\mathcal{F}_c$ respectively. In the eye-in-hand configuration, the first term in Equation (1.7) can be decomposed and the equation be rewritten in the following manner

$$\dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \boldsymbol{\xi}_c}\,\frac{\partial \boldsymbol{\xi}_c}{\partial \boldsymbol{\xi}_n}\,\frac{\partial \boldsymbol{\xi}_n}{\partial \mathbf{q}}\,\dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t} \tag{1.8}$$

More pleasingly, we can write [Chaumette 07]

$$\dot{\mathbf{s}} = \mathbf{L}_\mathbf{s}\ {}^c\mathbf{V}_n\,\mathbf{J}_\xi(\mathbf{q})\,\dot{\mathbf{q}} + \frac{\partial \mathbf{s}}{\partial t} \tag{1.9}$$

where

$$\mathbf{J}_\mathbf{s} = \mathbf{L}_\mathbf{s}\ {}^c\mathbf{V}_n\,\mathbf{J}_\xi(\mathbf{q}) \tag{1.10}$$

Examining each term in $\mathbf{J}_\mathbf{s}$ from the right,

- $\dfrac{\partial \boldsymbol{\xi}_n}{\partial \mathbf{q}} = \mathbf{J}_\xi(\mathbf{q})$ is the robot Jacobian that links the end-effector spatial motion to the changes in the joint space. This matrix satisfies the relation $\mathbf{J}_\xi\dot{\mathbf{q}} = \dot{\boldsymbol{\xi}}_n$, where $\boldsymbol{\xi}_n$ is the robot robot end-effector configuration expressed in its own reference frame $\mathcal{F}_n$. In this thesis, we assume the knowledge of this Jacobian for each robot.

- The robot Jacobian is expressed in the end-effector frame and the visual features and interaction matrix are defined in the camera coordinate frame $\mathcal{F}_c$. The matrix

$\frac{\partial \boldsymbol{\xi}_c}{\partial \boldsymbol{\xi}_n} = {}^c\mathbf{V}_n$ is known as the screw transformation matrix and is given by:

$$ {}^c\mathbf{V}_n = \begin{bmatrix} {}^c\mathbf{R}_n & [{}^c\mathbf{t}_n]_\times \, {}^c\mathbf{R}_n \\ 0 & {}^c\mathbf{R}_n \end{bmatrix} \tag{1.11} $$

It acts as the bridge to transform the end-effector velocity screw $\dot{\boldsymbol{\xi}}_n$ from the end-effector frame to the camera frame [Khalil 02]. The velocity screw (the spatial velocity of the camera) expressed in the camera frame is therefore given by ${}^c\mathbf{V}_n \, \mathbf{J}_\xi(\mathbf{q}) \, \dot{\mathbf{q}}$ and is denoted as $\mathbf{v}_c$.This screw transformation depends on the relative pose between the camera and end-effector as seen from Equation (1.11). Estimating this transformation is the classic hand-eye calibration problem in robotics. There are a plethora of methods[1] available to solve this problem including [Daniilidis 99] and [Horaud 95]. In the eye-in-hand configuration, the camera is rigidly fixed on the end-effector and the relative pose between the two is constant throughout the servo. Hence, it is sufficient to estimate this transformation once. We assume in this work that this matrix has been estimated (a coarse approximation is sufficient) and is readily available.

- In visual servoing, vision feedback is used to specify the task function. $\mathbf{L_s}$ is called the interaction matrix and it links the visual feature variations to the camera motion quantified by the velocity screw $\mathbf{v}_c$. Development or learning of this matrix is the core of any visual servoing application.

We considered so far the classic eye-in-hand visual servoing with the camera mounted on the end-effector of the robot. Next, let us consider the eye-to-hand configuration in which the camera is mounted rigidly in the robot environment. In this case, $\mathbf{J_s}$ in Equation (1.7) can be decomposed as [Chaumette 07]

$$ \mathbf{J_s} = -\mathbf{L_s} \, {}^c\mathbf{V}_n \, \mathbf{J}_\xi(\mathbf{q}) \tag{1.12} $$

In eye-to-hand configuration, the relative pose between the end-effector and the camera is not constant due to the independent motion of the robot. Therefore, ${}^c\mathbf{V}_n$ is not constant and requires an estimation of the pose between the robot end-effector and the camera. Alternatively, Equation (1.12) can be written as

$$ \mathbf{J_s} = -\mathbf{L_s} \, {}^c\mathbf{V}_o \, {}^o\mathbf{J}_\xi(\mathbf{q}) \tag{1.13} $$

where the jacobian ${}^o\mathbf{J}_\xi(\mathbf{q})$ is expressed in the robot base frame. If the camera is motionless in the eye-to-hand case, its pose relative to the robot base frame is constant. Therefore, this pose can be estimated once and even coarsely and used for the visual servoing [Chaumette 07].

From (1.7), we can obtain an expression for the joint velocities.

$$ \dot{\mathbf{q}} = (\mathbf{J_s})^{-1} \, \dot{\mathbf{e}} - \widehat{\frac{\partial \mathbf{s}}{\partial t}} \tag{1.14} $$

[1] A listing of linear and non-linear hand-eye calibration methods is available at [TUM 12]

assuming indeed that the matrix $\mathbf{J_s}$ is invertible. The term $\widehat{\frac{\partial \mathbf{s}}{\partial t}}$ denotes an estimation of the target object motion. As for the errors, a simple option is to specify that the error components in $\mathbf{e}$ decrease exponentially.

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \tag{1.15}$$

Then, we can rewrite the control law in order to take into account the specified error profile as follows:

$$\dot{\mathbf{q}} = -\lambda \left(\mathbf{J_s}\right)^{-1} \mathbf{e} - \widehat{\frac{\partial \mathbf{s}}{\partial t}} \tag{1.16}$$

$\mathbf{J_s}$ is given by (1.12) or (1.13) for eye-to-hand and by (1.10) for eye-in-hand configurations. Irrespective of the provenance of $\mathbf{J_s}$, the above control law minimizes the error in the visual features, provided the stability conditions (to be discussed in Section (1.1.2) are satisfied. Several works in visual servoing use Equation (1.15) but it is not the only choice. For example, [Dahmouche 12] employs a second order decay of the task function.

$$\ddot{\mathbf{e}} + \mathbf{\Lambda}_v \dot{\mathbf{e}} + \mathbf{\Lambda}_p \mathbf{e} = 0 \tag{1.17}$$

where $\mathbf{\Lambda}_v$ and $\mathbf{\Lambda}_p$ are diagonal matrices with appropriate control gains. Few other alternatives in literature for error specification are discussed in [Mansard 07].

The interaction matrix $\mathbf{L_s}$ is built from visual features $\mathbf{s}$. In a typical visual servoing application, visual features are built from image measurements $\mathbf{s}(\mathbf{m}(\mathbf{I}), t)$. The measurements if not accurate make this mapping imprecise and further modelling errors might be introduced. Therefore the interaction matrix is only an approximation of the true one. By convention, an estimated quantity x is marked with $\widehat{x}$. The approximated interaction matrix is denoted $\widehat{\mathbf{L}}_\mathbf{s}$. If we assume that the target is motionless and does not induce any change in values of the features $\mathbf{s}$ and no potential disturbances are acting on the system, the second term in Equation (1.16) can be neglected. Then, this equation gives the classic visual servoing control law $\dot{\mathbf{q}} = -\lambda \widehat{\mathbf{J}}_\mathbf{s}^{-1} \mathbf{e}$. In practice, $\mathbf{J_s} \in \mathbb{R}^{k \times 6}$ is not always invertible[2]. So, the Moore-Penrose pseudoinverse $\widehat{\mathbf{J}}_\mathbf{s}^{+} \in \mathbb{R}^{6 \times k}$ is used in place of the actual inverse.

$$\dot{\mathbf{q}} = -\lambda \widehat{\mathbf{J}}_\mathbf{s}^{+} \mathbf{e} \tag{1.18}$$

The Moore-Penrose pseudoinverse of the matrix $\mathbf{J_s}$ provides the solution with minimal norm which satisfies the equation $\|\dot{\mathbf{e}} - \lambda \mathbf{J_s} \mathbf{J_s}^{+} \mathbf{e}\| = \|\dot{\mathbf{e}} - \mathbf{J_s} \dot{\mathbf{q}}\|$. When $\mathbf{J_s}$ is of full-rank 6, it can be calculated by $\mathbf{J_s}^{+} = \left(\mathbf{J_s}^\top \mathbf{J_s}\right)^{-1} \mathbf{J_s}^\top$. The Singular Value (SVD) technique is normally used to compute the pseudoinverse when $\mathbf{J_s}$ is not of full-rank 6.

In Equation (1.10), we can take $\mathbf{v}_c$ to be the control inputs instead of $\dot{\mathbf{q}}$. Then, Equation (1.9) can be written equivalently as:

$$\dot{\mathbf{s}} = \mathbf{L_s} \mathbf{v}_c + \frac{\partial \mathbf{s}}{\partial t} \tag{1.19}$$

If we assume $\mathbf{s}$ does not vary explicitly with time, that is, $\partial \mathbf{s}/\partial t = 0$ in (1.19), we have the classic relation $\dot{\mathbf{s}} = \mathbf{L_s}\, \mathbf{v}_c$. So, $\dot{\mathbf{s}}$ reflects variations in visual features due to camera motion only. This form is used often in VS literature [Chaumette 06]. It is assumed in

[2] for example, when $k \neq 6$ and when $k = 6$ but $\det(\mathbf{J_s}) = 0$

this case that ${}^{c}\mathrm{V}_{n}$ is known. Then, for an exponential decrease of errors as previously, the control inputs are given by the following control law:

$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{+} \mathbf{e} \tag{1.20}$$

Thus, this control law is based on the Jacobian of the visual feature and is equivalent to the Gauss-Newton minimization (GNM) method, based on the first order Taylor series expansion of $\mathbf{s}(\boldsymbol{\xi})$ [Malis 04]. If the Taylor's expansion is performed upto the second order, Hessian-based control laws can be developed which offered a better estimate of the direction of spatial displacement than Jacobian-based control laws [Lapresté 04].

1.1.1.1 Evaluation modes for the Interaction Matrix

It has to be noted that the value of the interaction matrix $\widehat{\mathbf{L}}_{\mathbf{s}}$ used in the control scheme depends on the visual features $\mathbf{s}$ and depth Z of the scene by virtue of its definition (see Equation (1.50) for instance). It can be calculated in a number of different ways neatly explained in [Chaumette 06].

1. It can be evaluated at each servo iteration using the updated values of the visual features $\mathbf{s}(t)$ and depth estimate $Z(t)$ at that iteration.

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}} \left(\mathbf{s}(t), \hat{Z}(t) \right) \tag{1.21a}$$

   In this case, it has to be ensured that the interaction matrix is of full rank during the system state trajectories. Further, when depth estimates are difficult to obtain or unavailable, an approximation $\hat{Z}(t) = \widehat{Z^*}$ is employed.

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}} \left(\mathbf{s}(t), \widehat{Z^*} \right) \tag{1.21b}$$

   So, there are two variants depending on the choice of Z.

2. Alternatively, it can be evaluated only once using the feature values computed from the image learnt from the desired pose. That is,

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \widehat{Z^*}) \tag{1.22}$$

   When a planar target is used and the camera and target planes are parallel, all the scene points are at the same depth Z^*. In practice, this depth is roughly measured and $\hat{Z}$ is replaced by $\widehat{Z^*}$. Many works in visual servoing adopt this owing to its simplicity of not requiring the interaction matrix to be computed online during the servo. On the other hand, this simplification causes the stability properties to hold only in the neighbourhood of the desired equilibrium state and the resulting trajectories might not be optimal. A control law using this configuration for the interaction matrix is known as the constant JPC (stands for Jacobian Pseudo-inverse Control) control law [Malis 04].

3. Using the mean of the above two matrices known as the PMJ (Pseudoinverse of the mean of the Jacobians) yielded a more satisfactory behaviour [Malis 04]. The VS

tutorial in [Chaumette 06] also showed performace with this choice of the interaction matrix is midway between the performances obtained with the previous two choices.

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \frac{\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \widehat{Z^*}) + \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), \hat{Z}(t))}{2} \tag{1.23}$$

4. In [Marey 08], a general combination of the above two matrices was proposed.

$$\widehat{\mathbf{L}}_{\mathbf{s}} = k\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}) + (1-k)\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}*) \tag{1.24}$$

   The constant $k \in \mathbb{R} \cap]0,1[$ and this enabled handling problems like local minima and singularities.

5. Improved version of the ESM control laws were proposed in [Tahri 08] and later in [Tahri 10a]. The latter discusses an inconsistency in the development of the ESM control laws which causes a performance degradration in specific cases, like large rotational displacements. The improved version presented in [Tahri 10a] is as follows:

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \frac{1}{2}\left(\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), Z(t)) + {}^{c}\mathbf{V}_{c^*}\,\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, Z^*)\right) \tag{1.25}$$

where ${}^{c}\mathbf{V}_{c^*}$ is the screw transformation matrix between the desired and current camera frames.

1.1.2 Stability in Visual Servoing

Stability of a general dynamical system $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ can be analysed using the framework of Lyapunov stability. The system state $\mathbf{x}$ is governed by f which can be linear or non-linear and might not be completely known. The state evolves as a result of the control inputs $\mathbf{u}$, properties of the system itself which determines the shape of f and its interactions with the environment. Lyapunov suggested that $\mathbf{x} = \mathbf{0}$ is an equilibrium point of the system if and only if there exists a function $\mathcal{L}$ such that $\mathcal{L} > 0$ for all $\mathbf{x}$ and $\mathcal{L}(\mathbf{0}) = 0$. Such a function is representative of the energy in a system. This energy function called the Lyapunov function of the system should decrease along the system state trajectories. For a concise treatment of Lyapunov stability theory, we refer the interested reader to [Murray 94]. Let us choose a simple Lyapunov function for the visual servoing system

$$\mathcal{L} = \frac{1}{2}\|\mathbf{e}\|^2 \tag{1.26}$$

Let us note that $\mathcal{L} > 0$ and $\mathcal{L} = 0$ if and only if $\mathbf{e} = \mathbf{0}$. Differentiating $\mathcal{L}$ with respect to time, we have

$$\dot{\mathcal{L}} = \mathbf{e}^\top \dot{\mathbf{e}} \tag{1.27}$$

Using (1.19) in (1.27), we can write

$$\dot{\mathcal{L}} = \mathbf{e}^\top \mathbf{L}_{\mathbf{s}} \mathbf{v}_c \tag{1.28}$$

Substituting (1.18) in (1.28), we obtain

$$\dot{\mathcal{L}} = -\lambda\, \mathbf{e}^\top \mathbf{L}_{\mathbf{s}} \widehat{\mathbf{L}}_{\mathbf{s}}^{+} \mathbf{e} = -\lambda\, \mathbf{e}^\top \mathbf{Q} \mathbf{e} \tag{1.29}$$

where $\mathbf{Q} = (\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+)$. From the basic theorems of Lyapunov stability (see [Murray 94]), if the Lyapunov function $\mathcal{L}(\mathbf{e},t) > 0$ and its derivative $\dot{\mathcal{L}}(\mathbf{e},t) < 0$ in a neighbourhood around the equilibrium point, then the equilibrium point (system desired state) is said to be locally stable in the Lyapunov sense. By definition, (1.26) is positive. Then for the system to be stable, (1.29) should be negative. This is ensured if and only if the matrix $\mathbf{Q}$ is positive definite (PD), denoted $\mathbf{Q} \succ 0$. However, it is not enough to merely state the condition $\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+ \succ 0$ and jump to conclusions about stability, without delving into the details about the number of features and the neighbourhood where the condition holds true. The stability criterion can be expressed by

$$\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+ \succ 0 \quad \text{when } k \leq n \tag{1.30}$$

where n is the number of actuated degrees of freedom. If this condition is satisfied, then as said before, the Lyapunov function derivative would be negative at all system states other than the desired equilibrium where $\mathbf{e} = \mathbf{0}$. In other words, the following condition is respected

$$\dot{\mathcal{L}} < 0 \quad \forall \mathbf{e} \neq \mathbf{0} \tag{1.31}$$

An important point to be noted is that the first matrix, $\mathbf{L_s}$ in condition (1.30) is the true interaction matrix. Strictly speaking, it should be denoted $\bar{\mathbf{L}}(\bar{\mathbf{s}}, \bar{Z})$. Also, a violation in the stability condition can sometimes be assessed even if this true interaction matrix is not available. An obvious example is $\widehat{\mathbf{L}}_\mathbf{s}$ losing rank in which case its determinant is 0 and so the above positivity condition is not true.

If condition (1.31) holds true for configurations throughout the workspace of the robot, then *global asymptotic stability* (GAS) is ensured. If condition (1.31) of the system is valid only in a neighbourhood around the equilibrium state, then this is termed *local asymptotic stability*. The size of this neighbourhood has not been established analytically although from practice it is surprisingly large for visual servoing methods [Chaumette 06]. The convergence domain has been studied empirically around specific desired poses in [Kallem 09], [Dame 11] and more recently in [Teuliere 14]. Let us note that the notion of stability should not to be confused with convergence to an equilibrium state. It is possible that a visual servoing system converges to an equilibrium state without necessarily being stable. This clearly occurs when a local minimum is reached. In this case, we would have $\dot{\mathcal{L}} = 0$ but $\dot{\mathbf{e}} \neq 0$. Furthermore, the stability condition might not necessarily be respected at all instants during the system state trajectory (except when Globally Asymptotically Stable).

- **Case** k=n : If a non-redundant number of visual features is chosen (that is, same as the robot degrees of freedom) and their interaction matrix is of full rank, then by the Rank Nullity Theorem[3], we can infer that condition (1.30) is always ensured. That is, if 6 features are chosen to control of a 6DOF system $k = n = 6$, $\widehat{\mathbf{L}}_\mathbf{s}$ and $\widehat{\mathbf{L}}_\mathbf{s}^+$ are full rank and not too coarsely approximated, $\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+ \succ \mathbf{0}$ in Equation (1.30) is a sufficient condition for stability [Chaumette 06]. In this thesis, one of the goals is to find a set of 6 visual features based on photometric image moments to control all the 6DOF of a robot.

[3]From basic linear algebra, we have the number of columns in a matrix N_c is equal to the sum of the column rank and dimension of the null space $rank(\mathbf{A}) + dim(\mathcal{N}(\mathbf{A})) = N_c$ where $\mathcal{N}(\mathbf{A})$ is the null space of $\mathbf{A}$

- **Case** k>n : When $k > n$, we fall in the redundant case where the product $\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+ \in \mathbb{R}^{k\times k}$ can be utmost[4] of rank n. This means that there might exist feature states $\mathbf{s} \neq \mathbf{s}^* \in \mathbb{R}^k$ which satisfy $\widehat{\mathbf{L}}_\mathbf{s}^+ (\mathbf{s} - \mathbf{s}^*) = \mathbf{0}$. That is, $\mathbf{s} - \mathbf{s}^*$ belongs to the null space of $\widehat{\mathbf{L}}_\mathbf{s}^+$.

$$\mathbf{s} - \mathbf{s}^* \in \mathtt{Ker}(\widehat{\mathbf{L}}_\mathbf{s}^+) \tag{1.32}$$

  Such undesired configurations correspond to local minima. A claim for local asymptotic stability should be evidenced by showing there are no undesired configurations as above in a small neighbourhood around the desired configuration $\boldsymbol{\xi}^*$. Typically in the case of a local minimum, $(\mathbf{s} - \mathbf{s}^*) \neq 0$ but $\mathbf{v}_c = \mathbf{0}$. This would mean that the specified task function error has not yet been driven to $\mathbf{0}$ but control velocities are not generated anymore to regulate them. To recap, for $k > n$, the condition in Equation (1.30) will never be ensured. To find the stability condition for this case, it is necessary to define a new error [Chaumette 06]

$$\mathbf{e}' = \widehat{\mathbf{L}}_\mathbf{s}^+(\mathbf{s} - \mathbf{s}^*) \tag{1.33}$$

  The matrix $\widehat{\mathbf{L}}_\mathbf{s}^+$ is nothing but the matrix $\mathbf{C}$ in Equation (1.5) which handles the regularity condition. We differentiate this equation and then substitute $\dot{\mathbf{e}} = \mathbf{L_s}\mathbf{v}_c$, we have

$$\dot{\mathbf{e}'} = \widehat{\mathbf{L}}_\mathbf{s}^+\dot{\mathbf{e}} + \dot{\widehat{\mathbf{L}}}_\mathbf{s}^+\mathbf{e} = \widehat{\mathbf{L}}_\mathbf{s}^+\mathbf{L_s}\mathbf{v}_c + \dot{\widehat{\mathbf{L}}}_\mathbf{s}^+\mathbf{e} \tag{1.34}$$

  This can be written as

$$\dot{\mathbf{e}'} = (\widehat{\mathbf{L}}_\mathbf{s}^+\mathbf{L_s} + \mathbf{O})\mathbf{v}_c \tag{1.35}$$

  where the matrix $\mathbf{O} \in \mathbb{R}^{n\times n}$ is equal to $\mathbf{0}$ if $\mathbf{e} = 0$, regardless of the choice of $\widehat{\mathbf{L}}_\mathbf{s}^+$. Further, if we adopt the control law $\mathbf{v}_c = -\lambda\mathbf{e}'$ and follow same steps as from Equation (1.29) with the Lyapunov function $\mathcal{L}' = \frac{1}{2}\|\mathbf{e}'\|^2$, we arrive at the following condition

$$\widehat{\mathbf{L}}_\mathbf{s}^+\mathbf{L_s} + \mathbf{O} \succeq 0 \tag{1.36}$$

  If only a small neighbourhood of the equilibrium state $\mathbf{e} = \mathbf{e}^* = \mathbf{0}$ is considered, then the condition reduces to

$$\widehat{\mathbf{L}}_\mathbf{s}^+\mathbf{L_s} \succeq 0 \quad k > n \tag{1.37}$$

  Thus, only local asymptotic stability can be obtained for VS when $k > n$. IBVS methods which use coordinates of several image points (more than the degrees of freedom) typically fall under the $k > n$ case. (We will discuss how image points can be used as visual features in Section 1.2.1)

- **Case** k<n : In this case, the number of features is less than the degrees of freedom of the system. This is not a very common scenario but in principle, can be used for visual servoing when the task to be realized does not constrain all the degrees of freedom of the system. In that case, the condition in Equation(1.30) is sufficient to ensure stability.

[4]From matrix theory, we know that the rank of a product of two matrices $\mathbf{A}$ and $\mathbf{B}$ cannot exceed the rank of its individual matrices, $rank(\mathbf{AB}) \leq \min(rank(\mathbf{A}), rank(\mathbf{B}))$

Robustness Analysis In (1.5), if there exists a robot configuration $\mathbf{q}^*$ such that the task function $\mathbf{e} = 0$ when $\mathbf{s}(\mathbf{q}) = \mathbf{s}^*$, we can perform a Taylor's expansion of this task function around the desired equilibrium $\mathbf{s}^*$.

$$\mathbf{e}(\mathbf{s}) = \mathbf{e}(\mathbf{s}*) + \left(\frac{\partial \mathbf{e}}{\partial \mathbf{s}}\bigg|_{\mathbf{s}=\mathbf{s}*}\right)(\mathbf{s}-\mathbf{s}^*) + \frac{1}{2}(\mathbf{s}-\mathbf{s}^*)^\top \left(\frac{\partial^2 \mathbf{e}}{\partial \mathbf{s}^2}\bigg|_{\mathbf{s}=\mathbf{s}*}\right)(\mathbf{s}-\mathbf{s}^*) + \mathrm{H}(\mathbf{e})(\mathbf{s}-\mathbf{s}^*) \quad (1.38)$$

where H is a grouping of all the higher order terms in the Taylor's expansion. This can be re-written as a matrix transformation

$$\mathbf{e} = \mathbf{U}(\mathbf{s} - \mathbf{s}^*) \quad (1.39)$$

where $\mathbf{U} = \mathbf{U}(\mathbf{s}, \mathbf{s}^*, \mathbf{A}, \natural)$. The matrix $\mathbf{U}$ is a function of the current and desired features, the camera calibration matrix $\mathbf{A}$ and $\natural$, a vector of parameters of all geometrical transformations between the image and scene. For example, this could be the parameters that describe a homography matrix or a homogeneous transformation matrix that describes the relative pose of the camera with respect to the scene. In general however, $\mathbf{U}$ is not known analytically and the finding of which is non-trivial. This in fact poses a challenge in the stability analysis of visual servoing systems. For simplicity, we can assume that the signal measurement errors are corrupted by additive noise $\hat{\mathbf{s}} = \mathbf{s} + \boldsymbol{\epsilon}$. Using this noise model and Equation (1.39) and after some developments, we obtain [Malis 98].

$$\hat{\mathbf{e}} = \mathbf{E}\mathbf{e} + \mathbf{n} \quad (1.40)$$

where $\mathbf{E} = \mathbf{U}(\mathbf{s}, \mathbf{s}^*, \hat{\mathbf{A}}, \hat{\natural})\, \mathbf{U}^+(\mathbf{s}, \mathbf{s}^*, \mathbf{A}, \natural)$ and $\mathbf{n}$ is the noise introduced in the task function. It has to be noted that $\mathbf{n}$ results from estimation errors in $\mathbf{s}$ whereas the effect of incorrect camera calibration and geometrical parameters are encapsulated in $\mathbf{E}$. Besides modelling errors, the image formation itself and subsequent measurements from it, $\mathbf{m}(\mathbf{I})$ might induce noise in the system. In general, for robustness, visual servoing methods should be immune to signal measurement errors and inaccurate camera parameters $\hat{\mathbf{A}}$. A probabilistic approach to study the effect of measurement errors on the visual servoing is discussed in [Kyrki 06]. Recently, [Assa 13] considered VS as a discrete-time system and modelled the uncertainty of IBVS and PBVS methods.

1.1.3 A primer on Visual Feature Selection

First, we will revisit the image formation basics and introduce a few basic notions before stepping into visual feature selection. This foundation is necessary before the problems can be explicitly specified.

Image formation In general, an image $\boldsymbol{\mathcal{I}}$ is a set $\{I_i\}$, $(i = 1 \ldots N)$ of N elements, each of which can be described by the following function $I : D \subset \mathbb{R}^2 \rightarrow \mathbb{R}^k_+$ where k indicates the number of channels in the image. Inspired by [Soatto 11], let us consider a generative image model for image formation, but with some specific differences which will be highlighted in due course.

$$\begin{cases} I(\mathbf{x}) = h\left(\Lambda, \boldsymbol{\nu}\right) \\ \mathbf{x} = \pi(\boldsymbol{\xi}(\mathbf{X})),\ \mathbf{X} \in S \end{cases} \quad (1.41)$$

where h is the contrast functional. The pose $\boldsymbol{\xi}$ will induce effects observed in the image due to changes in viewpoint. Because the pose $\boldsymbol{\xi}$ has a group structure, its action is invertible. $\boldsymbol{\xi}$ is considered as a nuisance for recognition tasks in [Soatto 11] whereas in visual servoing, this viewpoint transformation is indeed the result of a control action $\mathbf{u}(t)$ decided based on the image. We denote the set of all points (3D) in the imaged scene as $\mathbf{X} \in \mathbb{R}^3$ and their corresponding projections in the image plane as $\mathbf{x} \in \mathbb{R}^2$. π is the perspective projection of a spatial point to the image plane. Λ refers to the photometric and geometric properties of objects in the imaged scene S. The variable $\boldsymbol{\nu}$ refers to all nuisances like illumination variations, occlusions, quantization effects and shadows that cannot be explicitly modelled but still affect the acquired image.

Visual Feature A visual feature is any deterministic function of the image that produces a real value, that is $s(\boldsymbol{\mathcal{I}}) \rightarrow \mathbb{R}$.

Motion Perceptibility The motion perceptibility of a visual feature s with respect to a specific camera motion is defined as the ratio of the change in the visual feature to change in the variable parametrizing that motion. The motion can be either translation or rotation along or around any axis of an appropriate Cartesian reference frame. A simple example can be helpful in understanding motion perceptibility. With no loss of generality, let us assume that rotations are parametrized by the axis-angle representation and the camera is oriented at an angle of θ_z around the optic axis. The visual feature measured from the image reads s_z. Then, let us apply a pure rotational motion around the camera optic axis specified by rotation matrix $\mathbf{R}_z(\Delta\theta_z)$ resulting in new camera orientation $\theta_z^{'}$, whose effect in the image is measured as $s_z^{'}$. Then, the motion perceptibility of the visual feature s with respect to optic axis rotation can be computed using

$$\varrho_{s_z} = \frac{s_z^{'} - s_z}{\Delta\theta_z} \tag{1.42}$$

where $\Delta\theta_z = \theta_z^{'} - \theta_z$. In any case, visual features should have high motion perceptibility [Sharma 97] [Corke 11].

Ideal Visual Feature Set A set of visual features $\mathbf{s} = \{s_i(\boldsymbol{\mathcal{I}})\}_{i=1\ldots n}$ is considered to be ideal for visually servoing a n-DOF actuated system if its interaction matrix is the identity matrix

$$\mathbf{L_s} = \mathbb{I}_{n\times n} \tag{1.43}$$

This would mean that each visual feature corresponds to one and only one specific camera motion and also varies linearly with respect to it. Achieving this is considered as the *holy grail* of visual servoing. To obtain this in practice is certainly extremely difficult if not impossible.

Invariance It is difficult to obtain an ideal visual feature set $\mathbf{s}$ such that Equation (1.43) holds. But methods like 2.5D visual servoing [Malis 00], to be discussed in Section 1.2.3, succeeded in finding a set of features for which the rotational motions are decoupled from the translations. In fact, it is the invariance property of a visual feature with respect to

some dof which is the most important enabler towards achieving this decoupling. We will see next that features based on image moments results in a high degree of decoupling.

In order for visual features to be useful in real-world visual servoing, they must be invariant not only to geometrical transformations but also to nuisances like image noise and lighting variations gathered collectively as $\boldsymbol{\nu}$ in Equation(1.41). A visual feature which is variant w.r.t to 1 specific dof, invariant to others and to image nuisances is considered to be an attractive visual feature (to control that specific dof).

Visual servoing methods can be broadly classified into feature-based and correspondance-less visual servoing methods based on the nature of the visual features developed from image feedback. We will describe these methods next, along with their respective advantages and shortcomings.

1.2 Geometric Feature-based methods

In feature-based methods, a set of image-based primitives $\mathbf{m}_i \in \mathcal{M}_{2D}$ are extracted from the image, where $\mathcal{M}_{2D}$ is the set of all possible measurements in the image. This extraction employs image processing techniques like visual detection, matching, visual tracking and/or segmentation. These primitives are normally projections in the image of the corresponding 3D primitives (points, straight lines) $\mathbf{P}_i$ available in the imaged scene $\mathbf{P}_i \in \mathcal{S}_{3D}$. Based on how the extracted image measurements are used to derive visual features, feature-based methods can be classified into image-based visual servoing (IBVS), pose-based visual servoing (PBVS) and also hybrid methods combining the above known as 2.5D visual servoing.

1.2.1 Image-Based Visual Servoing (IBVS)

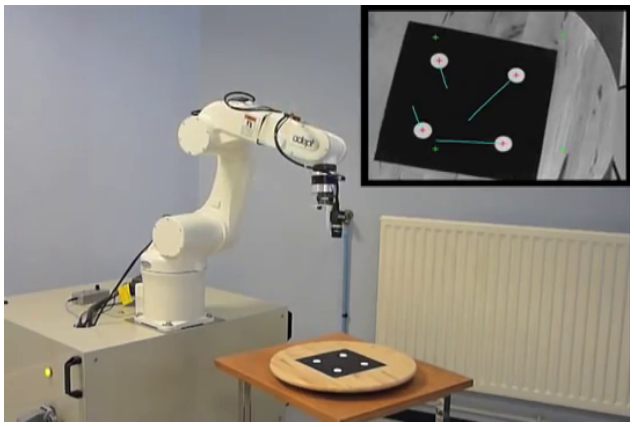
In IBVS, the task function is defined directly from measurements of 2D primitives observed in the image. Hence, IBVS is also called 2D visual servoing.

$$\mathbf{e} = \mathbf{s}(\mathbf{m}(\mathbf{I}(t))) - \mathbf{s}^* \tag{1.44}$$

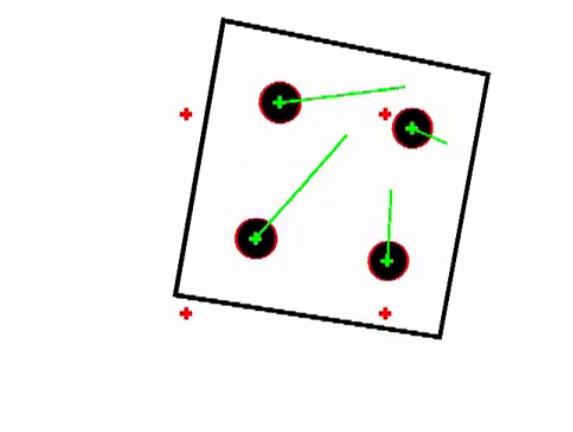
The most commonly employed visual features $\mathbf{s} \in \mathbb{R}^{2n}$ are the coordinates of n points that are observed in the image plane [Weiss 87] [Feddema 89]. Let us consider as visual features coordinates of a set of points observed by the camera (see Figs.1.3(a) and 1.3(b)): $\mathbf{s} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \ \ldots, \mathbf{x}_n)$. Let $\mathbf{s}^*$ be the feature set learnt from the desired pose. The visual servoing control law

$$\mathbf{v}_c = -\lambda \hat{\mathbf{L}}_{\mathbf{s}}^{+} \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}_1^* \\ \mathbf{x}_2 - \mathbf{x}_2^* \\ \vdots \\ \mathbf{x}_n - \mathbf{x}_n^* \end{bmatrix} \tag{1.45}$$

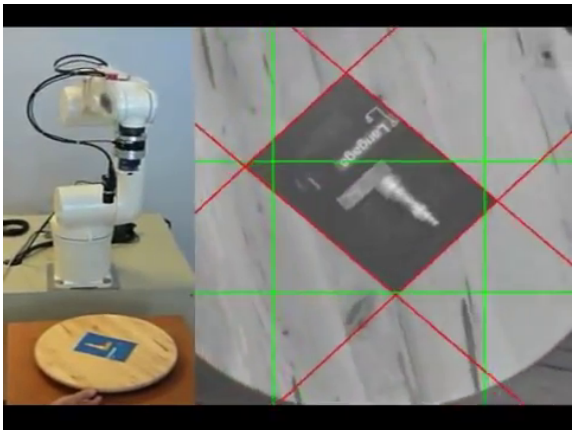
should drive the error $\mathbf{e}$ in (1.44) to 0 in an exponential manner. The interaction matrix for an image point is developed next since it is the basic foundation for all visual features that appear later.



(a) Visual Servoing with a set of 4 dots in the image



(b) Camera view with feature trajectories



(c) Visual servoing with lines

Figure 1.3 – Image-Based Visual Servoing with image points and straight lines as visual features

1.2.1.1 Interaction Matrix for an image point

Let $\mathbf{X} = (X, Y, Z)$ be a 3D point in the scene whose coordinates are expressed in the camera frame. Let $\mathbf{A} = (u_0, v_0, f, \alpha)$ be the parameters of a pin-hole camera imaging the scene, where (u_0, v_0) is the principal point in the image, f is the camera focal length and α is the ratio of pixel size. Then, under perspective projection, the 3D scene point $\mathbf{X}$ projects in the image as a 2D point $\mathbf{x}$, given by

$$\begin{cases} x = \dfrac{X}{Z} = \dfrac{u - u_0}{f\alpha} \\ y = \dfrac{Y}{Z} = \dfrac{v - v_0}{f} \end{cases} \tag{1.46}$$

The velocity of the camera is linked to the velocity of the 3D point by $\dot{\mathbf{X}} = -\mathbf{v}_c - \omega_c \times \mathbf{X}$. This fundamental kinematics relation connects the variations of a point in the physical world with the motions of the camera and forms the basis of the development of the

interaction matrix [Espiau 92] [Chaumette 06]. Expanding this relation, we can write

$$\begin{cases} \dot{X} = -v_x - \omega_y Z + \omega_z Y \\ \dot{Y} = -v_y - \omega_z Z + \omega_x Z \\ \dot{Z} = -v_z - \omega_x Z + \omega_y X \end{cases} \tag{1.47}$$

Further, differentiating Equation (1.46), we have

$$\begin{cases} \dot{x} = \dfrac{\dot{X} - x\dot{Z}}{Z} \\ \dot{y} = \dfrac{\dot{Y} - y\dot{Z}}{Z} \end{cases} \tag{1.48}$$

Applying Equation (1.47) in Equation (1.48), we have

$$\begin{cases} \dot{x} = -\dfrac{v_x}{Z} + \dfrac{xv_z}{Z} + xy\omega_x - (1+x^2)\omega_y + y\omega_z \\ \dot{x} = -\dfrac{v_y}{Z} + \dfrac{yv_z}{Z} + (1+y^2)\omega_x - xy\omega_y - x\omega_z \end{cases} \tag{1.49}$$

From this, the interaction matrix of an image point (that satisfies $\dot{\mathbf{x}} = \mathbf{L_x}\mathbf{v}_c$) can be written directly:

$$\mathbf{L_x} = \begin{bmatrix} \mathbf{L}_x \\ \mathbf{L}_y \end{bmatrix} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{bmatrix} \tag{1.50}$$

Hence $\mathbf{L_x}$ is a function of the visual features $\mathbf{s}$, depth Z and the camera intrinsic parameters $\hat{\mathbf{A}}$. The Equation (1.50) links the motion of the image points to the motion of the camera and this result is made use of in the development of the interaction matrices of intensity-based features. The interaction for a set of n points is obtained by stacking the interaction matrices of each of the individual points.

$$\mathbf{L_s} = \begin{bmatrix} \mathbf{L}_{\mathbf{x}_1}(\mathbf{x}_1, Z_1, \hat{\mathbf{A}}) \\ \mathbf{L}_{\mathbf{x}_2}(\mathbf{x}_2, Z_2, \hat{\mathbf{A}}) \\ \vdots \\ \mathbf{L}_{\mathbf{x}_n}(\mathbf{x}_n, Z_n, \hat{\mathbf{A}}) \end{bmatrix} \tag{1.51}$$

On the same lines, [Iwatsuki 05] transformed the tracked points to the cylindrical coordinate system $\mathbf{s} = (\rho, \theta)$ with $\rho = \sqrt{x^2 + y^2}$ and $\theta = \arctan(\frac{y}{x})$ using them as visual features. This work solved the camera retreat problem where a large error in rotational pose around the optical axis caused a retreating motion of the camera along this axis [Chaumette 98]. Other geometrical primitives like straight lines (see Fig. 1.3(c)), ellipses and contours in the image of tridimensional primitives like spheres and cylinders can also be equally used [Espiau 92][Chaumette 90]. This involves detection and tracking of these primitives throughout the servoing process.

Since the error measure $\mathbf{e}$ to be regulated to 0 is defined directly in the image space, satisfactory feature trajectories are obtained at the image level. But the pose or end-effector motions are not directly controlled. Hence, the obtained Cartesian trajectories

exhibit a detour from an ideal geodesic. One drawback of IBVS is the unwanted possibility of convergence to local minima in the case of redundant features $k > 6$. Therefore, the visual servoing may fail when large displacements are involved. Also only a guarantee of local asymptotic stability exists for IBVS. Stability and convergence problems that might appear in IBVS are discussed in [Chaumette 98]. IBVS is shown to be robust to modelling errors in [Espiau 94].

1.2.1.2 Image Moments-Based Visual Servoing

Image moments are a simplified representation of the image signal [Mukundan 98] and were introduced in visual servoing in [Chaumette 04]. In [Chaumette 04], the analytical form for the interaction matrix was determined and it was shown that the numerical stability of the system was greatly improved and an excellent decoupling was obtained by using visual features derived from image moments. Later, a new set of visual features based on image moment invariants were proposed in [Tahri 05a]. Image moments can be computed from a set of points or from a well-segmented region in the image. In the latter case, no feature extraction or matching steps are performed. The contributions of this thesis fall under this area and so image moments-based visual servoing is discussed in complete details in the subsequent chapter.

1.2.1.3 Visual Servoing based on Gaussian Mixture Models

A gaussian mixture model(GMM) is a convex combination of a certain number (lets say N) of Gaussian probability distribution functions (pdfs). In [AbdulHafez 08], a set of feature points $\mathbf{X}$ was extracted from the image and then modelled as a GMM. A GMM is parametrized by the mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$ of each of those N pdfs. It can be estimated by EM(Expectation Maximization) algorithm given $\mathbf{X}$. The GMM from feature points at the current and desired poses are given by:

$$s = f(\mathbf{X}, \boldsymbol{\xi}) = \sum_{i=1}^{N} \alpha_i \mathcal{N}(\mathbf{X}; \mathbf{R}\boldsymbol{\mu}_i + \mathbf{t}, \mathbf{R}\boldsymbol{\Sigma}_i\mathbf{R}^\top) \tag{1.52a}$$

$$s^* = g(\mathbf{X}) = \sum_{i=j}^{N} \beta_i \mathcal{N}(\mathbf{X}; \boldsymbol{\mu}_i^*, \boldsymbol{\Sigma}_i^*) \tag{1.52b}$$

Visual servoing is then equivalent to minimizing the cost function namely the L_2 norm between the GMMs estimated at the current and desired poses.

$$\mathcal{C} = \|s - s^*\| = \int [f(\mathbf{X}, \boldsymbol{\xi}) - g(\mathbf{X})]^2 d\boldsymbol{\xi} \tag{1.53}$$

The Lyapunov function $\mathcal{L} = \frac{1}{2}(\mathcal{C})^2$ can be minimized by choosing the following control input

$$\mathbf{u} = -(\mathcal{C})\frac{\partial \mathcal{C}}{\partial \boldsymbol{\xi}} \tag{1.54}$$

It is useful that the L_2 norm of the error and the gradient $\dfrac{\partial \mathcal{C}}{\partial \boldsymbol{\xi}}$ are available in closed form. The depth distribution of the feature points is necessary to control all the 6DOF of the

robot. The results presented in [AbdulHafez 08] show that the control signals generated have non-smooth profiles and the camera trajectory is not close to a geodesic between the current and desired poses. The convergence properties of the algorithm with respect to errors in the estimated GMM parameters α_i, β_j are not known. In addition, whether the case $\alpha_i \neq \beta_j$ will affect the algorithm convergence also remains unknown. This method does not need any visual tracking steps but the extraction of image points $\mathbf{X}$ is unavoidable.

1.2.2 Pose-Based Visual Servoing (PBVS)

In Pose-Based Visual Servoing (PBVS) [Wilson 96], [Martinet 99] the error to be regulated to 0 is directly linked to the robot pose. The error in PBVS is defined in pose space and not image space. Hence it known as 3D visual servoing.

$$\mathbf{s} = \mathbf{s}(m(\mathbf{I}(t), \mathrm{A}, \mathbf{X}) \tag{1.55}$$

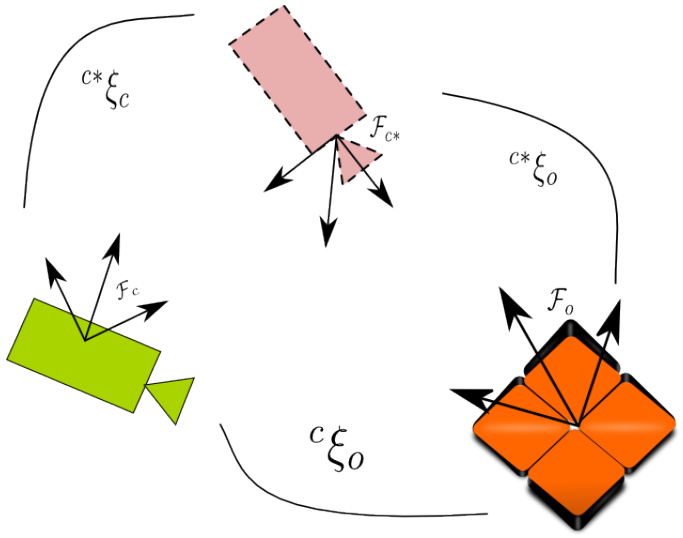


Figure 1.4 – Pose-based visual servoing scheme

The key aspect of PBVS is the estimation of the camera pose ${}^c\boldsymbol{\xi}_o(t)$ from the acquired images. As explained previously, elements of $\mathbb{SO}(3)$ can be expressed in various parametrizations. If we choose the minimal axis-angle parametrization, we have $\mathbf{s} = (\mathbf{t}, \boldsymbol{\theta}\mathbf{u})$. Essentially, a pose estimation algorithm is mandatory in the control loop to measure the current location of the visual sensor with respect to some observable target. For instance, a pose estimation method based on the Extended Kalman Filter is used in [Wilson 96]. Several methods including virtual visual servoing (VVS) [Marchand 02] and model-based methods [Comport 04] [Petit 13] exist for the pose estimation problem. [Doignon 07] is a nice reference that discusses the pose estimation problem in the context of robot vision.

There are two variants of PBVS based on whether the control is formulated in either the current or desired camera frames [Chaumette 07]. If we choose to perform the control

using object pose relative to the current sensor frame, that is $\mathbf{s} = ({}^{c}\mathbf{t}_o, \boldsymbol{\theta}\mathbf{u})$, $\mathbf{s}^* = ({}^{c*}\mathbf{t}_o, \mathbf{0})$, then $\mathbf{e} = ({}^{c}\mathbf{t}_o - {}^{c*}\mathbf{t}_o, \boldsymbol{\theta}\mathbf{u})$. The interaction matrix can then be obtained as

$$\mathbf{L}_{\mathbf{s}} = \begin{bmatrix} -\mathbf{I}_3 & [{}^{c}\mathbf{t}_o]_\times \\ \mathbf{0} & \mathbf{L}_{\theta\mathbf{u}} \end{bmatrix} \tag{1.56}$$

where

$$\mathbf{L}_{\theta\mathbf{u}} = \mathbf{I}_3 + \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\operatorname{sinc}\theta}{\operatorname{sinc}^2\frac{\theta}{2}}\right)[\mathbf{u}]_\times^2 \tag{1.57}$$

where $[\mathbf{v}]_\times$ denotes the skew-symmetric matrix of $\mathbf{v}$. $\operatorname{sinc}\theta = \sin\theta/\theta$ is the sinus cardinal. The matrix $\mathbf{L}_{\theta\mathbf{u}}$ has the property ${\mathbf{L}_{\theta\mathbf{u}}}^{-1}\theta\mathbf{u} = \theta\mathbf{u}$. The $\mathbf{L}_{\theta\mathbf{u}}$ matrix is non-singular when $\theta \neq 2k\pi$, k being a non-zero integer. Then, the translational velocity screw can be calculated as

$$\boldsymbol{\upsilon}_c = -\lambda\Big(({}^{c*}\mathbf{t}_o - {}^{c}\mathbf{t}_o) + [{}^{c}\mathbf{t}_o]_\times\theta\mathbf{u}\Big) \tag{1.58}$$

Alternatively, the control can be defined by the pose of the current sensor frame expressed in the desired sensor frame. In this case, we have $\mathbf{s} = \begin{bmatrix} {}^{c*}\mathbf{t}_c & \boldsymbol{\theta}\mathbf{u} \end{bmatrix}$ and $\mathbf{e} = \mathbf{s}$ since $\mathbf{s}^* = 0$. Typically, the output from pose estimation algorithms is the relative pose of the target expressed in the sensor frame ${}^{c}\mathbf{X}_o$. Then, we use the following relation to get the relative pose of the sensor expressed in the desired sensor frame.

$${}^{c*}\mathbf{X}_c = {}^{c*}\mathbf{X}_o\,{}^{c}\mathbf{X}_o{}^{-1} \tag{1.59}$$

The interaction matrix for this case exhibits the following structure

$$\mathbf{L}_{\mathbf{s}} = \begin{bmatrix} {}^{c*}\mathbf{R}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{\theta\mathbf{u}} \end{bmatrix} \tag{1.60}$$

From Eq(1.60), we see that there is a decoupling of the translational and rotational motions. The translational velocity screw in this case is obtained as

$$\boldsymbol{\upsilon}_c = -\lambda\,{}^{c*}\mathbf{R}_c\,{}^{c*}\mathbf{t}_c \tag{1.61}$$

In both equations (1.56) and (1.60), the interaction matrix with respect to rotational motions is the same [Malis 00] [Chaumette 06] and the control law for these dof is given by

$$\boldsymbol{\omega}_c = -\lambda\theta\mathbf{u} \tag{1.62}$$

In PBVS, the control laws result in an exponential decrease of errors in the rotational pose in fact following a geodesic. The camera on the other hand follows a straight line trajectory with control law (1.60) but not with (1.56). It is to be noted that the depth Z is involved directly in the pose features in PBVS whereas in IBVS, only the interaction matrix depended on Z while the features themselves were free of it. Many works in visual servoing mention that PBVS has no control of trajectories in the feature space and visual features might leave the field of view. But [JanabiSharifi 11] in fact showed that there is an indirect control of image trajectories (except in the case where strong assumptions are made to simplify the interaction matrix).

Stability in PBVS For PBVS, the stability condition in Equation (1.30) is easily ensured since $\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^+ = \mathbf{I}_6$, under the strong assumption that the pose is estimated perfectly [Chaumette 06]. This is true in Equations (1.56) and (1.60) when $\mathbf{L}_{\theta\mathbf{u}}$ is non-singular (for all $\theta \neq 2k\pi$ as mentioned above). The pose features used in the control law are estimated from visual feature measurements in the image. When these measurements are erroneous, the pose estimation is not accurate breaking the strong assumption made above. The PBVS method is therefore adversely affected by errors in the pose estimation.

For a concise explanation of IBVS and PBVS including their relative advantages and stability aspects, we refer the reader to [Chaumette 06]. A formal well-rounded treatment can be found in [JanabiSharifi 11] : a common comparison framework to study IBVS and PBVS is presented from the point of view of stability, robustness, sensitivity to modelling errors and performance in image and Cartesian spaces.

1.2.3 Hybrid Visual Servoing

Hybrid methods were developed to combine the advantages in PBVS with those in IBVS and avoid their respective shortcomings. We can rewrite the relation $\dot{\mathbf{s}} = \mathbf{L}_s\mathbf{v}_c$ as

$$\dot{\mathbf{s}} = \begin{bmatrix}\mathbf{L}_{\boldsymbol{\upsilon}} & \mathbf{L}_{\boldsymbol{\omega}}\end{bmatrix}\begin{bmatrix}\boldsymbol{\upsilon}_c \\ \boldsymbol{\omega}_c\end{bmatrix} = \mathbf{L}_{\boldsymbol{\upsilon}}\boldsymbol{\upsilon}_c + \mathbf{L}_{\boldsymbol{\omega}}\boldsymbol{\omega}_c \tag{1.63}$$

Then

$$\boldsymbol{\upsilon}_c = \mathbf{L}_{\boldsymbol{\upsilon}}^+(-\mathbf{L}_{\boldsymbol{\omega}}\boldsymbol{\omega}_c + \dot{\mathbf{s}}) \tag{1.64}$$

On substitution of $\dot{\mathbf{e}} = \dot{\mathbf{s}} = -\lambda\,\mathbf{e}(t)$ in (1.64), we get

$$\boldsymbol{\upsilon}_c = -\mathbf{L}_{\boldsymbol{\upsilon}}^+(\mathbf{L}_{\boldsymbol{\omega}}\boldsymbol{\omega}_c + \lambda\mathbf{e}(t)) \tag{1.65}$$

Thus, we see that the error to the translational control input is modified to take into account errors induced due to rotation motions.

2.5D Visual Servoing

As explained in Section1.2.2, we saw that PBVS control law resulted in the errors in rotation following a geodesic. The 2.5D method [Malis 00] proposed the following as visual features :

$$\mathbf{s} = \begin{bmatrix}\mathbf{s}_t \\ \theta\mathbf{u}\end{bmatrix} = \begin{bmatrix}\mathbf{x} \\ \log Z \\ \theta\mathbf{u}\end{bmatrix} \tag{1.66}$$

where $\mathbf{s}_t$ is the feature set responsible for controlling the translational motions. It contains $\mathbf{x} \in \mathbb{R}^2$ which is an image point that is used in IBVS (as in Section 1.2.1) and $Z \in \mathbb{R}$ which is the depth of that image point. $\theta\mathbf{u} \in \mathbb{R}^3$ is the rotational pose used in PBVS. The interaction matrix of the feature $\log Z$ is given by

$$\mathbf{L}_{\log Z} = \frac{1}{Z}\begin{bmatrix}0 & 0 & -1 & -y & x & 0\end{bmatrix} \tag{1.67}$$

Combining (1.67) with (1.50), we can write down the interaction matrix of the translational control input as:

$$\mathbf{L}_{\mathbf{s}_t} = \left[\frac{1}{Z} \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} xy & -(1+x^2) & y \\ 1+y^2 & -xy & -x \\ -y & x & 0 \end{bmatrix} \right] \tag{1.68}$$

The left 3×3 matrix exhibits a block triangular structure. The diagonal values suggest a strong correlation between the features chosen and the corresponding translational degree of freedom. Further, we know from Equation (1.56) that the rotations are decoupled from the translations. Assuming that the pose estimation algorithm is accurate and in the absence of coarse modelling errors, the 2.5D VS is globally asymptotically stable [Malis 02].

In [Cervera 03], 2-D homogeneous coordinates of the image points (by fusing depth information) are used as visual features. This method was shown to be free of local minima. Several hybrid schemes can be designed by combining 2D and 3D features. Since PBVS in general is believed not to exercise control over image space trajectories, [Chaumette 00] proposed to use the centre of gravity in the visual feature set. That is, $\mathbf{x}_g$ can be included as a feature in Equation (1.67). Then, both the camera trajectory and the trajectory of $\mathbf{x}_g$ in the image followed a straight line. 2.5D visual servoing was applied to the case of omnidirectional vision sensors in [Abdelkader 05].

In [Gans 11], an adaptive VS controller was developed to stabilize the errors in the image and pose spaces simultaneously. An image space task function with image points $\mathbf{e}_i$ (like in Equation(1.45)) and pose space task function $\mathbf{e}_\xi$ like in PBVS were defined. The depth is adaptively estimated and incorporated in the interaction matrix for image points. It is further injected into the estimated homography (Methods based on Homography estimation are discussed in 1.2.4) to recover the translational pose between the desired and current camera frames. Without proof, the asymptotic convergence of this method is believed to lie at the intersection of the convergence of IBVS and PBVS methods $\mathcal{D}* = \mathcal{D}_i \cap \mathcal{D}_\xi$.

Thus, the above methods tried to partition the feature space and connect the features to individual degrees of freedom.

The hybrid method [Nematollahi 12] formulates an objective function incorporating both the IBVS and PBVS errors as follows.

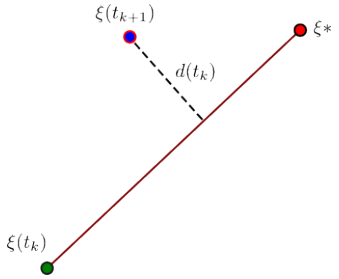


Figure 1.5 – Distance between the next pose and the geodesic between the current and desired poses

$$\min \lambda d_c(t_k) + d_s(t_k) \tag{1.69}$$

where λ is a positive constant, $d_c = \frac{\|\mathbf{e}_c(t_{k+1}) \times \mathbf{e}_c(t_k)\|}{\|\mathbf{e}_c(t_k)\|}$ is the distance between the next camera pose and the line connecting the current pose and desired pose (see Figure 1.5). In the same way, the distance between the visual features at the next instant and the line connecting the current and desired feature values is defined as $d_s = \frac{\|\mathbf{e}_s(t_{k+1}) \times \mathbf{e}_s(t_k)\|}{\|\mathbf{e}_s(t_k)\|}$. The visual features in this method are the coordinates of points tracked in the image. In fact, the average length of the image feature trajectories and the camera trajectories are minimized. All the necessary behaviour like the monotonous decrease in errors are introduced as constraints to this second order conic optimization (SOCO) problem. So, the drawback of the method is its computational expensive cost.

Switching Methods

In hybrid methods based on control law switching, the system does not partition the feature space. Instead, a switch between two disparate control laws is performed based on specific criteria. In [Chesi 04], if the visual features are not present in the field of view, the control switches from PBVS controller to another which performs a receding motion along the optic axis. This motion facilitates to bring back the features into the camera's field of view. Here, a visibility criterion decides the switching instants. Another switching approach can be found in [Gans 07] which used both IBVS and PBVS control laws. The quadratic Lyapunov function for PBVS is given by:

$$\mathcal{L}_{\mathcal{P}} = \frac{1}{2}\mathbf{e}_p^{\top}\mathbf{H}\mathbf{e}_p \quad \text{with} \quad \mathbf{H} = \begin{bmatrix} \eta\mathbf{I} & \mathbf{0} \\ \mathbf{0} & (1-\eta)\mathbf{I} \end{bmatrix} \text{ and } \eta \in (0,1) \tag{1.70}$$

where $\mathbf{e}_p$ is the error in pose as in PBVS. For the IBVS controller, the Lyapunov function is given by:

$$\mathcal{L}_{\mathcal{I}} = \frac{1}{2}\mathbf{e}_i^{\top}\mathbf{e}_i \tag{1.71}$$

where $\mathbf{e}_i$ are errors defined in image space. The visual servoing system then performs a state-based switching as follows:

- When operating in IBVS mode, if $\mathcal{L}_{\mathcal{P}} \geq (1/2)\gamma_p^2$, the control switches to PBVS mode.
- Similarly, when operating in PBVS mode, if $\mathcal{L}_{\mathcal{I}} \geq (1/2)\gamma_i^2$, the control switches to IBVS mode.

In general, it is difficult to prove the stability of switched systems [Liberzon 99]. Proof for local stability for both state-based and arbitrary switching can be found in [Gans 07].

1.2.4 Visual servoing based on Projective Geometric Measures

Projective geometry is a generalization of Euclidean geometry that relates geometric entities (like points/lines) in projective space [5]. Computer Vision applied concepts from projective geometry to establish correspondences between multiple views of scenes imaged

[5] If V is a vector space with $\mathrm{V} \in \mathbb{R}^{n+1}$, then the projective space $\mathrm{P(V)}$ of dimension n is the set of all single dimensional vector subspaces of V

by vision sensors [Hartley 04], [Faugeras 01]. Visual servoing is basically the synthesis of control laws which is based on the views of a scene captured by a camera. Several methods naturally proposed features based on measures adopted from projective geometry.

Homography based Visual Servoing

Homography is an invertible relationship that links the projective points between two images induced by a plane in 3D space. Let us consider a 3D point observed by a camera at two different poses. Let the coordinate frames for the camera at these poses be denoted as $\mathcal{F}^*$ and $\mathcal{F}$ respectively. The projections of the 3D point $\mathbf{P}^* = [X^*, Y^*, Z^*]$ (expressed in the frame $\mathcal{F}^*$) into the image planes of these two cameras are $\mathbf{p}^*$ and $\mathbf{p}$. Let these projections be expressed by $\mathbf{p}^* = \mathbf{H}_1\mathbf{P}^*$ and $\mathbf{p} = \mathbf{H}_2\mathbf{P}^*$. If $\mathbf{P}^*$ belongs to a plane, then the two projected image points are related by a homography $\mathbf{H}$ induced by the plane containing the 3D point [Hartley 04].

$$\mathbf{p}^* \simeq \mathbf{H}_1\mathbf{H}_2^{-1}\mathbf{p} \simeq \mathbf{H}\mathbf{p} \tag{1.72}$$

The $\mathbf{H}$ which relates the two projective points upto a scale is known as the plane induced homography. If the relative pose between $\mathcal{F}^*$ and $\mathcal{F}$ is given by $\mathbf{R} \in \mathbb{SO}(3)$ and $\mathbf{t} \in \mathbb{R}^3$ respectively. Then, the homography relating points between images from the two cameras can be expressed as:

$$\mathbf{H} = \mathbf{R} + \mathbf{t}\,\frac{\mathbf{n}_*^\top}{d^*} \tag{1.73}$$

where d^* is the signed distance of the plane from the center of projection of camera at $\mathcal{F}^*$ (reference pose), $\mathbf{n}_*$ is the normal to the plane π containing the 3D point $\mathbf{P}^*$ expressed in $\mathcal{F}^*$. $\mathbf{R} = \exp(\theta\mathbf{u}), \theta \in]-\pi, \pi[$. In theory, with four known non-collinear corresponding points between the two images, $\mathbf{H}$ can be estimated [Hartley 04]. A survey of various planar homography estimation techniques can be found in [Agarwal 05]. Homography was used in [Deguchi 98] where it was shown that an accurate d^* is not required since it appears as a gain ratio term between the translational and rotational velocities.

Control Aspects

Homography was applied for visual servoing in [Benhimane 06] and [Benhimane 07]. When the homography between the current image and the image obtained from the reference pose $\mathcal{F}^*$ is known, then the task function (1.76) can be expressed as

$$\mathbf{e} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} = \begin{bmatrix} (\mathbf{H} - \mathbf{I})\mathbf{p}^* \\ z^{-1}(\mathbf{H} - \mathbf{H}^\top) \end{bmatrix} \tag{1.74}$$

where z^{-1} transforms a skew symmetric matrix to vector $\mathbf{v}$, the point $\mathbf{p}^* = \mathbf{P}^*/Z^* \in \mathbb{R}^2$ is obtained from the image (without measuring the 3D structure of the scene). In [Benhimane 07], $\mathbf{p}^*$ is chosen as the centre of gravity of a point cloud available in the image. In Equation (1.74), if $\mathbf{e}_1 = 0$, then the projections of the 3D point $\mathbf{P}^*$ in the current and desired images coincide $\mathbf{p}^* = \mathbf{p}$. If $\mathbf{e}_2 = 0$, the homography matrix $\mathbf{H}$ is symmetric

and allows to reach the desired pose. In [Silveira 12], a more generalized version than in Equation (1.74) was proposed.

$$\mathbf{e} = \begin{bmatrix} \mathbf{e}_\nu \\ \mathbf{e}_\omega \end{bmatrix} = \begin{bmatrix} (\mathbf{H} - \mathbf{I})\mathbf{m}^* + \rho^* \mathbf{e}_{\text{epi}} \\ \vartheta \boldsymbol{\mu} \end{bmatrix} \tag{1.75}$$

where $\mathbf{e}_\omega$ can be computed from only the homography matrix and comprises of ϑ, the angle of rotation and $\boldsymbol{\mu}$ is the axis of rotation. As for $\mathbf{m}^*$, a control point in the image close to the center of the object or region of interest is chosen and then transformed using $\mathbf{m}^* = \mathbf{K}\mathbf{p}^*$, where $\mathbf{K} \in \mathbb{R}^{3\times3}$. Finally, $\mathbf{e}_{\text{epi}} \in \mathbb{P}^2$ is the epipole while ρ^* is the projective parallax of the 3D point (provenance of $\mathbf{p}^*$) relative to the dominant plane in the image. Homography based task functions such as in Equations (1.74) and (1.75) are isomorphic to the pose, except the limitation that the rotations are constrained to $]-\pi, \pi[$. This isomorphism means that, for the task function $\mathbf{e}$ defined by

$$\mathbf{e} = \begin{bmatrix} (\mathbf{t} + (\mathbf{R} - \mathbf{I})\mathbf{P}^*)/Z^* \\ 2\sin\theta\mathbf{u} + [\mathbf{n}_*]_\times \mathbf{t} \end{bmatrix} \tag{1.76}$$

$\mathbf{e} = \mathbf{0}$ if and only if $\theta = 0$ and $\mathbf{t} = 0$. The regulation to 0 of $\mathbf{e}$ in Equations (1.76),(1.75) and (1.74) is the same as the reference frame $\mathcal{F}$ getting aligned with $\mathcal{F}^*$. The control law is then simply given by [Benhimane 07]

$$\begin{bmatrix} \boldsymbol{\nu}_c \\ \boldsymbol{\omega}_c \end{bmatrix} = - \begin{bmatrix} \lambda_1 \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \lambda_2 \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} \tag{1.77}$$

where λ_1 and λ_2 are gains that can be used to tune the speeds of convergence of $\mathbf{e}_1$ and $\mathbf{e}_2$.

Photo-Geometric Registration

In the previous section, we assumed that the homography was estimated and available for the construction of the control laws. The modus operandi of the projective geometry methods is to estimate first the projective geometric quantities and then employ them for visual servoing [Silveira 09]. There are 2 possibilities for estimation of the homography $\mathbf{H}$. First, like mentioned above, when a set of matched points between images are available. The second is by solving a dense image registration problem. For the latter, a non-trivial photogeometric image registration algorithm is used to solve for the projective homography matrix $\mathbf{G} \in \mathbb{SL}(3)$, from which $\mathbf{H}$ is later extracted.

$$\mathbf{H} = \mathbf{K}^{-1}\mathbf{G}\mathbf{K} \tag{1.78}$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ is the camera calibration matrix using the intrinsic parameters. The following general model is used in [Silveira 07a], [Silveira 09], [Silveira 12] to describe the transformation between the pixels of two images acquired by a camera.

$$\boldsymbol{\mathcal{I}}' = \boldsymbol{\mathcal{S}} \circ \boldsymbol{\mathcal{I}}\Big(w(\mathbf{g}, \mathbf{p}^*)\Big) + \boldsymbol{\beta} \tag{1.79}$$

1. The geometric part of the model in Equation (1.79) is encapsulated in the warping function w which acts on the pixel at $\mathbf{p}^*$ with geometric parameters included in $\mathbf{g}$. In

[Silveira 07a], the geometric transformation is expressed by the following general projective model

$$\mathbf{p} \propto (\mathbf{G}_\infty + \mathbf{e}\,\boldsymbol{\phi}^\top)\mathbf{p}^* + \rho^*\mathbf{e} \tag{1.80}$$

$\mathbf{g} = \{\mathbf{G}, \mathbf{e}, \rho^*\}$. In this model, $\mathbf{G} = \begin{bmatrix} \mathbf{G}_\infty & \mathbf{e} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SL}(3) \times \mathbb{R}^3$ is the projective homography that relates image points in two views, $\mathbf{e} \in \mathbb{P}^2$ is the epipole in $\boldsymbol{\mathcal{I}}$ and $\rho^* \in \mathbb{R}$ is the projective parallax defined in relation to a dominant plane of the target object [Silveira 12]. When a planar object is considered, this transformation reduces to a plane induced homography as in [Benhimane 06].

2. The photometric transformation is captured by the model parameters in $\mathbf{h} = \{\boldsymbol{\mathcal{S}}, \boldsymbol{\beta}\}$. Unlike conventional illumination models, $\boldsymbol{\mathcal{S}}$ is parametrized as a time varying surface to capture intensity variations without assumptions on imaging conditions and materials [Silveira 07b]. Equation (1.79) is a multiplicative model, the $\circ$ in $\boldsymbol{\mathcal{S}} \circ \boldsymbol{\mathcal{I}}$ is the elementwise multiplication operator. The surface $\boldsymbol{\mathcal{S}}$ can be modelled using radial basis functions or as a discretized surface representative of local illumination variations [Silveira 07b] [Silveira 10]. Global illumination variations and camera bias are represented by $\boldsymbol{\beta} \in \mathbb{R}^c$ where c is the number of image channels.

The photogeometric image registration problem can be cast as the following non-linear optimization problem.

$$\min_{\mathbf{g},\mathbf{h}} \frac{1}{2} \sum_{\mathbf{p}_i^* \in \boldsymbol{\mathcal{R}}^*} \boldsymbol{\mathcal{I}}'(\mathbf{g}, \mathbf{h}, \mathbf{p}_i^*) - \boldsymbol{\mathcal{I}}^*(\mathbf{p}_i^*) \tag{1.81}$$

where $\boldsymbol{\mathcal{R}}^* \subset \boldsymbol{\mathcal{I}}^*$ is a region of interest in the image. In fact, this is a visual tracking problem because of the estimation of transformations that align the images $\mathcal{I}(t)$ with $\mathcal{I}^*$. The challenging aspects of solving problem (1.81) are finding appropriate parametrizations for $\mathbf{g}$ and $\mathbf{h}$ (described above), reasonable initialization and indeed developing a computationally efficient method to converge to the right solution. As for solving this problem, the standard technique is to make a Taylor's expansion upto second order, apply the optimality condition (cost function derivative is 0 at the solution) and iterate until convergence [Silveira 10] [Benhimane 07]. The same registration problem has been solved in the SLAM context in [Silveira 08]. In [Benhimane 06] and [Benhimane 07], the tracking algorithms are based on the same principle and known as ESM (Efficient Second order Minimization). This necessitates the manual specification of a region of interest and subsequent visual tracking of this region (See Fig1.6). For visual servoing, the matching of these 2 regions of interest is mandatory. Inclusion of tracking procedures are accompanied by demanding computational requirements. For instance, the authors in [Silveira 12] recommend using not more than 30000 pixels (so we imagine $\mathcal{I}$ with a 150×200 image size restriction) to estimate 10 parameters to operate at a 30Hz frame rate on a 2.4GhZ processor. This raises some concern since the homography matrix alone would require 8 parameters (8 degrees of freedom).

A combination of the fundamental matrix and homography has been used in [Li 13] to control a nonholonomic mobile robot. The fundamental matrix and the camera parameters are first estimated simultaneously. This is followed by estimating the projective homography matrix, based on which a two stage control law is designed. Visual servoing

Figure 1.6 – Homography based visual servoing [Benhimane 06]. On the left is the initial image and right is the final image. The regions of interest in reference pose (red) and initial pose (green) are shown. On the right is the final image after the visual servoing where the current and desired regions of interest coincide

schemes based on homography have been used for control of mobile robot in [Fang 02] and for aerial vehicles in [Gonçalves 10] and [dePlinval 11].

Visual Servoing based on the Trifocal Tensor

The trifocal tensor $\mathcal{T}$ is a $3 \times 3 \times 3$ array of 27 elements that links points (or lines) from two known images to a third image. It is dependent only upon the camera projection matrices and is independent of the scene itself. This tensor which relates the geometry of three views is dealt with in detail in [Hartley 04]. In [Shademan 10], the trifocal tensor that relates the initial image, current image and the desired image has been used for 6DOF visual servoing. The visual features are a subset of the set of 27 elements of the trifocal tensor $\mathbf{s} \subset \mathcal{T}$ and chosen according to the type of motions considered. The interaction matrix is estimated numerically and used in the control law. In [Shademan 10], a set of six non-coplanar points are matched across the three views and the algorithm in [Hartley 00] is used to estimate $\mathcal{T}$.

The 1D trifocal tensor relating the bearing measurements is used for a mobile robot path following task in [Aranda 13]. Here, the trifocal tensor is not used directly but rather the geometry, namely the translation between the poses and the orientation is retrieved from it. The final control law is a simple function of these geometrical parameters obtained from the trifocal tensor. This method is an uncalibrated visual servo method and handled cases like large rotations considered difficult in traditional visual servoing. An analytical form of the interaction matrix is not available for the trifocal tensor and visual tracking of the non-coplanar interest points was still necessary. In the following section, we will examine the photometric or intensity-based visual servoing methods.

1.3 Photometric Visual Servoing Methods

A camera image is a spatial sampling of brightness variations that depend on ambient illumination, reflectance properties and geometry of the elements in the imaged scene [Sonka 14]. In feature-based methods discussed in Section1.2, we observe that the image simply gets reduced to a set of points or geometrical primitives once the image processing

steps have been done. Clearly, rich information that is available in the image as intensity patterns is simply lost and not exploited at all. Photometric visual servoing methods avoid usage of extracted geometric features and use intensity information available from the image. Since no tracking and matching procedures are involved, these methods are also known as correspondanceless visual servoing methods. In general, we call the methods that use intensity information from the image as photometric visual servoing methods.

1.3.1 Visual Servoing on Reduced Dimensional Subspaces

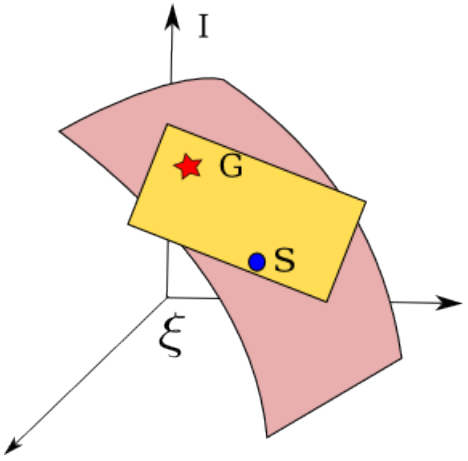


Figure 1.7 – Visual Servoing on reduced dimensional subspaces. $\boldsymbol{\xi} \in \mathbb{R}^6$ is the robot pose and $\mathbf{I} \in \mathbb{R}^N$ is the image acquired at some $\mathbf{r}$. Learning the interaction matrix $\mathbf{L}$ is equivalent to finding the coefficients of the tangent plane to the non-flat surface in $\mathbf{r}$-$\mathbf{I}$ space

The visual servoing methods proposed in [Nayar 96], [Deguchi 97] [Deguchi 00] can be classified under this category. It was [Nayar 96] which advocated finding visual representations that encode appearance and not shape. It was one of the earliest works which used intensity data from the image for visual servo control. An image $\mathbf{I}$ acquired for the purpose of visual servoing depends on the robot pose $\boldsymbol{\xi}$ and can be represented as $\mathbf{f} : \boldsymbol{\xi} \rightarrow \mathbf{I}$. This is an inherently high dimensional mapping with $\boldsymbol{\xi} \in \mathbb{R}^6$ and $\mathbf{I} \in \mathbb{R}^N$, N is the number of pixels. The tuple $(\boldsymbol{\xi}, \mathbf{I})$ can be considered as a data point in $6 + N$ dimensional space. Typically, these methods sample a set of images acquired from all around the goal pose in the robot workspace. Then, they use principal component analysis (PCA) to transform $\mathbf{I}$ to $\mathbf{p}$. $\mathbf{f}_{pca} : \mathbf{I} \rightarrow \mathbf{p}$ where $\mathbf{p} \in \mathbb{R}^M$ where $M \ll N$. This is similar to geometric feature-based methods which also derive a set of visual features $\mathbf{s}$ from $\mathbf{I}$, a difference being that here the features extracted using PCA are not geometrical.

Precisely, the vector space spanned by the basis set $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_m\}$, $\mathbf{e}_i \in \mathbb{R}^m$ whose linear combination can produce $\mathbf{p}$ is known as eigenspace. Now, a small change in the pose $\boldsymbol{\xi}$ manifests in the image and therefore in the reduced representation $\mathbf{p}$ via the straightforward relation

$$\mathrm{d}\mathbf{p} = \mathbf{L}_\mathbf{p} \mathrm{d}\boldsymbol{\xi} \tag{1.82}$$

where $\mathbf{L}_\mathbf{p}$ is estimated numerically either online [Deguchi 96] or simply at the goal pose [Deguchi 97]. This amounts to finding the coefficients of the tangent plane (rectangular

patch in Fig.1.7) to the surface. The underlying surface in $\boldsymbol{\xi} - \mathbf{I}$ space is not flat which causes a non-linearity. In the case of this space being flat, there will be a linear change in $\mathbf{I}$ with change in $\boldsymbol{\xi}$. As reported in [Deguchi 00], the determination of $\mathbf{L_p}$ is cumbersome and prohibitive with increase in number of degrees of freedom.

As for the visual servoing, the remapping from $\mathbf{d}(t)$ to $\mathbf{r}(t)$ has to be performed. The manifold parametrized by the motion variables (degrees of freedom of the robot displacement in the learning process). The control action is finally given by

$$\mathbf{v} = -\lambda \Delta \boldsymbol{\xi} = -\lambda \left(\mathbf{L_p}\right)^{+} \left(\mathbf{p} - \mathbf{p}_g\right) \tag{1.83}$$

just like in classical visual servoing. The experimental results were limited to 2DOF displacements. When displacements involve more degrees of freedom, acquiring images for learning becomes cumbersome. In direct contrast to feature-based methods, featureless methods avoided the extraction of points or other intermediate geometric primitives from the image. This is a learning-based visual servoing approach with no analytical modelling of image features or the interaction matrix. This makes it difficult to study failure conditions or characterize stability.

1.3.2 Direct Visual Servoing

In direct visual servoing approaches, no intermediate geometric entities are tracked or matched in the image. The intensity of the pixels are used in the control law. The problem of finding the appropriate control inputs in visual servoing is in fact a problem of non-linear optimization.

$$\hat{\boldsymbol{\xi}} = \arg\min_{\boldsymbol{\xi}} \mathcal{C}(\mathbf{I}(\boldsymbol{\xi}), \mathbf{I}^*) \tag{1.84}$$

$$\text{with} \quad \mathcal{C}(\boldsymbol{\xi}) = \frac{1}{2} \left\| \mathbf{e}(\mathbf{I}(\boldsymbol{\xi}), \mathbf{I}^*) \right\|^2 \tag{1.85}$$

is the cost function to be minimized, and is nothing other than the Lyapunov function for stability, $\boldsymbol{\xi}$ denotes the robot pose as usual while $\mathbf{e}$ is the error measure calculated using the current image and the image learnt from the desired pose ($\mathbf{I}^* = \mathbf{I}(\boldsymbol{\xi}^*)$). In the case of direct visual servoing methods, the error $\mathbf{e}$ is not from measurements involving extracted geometric features in the image. Instead, this error is photometric, that is, based on the intensity distribution in the image.

1.3.2.1 Pure Photometric Visual Servoing

In pure photometric visual servoing [Collewet 11], the error function is directly defined as difference in intensities (luminance) between the images acquired at the current and desired poses $\mathbf{e} = \mathbf{I}(\boldsymbol{\xi}) - \mathbf{I}(\boldsymbol{\xi}^*)$. The cost function to be optimized is

$$\hat{\boldsymbol{\xi}} = \arg\min_{\boldsymbol{\xi}} \sum_{\mathbf{x}} \left[I(\mathbf{x}(\boldsymbol{\xi})) - I(\mathbf{x}(\boldsymbol{\xi}^*)) \right]^2 \tag{1.86}$$

where $\mathbf{x} = (x, y) \in \mathbb{R}^2$ refers to the pixel indexing variable. The error norm is exactly the SSD (Sum of the Squared Differences) function used for image registration in the KLT algorithm [Baker 04].

(a) Initial image (b) Final Image (c) Difference image at initial pose (d) Difference image at final pose

Figure 1.8 – Pure luminance-based Visual Servoing. (Figures adapted from [Collewet 11])

The set of visual features is the entire set or a subset of the pixels observed in the image.

$$\mathbf{s} = \mathbf{I} = \begin{bmatrix} I(\mathbf{x}_1) \\ I(\mathbf{x}_2) \\ \vdots \\ I(\mathbf{x}_N) \end{bmatrix}$$

where the number of pixels is denoted by N. When only a specific subset of the pixels available in the image is used, this subset has to be made available on a frame-by-frame basis. The interaction matrix related to pixel luminance can be obtained from the derivative of $\mathbf{s}$

$$\dot{\mathbf{I}} = \mathbf{L_I} \mathbf{v}_c = \left[\mathbf{L}_I(\mathbf{x}_1), \ldots, \mathbf{L}_I(\mathbf{x}_n) \right]^\top \mathbf{v}_c \tag{1.87}$$

In [Collewet 11], the interaction matrix of the luminance of an isolated image pixel was developed under the classical brightness constancy assumption [Horn 81]. According to this constraint, the brightness of a moving image point is constant between small time interval δt.

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \tag{1.88}$$

The first order Taylor's expansion of this equation gives

$$\nabla I_x \dot{x} + \nabla I_y \dot{y} + \dot{I} = 0 \tag{1.89}$$

where $\nabla I = \begin{bmatrix} \nabla I_x & \nabla I_y \end{bmatrix}^\top = \begin{bmatrix} \frac{\partial I}{\partial x} & \frac{\partial I}{\partial y} \end{bmatrix}^\top$ is the gradient of the image pixel at (x,y). The interaction matrix of the luminance of pixel (x, y) can then be obtained in a straightforward manner.

$$\mathbf{L}_I(\mathbf{x}) = -\nabla I^\top \mathbf{L_x} \tag{1.90}$$

where $\mathbf{L_x}$ is the interaction matrix of an image point that was developed in Sec.1.2.1 and given in Equation (1.50). An important remark is that the depth Z of the image point is still present, propagated from $\mathbf{L_x}$ to $\mathbf{L}_I$. Besides, the brightness constancy hypothesis can be violated even in the case of Lambertian surfaces when there is relative movement between the surfaces and lighting sources. In this case, obtaining the interaction matrix becomes more difficult. The interaction matrix is obtained using the Phong Illumination Model [Phong 75] and is presented in complete details in [Collewet 08b].

It is well-known that convergence to the right solution in non-linear optimization problems depends on the distance between the solution and the initial starting state. Naturally,

visual servoing inherits the advantages and drawbacks depending on the method adopted to solve the optimization. Optimization problems such as in Eq(1.85) can be solved by using a variety of iterative non-linear optimization methods [Ruszczyński 06]. The essence of these methods is to choose a specific direction of descent in the solution space such that the optimization objective is attained. The control action then takes the form

$$\mathbf{v} = \lambda\, \mathbf{d}(\boldsymbol{\xi}) \tag{1.91}$$

where $\mathbf{d}(\boldsymbol{\xi})$ is the direction of descent and $\boldsymbol{\xi}$ denotes robot pose. Following the Levenberg-Marquardt method of descent, we deduce the following control law.

$$\mathbf{v} = -\lambda \left(\mathbf{H} + \mu diag\left(\mathbf{H}\right)\right)^{-1} \hat{\mathbf{L}}_{\mathbf{I}}^{+} \left(\mathbf{I}(\boldsymbol{\xi}) - \mathbf{I}(\boldsymbol{\xi}^{*})\right) \tag{1.92}$$

where λ and μ are positive scalars. $\mathbf{H}$ is an approximation of the Hessian matrix given by $\mathbf{H} = \mathbf{L}_{\mathbf{I}}^{\top}\mathbf{L}_{\mathbf{I}}$, $\mathbf{L}_{\mathbf{I}}$ being the interaction matrix at any pose $\boldsymbol{\xi}$.

- Due to non-utilisation of intermediate measures from the image and use of redundant feature set, this method benefits from high accuracies at convergence.
- The convergence domain reduces with the amount of error in planar rotation between the images of the starting and final robot poses [Collewet 11]. This can be explained partly by the high non-linearity in Equation (1.90). On the contrary, in the case of classical visual servoing methods, simpler solutions (Gauss-Newton method) are sufficient for the servoing.

The work in [Collewet 11] showed that the intensity of the image pixels can be used to control the motions of a robot. Further, visual servoing was demonstrated without any feature detection and tracking. But the image processing step of calculating the gradients in (1.90) was necessary. As in other IBVS methods, only local stability can be proved. Further, the method suffers from the problem of lighting variations which directly alter the minimum of the cost function.

1.3.2.2 Visual Servoing based on Information Theoretic Measures

Information Theory was a field born after Claude Shannon published his seminal work *"A Mathemetical theory of Communication"* in 1948 [Shannon 48]. Information theory deals with quantification of information (in the Shannon sense) and was developed for the purpose of data storage and communication. In fact, each pixel of the image can be regarded as a random variable which takes values from a finite set. So, the image can be thought of as a probability distribution. There are many information theoretic measures that study the amount of shared information between probability distributions of random variables. In the following, we explain methods in visual servoing which are based on tools borrowed from information theory.

Mutual Information based Visual Servoing The main interest in using mutual information for visual servoing stems its independence on the imaging modality. Given that there is a sufficient overlap between the two images, this measure is a reliable similarity

indicator regardless of the actual imaging modality used [Dame 10] [Dame 11]. MI based visual servoing aims to solve the following optimization problem

$$\hat{\boldsymbol{\xi}} = \arg\max_{\boldsymbol{\xi}} \mathrm{MI}(\mathbf{I}(\boldsymbol{\xi}), \mathbf{I}^*) \tag{1.93}$$

where $\mathrm{MI}(\mathbf{I}, \mathbf{I}^*)$ is the mutual information between the current and desired images. MI is computed as a relationship between the marginal and joint entropies of the intensity distribution in the current and desired images.

$$\mathrm{MI}(\mathbf{I}, \mathbf{I}^*) = H(\mathbf{I}(\boldsymbol{\xi})) + H(\mathbf{I}^*) - H(\mathbf{I}(\boldsymbol{\xi}), \mathbf{I}^*) \tag{1.94}$$

The marginal entropy of an image $\mathbf{I}$ is denoted $H(\mathbf{I})$ and can be calculated as follows

$$H(\mathbf{I}) = -\sum_{i=0}^{N_I} p_{\mathbf{I}}(i) \log(p_{\mathbf{I}}(i)) \tag{1.95}$$

where i is one of the allowed values of the intensity from the set $\mathcal{I} = [0, N_I]$, $p_{\mathbf{I}}(i)$ is the probability of occurence of the intensity value i from sample space of $\mathcal{I}$. It can be described by a simple mathematical formula

$$p_{\mathbf{I}}(i) = \frac{1}{N_I} \sum_{\mathbf{x}} \delta(i - I(\mathbf{x}, \boldsymbol{\xi})) \tag{1.96}$$

δ is the Kronecker delta function whose output is i when $i = \mathbf{I}(\mathbf{x})$ and 0 otherwise. In a similar way, the joint entropy is given by

$$H(\mathbf{I}, \mathbf{I}^*) = -\sum_{i=0}^{N_I} \sum_{j=0}^{N_I} p_{\mathbf{I}\mathbf{I}^*}(i, j) \log(p_{\mathbf{I}\mathbf{I}^*}(i, j)) \tag{1.97}$$

Analogous to (1.96), $p_{\mathbf{I}\mathbf{I}^*}(i, j)$ describes the joint probability of occurence of the pixel intensity pair (i, j) and has an elegant formula

$$p_{\mathbf{I}\mathbf{I}^*}(i, j) = \frac{1}{N_I} \sum_{\mathbf{x}} \delta(i - I(\mathbf{x}, \boldsymbol{\xi}))\, \delta(j - I^*(\mathbf{x})) \tag{1.98}$$

The MI between two images is maximised when maximum information is shared between them. In visual servoing, this situation occurs when the camera pose reaches its goal position, aligning the camera image with the desired one. When this alignment is achieved, the gradient of the MI becomes null. Therefore to maximize MI, the task function to be regulated to 0 is the gradient of MI, given by

$$\mathbf{e} = \sum_{i,j} \frac{\partial p_{\mathbf{I}\mathbf{I}^*}}{\partial \boldsymbol{\xi}} \left(1 + \log\left(\frac{p_{\mathbf{I}\mathbf{I}^*}}{p_{\mathbf{I}}}\right)\right) \tag{1.99}$$

The visual servoing control law is then given by

$$\mathbf{v}_c = -\lambda \mathbf{L}_{\mathbf{e}}^{-1} \mathbf{e} \tag{1.100}$$

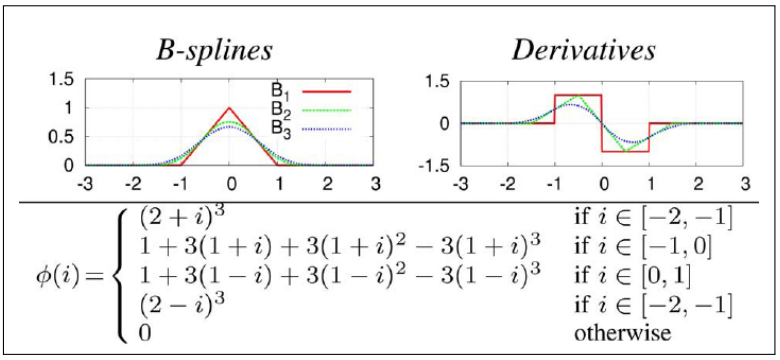


Figure 1.9 – B-spline functions used in MI *based Visual Servoing [Dame 10]*

where $\mathbf{L_e}$ is the interaction matrix of $\mathbf{e}$ in Equation (1.99). To compute $\mathbf{L_e}$, $\frac{\partial p_{\mathbf{II}^*}}{\partial \boldsymbol{\xi}}$ in (1.98) and thereby δ should be twice differentiable. This is achieved by replacing δ in Eqs (1.96) and (1.98) with a cubic B-Spline function ϕ (see Figure 1.9). This smoothes the MI function which otherwise would exhibit sharp peaks other than at convergence. Also, the maximum number of bins (the original number of intensity levels $N_I = 255$) is transformed to $N_c = 8$ bins by scaling of the individual pixel intensities. As detailed in [Dame 11], these steps serve to smooth the MI function and facilitate a simple computation. The above are two vital steps in the adaptation of the original mutual information measure for visual servoing.

Figure 1.10 – MI *based Navigation Task in [Dame 11] with Multimodal information*

Besides being robust to occlusions and lighting variations, MI seems to be an attractive feature when more than a single imaging modality is used. In Figure 1.10, a navigation task from [Dame 11] has been shown where the images for servoing are acquired from a monocular camera whereas the reference images in the learning phase are from a geographical map. A comparison between MI, pure photometric (SSD) and SIFT feature based approaches is made in [Dame 11]. The convergence of the MI based method degrades with respect to increased rotation of the camera around the optic axis. In this case,

the failure is attributed to less shared information between the image at the pose with increased rotation and images acquired during the visual servoing. Hence, we can say that more research is necessary to find new ways to design visual servoing systems that work in spite of missing or less shared information.

Visual Servoing using the Sum of Conditional Variance

The sum of conditional variance is another information theoretic measure which is simpler in its computation when compared with MI and is introduced for visual servoing in [Delabarre 12].

$$\mathrm{SCV} = \sum_{\mathbf{x}} (I(\mathbf{x}) - \hat{I}(\mathbf{x})) \tag{1.101}$$

The novelty introduced in this approach lies in the adaptation of the reference image in relation to the currently observed image. That is,

$$\hat{\mathbf{I}}(\mathbf{x}) = \mathcal{E}(\mathbf{I}(\mathbf{x})|\mathbf{I}^*(\mathbf{x})) \tag{1.102}$$

where $\mathcal{E}$ is the expectation operator in probability theory and can be computed from the joint probability distribution between $\mathbf{I}$ and $\mathbf{I}^*$. The control law is developed using the Levenberg-Marquardt approach. It is given by

$$\mathbf{v}_c = -\lambda(\mathbf{H}_{\mathrm{S}} + \mu \mathrm{diag}(\mathbf{H}_{\mathrm{S}}))^{-1} \mathbf{L}_{\mathrm{SCV}} (\mathbf{I}(\boldsymbol{\xi}) - \hat{\mathbf{I}}) \tag{1.103}$$

where $\mathbf{H}_{\mathrm{S}} = \mathbf{L}_{\mathrm{SCV}}^{\top} \mathbf{L}_{\mathrm{SCV}}$. As in the case of mutual information, the task function to be regulated is the gradient of the SCV. For an image point $\mathbf{x}$, we can write

$$\mathbf{L}_{\mathrm{SCV}}(\mathbf{x}) = -\nabla \mathbf{I}^{\top} \mathbf{L}_{\mathbf{x}} \tag{1.104}$$

where $\nabla \mathbf{I}^{\top} \in \mathbb{R}^{1\times 2}$ is its image gradient and $\mathbf{L}_{\mathbf{x}} \in \mathbb{R}^{2\times 6}$ is its interaction matrix. By stacking the $\mathbf{L}_{\mathrm{SCV}}(\mathbf{x})$ of all the image points (lets say N), the $\mathbf{L}_{\mathrm{SCV}}(\mathbf{I}) \in \mathbb{R}^{N\times 6}$ of the image can be obtained. The computationally expensive calculations involving B-splines in MI are absent in the case of SCV. The computations in the case of SCV are simpler than in case of MI in Equation (1.99). On the other hand, the SCV is not multimodal and is affected by local illumination variations. But it is comparable to pure photometric visual servoing [Collewet 11] under nominal conditions and better when the illumination variations are global. All these findings are reported in [Delabarre 12].

1.3.3 Kernel-Based Visual Servoing

Kernel-based Visual Servoing (KBVS) is a correspondanceless visual servoing method and does not use any geometric features or visual tracking. Inspired by kernel based visual tracking techniques, this approach was proposed in [Kallem 07] with the aim of unifying the tracking and control procedures. The KBVS approach was based on the projection of the image onto a spatial kernel. A kernel is a piece-wise continuous function $K : \pi \rightarrow \mathbb{R}$ that produces a measurement $\nu \in \mathbb{R}$ when values of spatially distributed signal are projected onto it.

$$\nu(t) = \iint_{\mathcal{I}} K(\mathbf{w}) \; s(\mathbf{w}, t) \; \mathrm{d}\mathbf{w} \tag{1.105}$$

$\nu(t)$ is known as the kernel measurement or kernel projected value, $s(\mathbf{w}, t)$ is the signal of interest and $\mathbf{w} \in \mathbb{R}^2$ is the spatial indexing variable. Image moments are closely related to this approach with the image as the signal, projected to a polynomial kernel. The differences are explained in more detail in Chapter 2. Like in classical visual servoing, minimization of the difference in error between the kernel projected values at the desired ν^* and current poses $\nu(t)$ is sought. The Lyapunov function is defined based on errors in the kernel projected values: $V = \frac{1}{2}(\nu - \nu^*)^2$ whose derivative is given by:

$$\dot{V} = (\nu - \nu^*)\dot{\nu} = (\nu - \nu^*)\frac{\partial \nu}{\partial \boldsymbol{\xi}}\dot{\boldsymbol{\xi}} \tag{1.106}$$

where $\frac{\partial \nu}{\partial \boldsymbol{\xi}}$ denotes the variations in the kernel-projected value with respect to robot motion. This is analogous to the classical visual servoing methods where the visual feature is differentiated analytically to obtain their interaction matrix $\dot{\mathbf{s}} = \mathbf{L}_\mathbf{s}\mathbf{v}_c$. To obtain $\frac{\partial \nu}{\partial \boldsymbol{\xi}}$, $\nu(t)$ should be differentiable which dictates that the kernel K be smooth. The control input $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is chosen such that the Lyapunov derivative $\dot{V} < 0$ to satisfy the stability criterion as in the classical visual servoing approach. The KBVS control inputs take the below general form:

$$\mathbf{u}(t) = -(\nu - \nu^*) \iint_{\mathcal{I}} [\mathbf{K}'\mathbf{s}'(\mathbf{w}, \boldsymbol{\xi}(t))d\mathbf{w}] \tag{1.107}$$

where $\mathbf{K}' = \left(\frac{\partial K}{\partial \mathbf{w}}\right)^\top$ (so the kernel should of course be differentiable with respect to the spatial variables $\mathbf{w}$), $\mathbf{s}'$ is the signal obtained after a specific transformation of the goal image. In fact, separate controllers are synthesized in [Kallem 07] for planar translation ($\mathbf{u}(t) \in \mathbb{R}^2$), scaling ($u(t) \in \mathbb{R}^1$) or planar rotation ($u(t) \in \mathbb{R}^1$). In any case, the choice of $\mathbf{u}(t)$ needs to ensure that $\dot{V} < 0$, at least in a neighbourhood of the goal configuration $\boldsymbol{\xi}^*$. To control the planar translations, the signal of interest is the acquired image itself. For control of translations along and rotations around the optic axis, the signal of interest was the magnitude of the spatial Fourier Transform (FT) of the image, that is, in Equation (1.105), $s(\mathbf{w}, t) = \|\mathrm{F}(u, v)\|$ where F is the FT of the acquired image indexed by u and v. The magnitude of the FT was chosen because it is invariant to planar translations. For control of planar translations, the kernel was either a 2D gaussian or two separate 1D gaussians, each for x and y directions placed at each pixel location in the image. To control translation along the optic axis, a 1D gaussian was used. These kernels were finitely truncated but the numerical details have not been provided in [Kallem 07]. As for control of rotations around the optic axis, a rotationally assymetric kernel was chosen.

$$K(u, v) = r^2 - (u^2 + v^2)\sin^2\left(\arctan(\frac{v}{u})\right) \tag{1.108}$$

where $\arctan(\frac{v}{u})$ is the orientation of each pixel location and $r \in \mathbb{R}$ is a constant radius chosen offline. For each of the above cases however, there are some operational difficulties in the deployment of this method for visual servoing.

- Signal and kernel combinations for controlling only 4DOF were proposed: planar translations, depth and roll motions. Control of rotational degrees of freedom perpendicular to the optic axis was ignored.

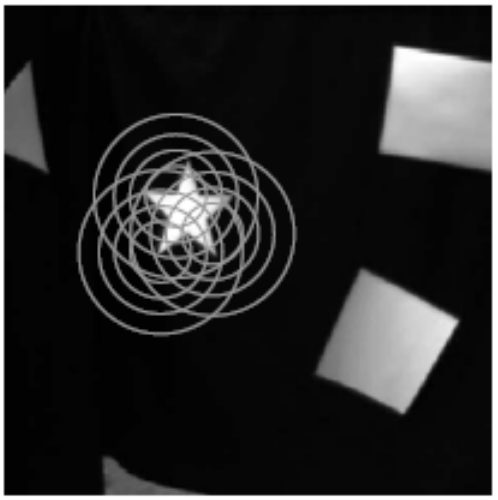


Figure 1.11 – Kernel placement in the acquired camera image (reproduced from [Kallem 09])

- One of the shortcomings of this method is its usage of the frequency domain feature, the FFT, the computation of which is prone to ringing effects. The FFT is exact only in the case of well-segmented images.

- Separate controllers were synthesized for planar translations, depth and roll motions. Since the kernel projected values are not decoupled, these controllers had to be applied sequentially. In [Kallem 07], for control of 3D translations, it is recommended to control depth motions before planar translation. Likewise, for control in $SE(2) + Depth$, convergence in depth should precede control of roll motions. This is because the kernel projected measurement for control of roll motions is not invariant to scaling transformation.

- An improper kernel selection results in instability of the system and convergence problems [Kallem 09]. There might exist a kernel placement such that $\dot{V} \geq 0$ holds true for multiple configurations of the robot in the neighbourhood of the equilibrium state [Swensen 10]. Contradictorily, for a stable convergence to the desired equilibrium, what is envisaged is $\dot{V} < 0$ for all configurations other than the goal and $\dot{V} = 0$ only at the desired goal configuration.

Alternatively, in the KBVS method, the interaction matrix could have been derived analytically with the kernel functions in question. This way the coupling properties between each kernel measure could have been taken into account.

1.4 Conclusion

Visual Servoing has been researched since several decades and from different perspectives. There are plenty of methods solely based on the visual features used in the control law. The choice of the visual features ultimately decides the successful completion of the task and behaviour in the image and Cartesian space. Therefore, deciding the visual feature is an extremely critical aspect of visual servoing. Almost all the geometric methods

rely on the availability of one or more of geometric entities like points or straight lines in the scene. This necessitates the integration of matching (establishment of correspondances) and visual tracking procedures for the visual servoing. Motivated by the objective to free the visual servoing control from geometric feature extraction, visual servoing using the intensity directly were developed. Later, information theoretic measures like mutual information(MI) and Sum of Conditional Variance(SCV) were proposed, each having its specific advantages and drawbacks. A positive transition from purely geometric features towards features which incorporate intensity available in the image began occurring. Existing methods like KBVS and those based on information-theoretic measures utilise the intensity information in the image. They introduce non-linearity in the control laws which limit their convergence domain. Visual servoing with image moments-based features were superior to traditional methods based on tracking geometric primitives since they did not require any tracking or matching steps. But state of the art moments-based methods still depend on the availability of a set of points or a well-segmented object in the image. We see that in spite of the several positive advances made, the goal of finding an appropriate set of features for visual servoing is not completely solved. This leaves ample space for improvements and contributions in this area.

Chapter 2

Tunable Visual Features for Image Moments-based Visual Servoing

In this chapter, the state of the art in image moments-based visual servoing methods is recalled. Then, the idea of shifted moments [Tamtsia 13b] is described. We argue that the default choice opted for with the shifted moments is not the best choice for all moment invariants-based visual features. We propose improvements to the shifted moments which allow to redefine them in such that they are optimal with respect to specific criteria. A few criteria are then proposed to select the shift points with respect to which the moments are defined. The results are discussed with the obtained improvements.

2.1 Introduction to Image Moments

The n^{th} moment of a real valued distribution function $f(x)$ about a specific value $x = v$ is defined by $\int_x (x - v)^n f(x) \mathrm{d}x$. When $v = 0$, this moment is called the raw moment or crude moment. When the moments are computed about the mean of the distribution function $v = \mu$, they are known as centred moments. Let us consider a more general distribution function $f(\mathbf{x})$ taking vector-valued arguments and mapping them to scalar values from a finite support $\mathbf{\Omega}$. The general definition of the moments of $f(\mathbf{x})$ is defined by

$$m_{\mathbf{p}} = \int \cdots \int_{\mathbf{\Omega}} \Phi_{\mathbf{p}}(\mathbf{x}) f(\mathbf{x}) \, \mathrm{d}x_1 \ldots \mathrm{d}x_n \tag{2.1}$$

where $\mathbf{x} = \{x_1, x_2 \ldots x_n\} \in \mathbb{R}^n$ refers to the spatial index. The basis function Φ is a continuous function of $\mathbf{x}$ parametrized by a vector of variables $\mathbf{p}$. Technically moments are obtained by projecting the distribution function over the basis function. Moments are a compact representation of the original information signal. So, scalar moments of a few orders are capable of capturing the important characteristics of any information signal.

The geometric moments of a 2D distribution function $f(x, y)$ can be expressed as

$$m_{pq} = \iint_{\Omega} x^p y^q f(x, y) \mathrm{d}x \mathrm{d}y \tag{2.2}$$

where Ω is the domain of integration, $p+q$ is the order of the moment. For a piecewise continuous function $f(\mathbf{x})$ with a bounded support, geometric moments of all orders exist and are finite. This is known as *Existence* theorem [Mukundan 98]. Then, given such moments $\{m_{pq}\}$, the original function from which these moments were generated can be reconstructed (in various degrees of accuracy of course depending on the number of moments). This assertion is known as the *Uniqueness theorem* [Flusser 09]. As we will see in this chapter, simple functions of lower moments have an intuitive geometrical interpretation.

In visual servoing, the information signal is indeed the image acquired from the camera from which the moments are computed. That is, $f(x,y)$ in Equation (2.2) is the intensity distribution from the image. In the following, we will call the moments in Equation (2.2) as the raw moments. Image moments have several useful properties which make them useful for visual servoing. One of the most important properties being the invariance of specific combinations of moments to specific geometrical transformations in the image.

Applications Image moments have been used for industrial grade alignment tasks in [Wang 08]. In medical robotics, 2D moments were used in the guidance of ultrasound probes [Mebarki 10]. In [Dani 13], image moments are used for 3D structure estimation to perform SLAM with a quadrotor platform. They been applied to the control of underactuated systems like UAVs [Jabbari 12] [Jabbari 14].

Image moments have been studied in [Chaumette 04] and [Tahri 05a], where the range of function $f(x,y)$ in Equation (2.2) was restricted to take on only two values : 0 for the image background regions and 1 for the image regions of interest corresponding to the object projection or corresponding to a set of tracked image points. So, the segmentation and tracking of that specific region is necessary. Existing methods based on image moments depend on the availability of a set of discrete points or a well-segmented region in the scene visible at all times of operation of a robot.

2.2 Modelling the time variation of Moments

Before using any feature for visual servoing, it is necessary to determine its interaction matrix. Next, we will develop systematically the interaction matrix employing two different methods to derive the variations of the image moments. The moment equation is characterized by a time-varying domain $\Omega(t) \subset \mathbb{R}^2$ and the object is assumed to be within this domain.

2.2.1 Using time variations of an image contour point

We will briefly recall the method presented in [Chaumette 04] to obtain the interaction matrix. There is a target object $\mathcal{O}$ whose projection in the camera image is $\Omega(t)$. Let its contour be denoted as $\mathcal{C}(t)$ (See Fig 2.1).

In Equation(2.2), let us denote $x^p y^q = k(x,y)$, the kernel of projection. In the following derivation, the intensity of the pixels are not considered. A binary image is always assumed to be available, over which the moments are computed. In this case,

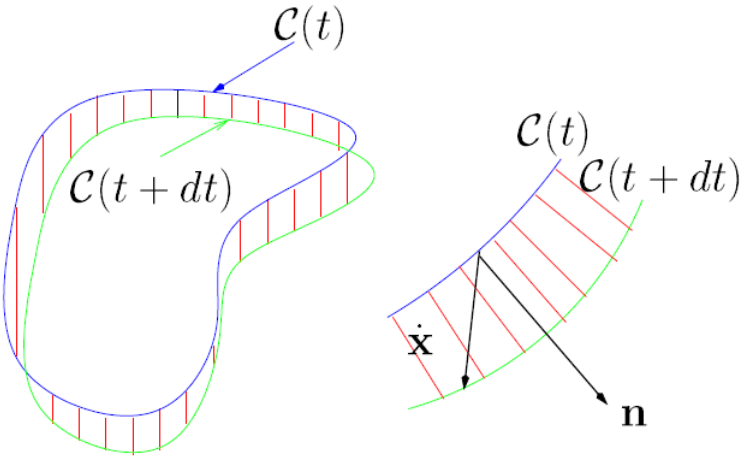


Figure 2.1 – Illustration of time variations of an image contour [Chaumette 04]

$f(x,y) = 1, \forall (x,y) \in \Omega$ and 0 elsewhere. Hence, Equation(2.2) can can re-written as

$$m_{pq} = \iint_{\Omega(t)} k(x,y) \mathrm{d}x\mathrm{d}y \tag{2.3}$$

Then, the derivative of the moments is expressed as

$$\dot{m}_{pq} = \oint_{\mathcal{C}(t)} k(x,y)\, \dot{\mathbf{x}}^{\top}.\mathbf{n}\, \mathrm{d}l \tag{2.4}$$

where $\mathbf{x} = (x,y)$, $\dot{\mathbf{x}}$ is the velocity of a point on the contour $\mathcal{C}(t)$ in the image, $\mathbf{n}$ is the unit normal at $\mathbf{x}$ and $\mathrm{d}l$ is an infinitesimal element of the contour $\mathcal{C}(t)$. There are two important conditions to further develop the above derivative.

1. $\mathcal{C}(t)$ is continuous by parts
2. the vector quantity $k(x,y)\dot{\mathbf{x}}$ is tangent to $\Omega(t)$ and continuously differentiable $\forall \mathbf{x} \in \Omega(t)$

Provided the above conditions hold true, Green's theorem can then be directly used to obtain

$$\dot{m}_{pq} = \iint_{\Omega(t)} \mathrm{div}(k(x,y)\, \dot{\mathbf{x}})\mathrm{d}\mathbf{x} \tag{2.5}$$

We can arrive at the same result in Equation (2.5) by using a different method which will be described next.

2.2.2 Using the method of Gâteaux derivatives

The method of Gâteaux derivatives (or Eulerian derivatives) is used in [Tamtsia 13a] to derive the moments' variation , adopting the method presented in [JehanBesson 03]. Unfortunately, some errors seem to have crept into the developments. Here, we present a

clearer version and show that same results as the method of Green's Theorem in 2.2.1 can be obtained. The moments equation in (2.2) can be written as

$$m_{pq} = \iint_{\Omega(t)} k(\mathbf{x}, \Omega) \mathrm{d}\mathbf{x} \tag{2.6}$$

where the polynomial basis functions and the distribution function are coupled into a single functional $k(\mathbf{x}, \Omega) = x^p y^q f(x, y)$. The main difficulty is that the domain $\Omega \in \mathcal{U}$ and the space of such domains $\mathcal{U}$ is not a vector space. To get the derivative of the moments, clearly differentiability of a function with respect to a domain needs to be established. This is done by introducing an application $T : \Omega \to \mathbb{R}^n$ and $T \in \mathbf{A}$, a set of applications. Further, $T \in \hat{\mathcal{A}}$, whose members are endowed with the following properties.

- T and its inverse are continuous, one to one functions.
- Both T and its inverse belong to a Banach space : $T, T^{-1} \in W^{(1,\infty)(\Omega,\mathbb{R}^n)}$.

Then, given any shape function $S : \mathcal{U} \to \mathbb{R}^+$ and an application $T \in \hat{\mathcal{A}}$, we can think of the shape function as a functional of the domain, denoted $\hat{S}(T(\Omega))$ [Paragios 06]. This allows us to define the derivative with respect to a time-varying domain.

In order to compute the Gâteaux derivative with respect to T, we introduce a family of deformations $T(\tau)_{\tau \geq 0}$ such that $T \in \mathbf{A}$ and is a smooth and bijective map. Then for any point $\mathbf{x} = (x, y) \in \Omega$, we can write [Paragios 06]

$$\mathbf{x}(\tau) = T(\tau, \mathbf{x}) \quad \text{with} \quad T(0, \mathbf{x}) = \mathbf{x} \tag{2.7a}$$

$$\Omega(\tau) = T(\tau, \Omega) \quad \text{with} \quad T(0, \Omega) = \Omega \tag{2.7b}$$

We can then define the velocity vector field $\mathbf{V}$ corresponding to $T(\tau)$ as

$$\mathbf{V}(\tau, \mathbf{x}) = \frac{\partial T}{\partial \tau}(\tau, \mathbf{x}) \quad \forall \mathbf{x} \in \Omega \ \ \forall \tau \geq 0 \tag{2.8}$$

To get the derivative $\dfrac{\partial T}{\partial \tau}$ in Equation (2.8), we introduce a simple first order variation for $\mathbf{x}$ like in [JehanBesson 03].

$$T(\tau, \mathbf{x}) = T(0, \mathbf{x}) + \tau \frac{\partial}{\partial \tau}(T(0, \mathbf{x})) \tag{2.9}$$

Using (2.7a) and (2.7b) and taking $\dfrac{\partial \mathbf{x}}{\partial \tau} = \mathbf{V}(\mathbf{x})$, we can rewrite this as

$$T(\tau, \mathbf{x}) = \mathbf{x} + \tau \mathbf{V}(\mathbf{x}) \tag{2.10}$$

The Gâteaux derivative of $m_{pq}(\Omega) = m(\Omega) = \iint_\Omega f(\mathbf{x}, \Omega) \mathrm{d}\mathbf{x}$ with respect to velocity vector field $\mathbf{V}$ is given by

$$\mathrm{d}m(\Omega, \mathbf{V}) = \lim_{\tau \to 0} \left(\frac{m(\Omega(\tau)) - m(\Omega)}{\tau} \right) \tag{2.11}$$

Then, the terms on the RHS can be developed in the following manner

$$m(\Omega(\tau)) - m(\Omega) = \iint_{\Omega(\tau)} k(\mathbf{x}(\tau), \Omega(\tau)) \mathrm{d}\mathbf{x} - \iint_\Omega k(\mathbf{x}, \Omega) \mathrm{d}\mathbf{x} \tag{2.12}$$

By applying a change of variables in the first integral $\mathbf{x}(\tau) = \mathbf{x} + \tau\mathbf{V}(\mathbf{x})$, we have

$$m(\Omega(\tau)) - m(\Omega) = \iint_{\Omega(\tau)} k(\mathbf{x}+\tau\mathbf{V}(\mathbf{x}), \Omega+\tau\mathbf{V}(\mathbf{x}))|\det(J_\tau(\mathbf{x}))|\mathrm{d}\mathbf{x} - \iint_{\Omega} k(\mathbf{x},\Omega)\mathrm{d}\mathbf{x} \tag{2.13}$$

where $\mathbf{V}(\mathbf{x}) = (V_x, V_y) = (\dot{x}, \dot{y})$ is the velocity of an image point on the contour and $\det(J_\tau(\mathbf{x}))$ is the determinant of the Jacobian developed from transformation of the variables in the first integral.

$$\det J_\tau(\mathbf{x}) = \begin{vmatrix} 1+\tau\frac{\partial V_x}{\partial x} & \tau\frac{\partial V_x}{\partial y} \\ \tau\frac{\partial V_y}{\partial x} & 1+\tau\frac{\partial V_y}{\partial y} \end{vmatrix} \tag{2.14}$$

Let us introduce the term

$$\iota = \iint_{\Omega} k(\mathbf{x},\Omega) det J_\tau(\mathbf{x})\mathrm{d}\mathbf{x} \tag{2.15}$$

Adding the integral term ι to the first integral and subtracting it from the second integral in (2.13) and dividing both sides by τ , we get

$$\left.\begin{aligned} \frac{m(\Omega(\tau)) - m(\Omega)}{\tau} = \frac{1}{\tau}\iint_{\Omega(\tau)} \Big(k(\mathbf{x}+\tau\mathbf{V}(\mathbf{x}), \Omega+\tau\mathbf{V}(\mathbf{x})) - k(\mathbf{x},\Omega)\Big) \,|\det(J_\tau(\mathbf{x}))|\mathrm{d}\mathbf{x} \\ + \iint_{\Omega} k(\mathbf{x},\Omega)\left(\frac{\det(J_\tau(\mathbf{x})-1)}{\tau}\right)\mathrm{d}\mathbf{x} \end{aligned}\right\} \tag{2.16}$$

If the limits $\tau \to 0$ are applied to the first integral in Equation (2.16), we get what is known as the *material derivative* [JehanBesson 03]. That is,

$$\lim_{\tau\to 0}\Big(k(\mathbf{x}+\tau\mathbf{V}(\mathbf{x}), \Omega+\tau\mathbf{V}(\mathbf{x})) - k(\mathbf{x},\Omega)\Big) = \dot{k}(\mathbf{x},\Omega,\mathbf{V}) \tag{2.17}$$

Taking a first order Taylor expansion of (2.17), we obtain the relation which connects the material and shape derivatives [JehanBesson 03]:

$$\dot{k}(\mathbf{x},\Omega,\mathbf{V}) = k_s(\mathbf{x},\Omega,\mathbf{V}) + \nabla k(\mathbf{x},\Omega)\mathbf{V}(\mathbf{x}) \tag{2.18}$$

where k_s is the *shape derivative* and given by

$$k_s(\mathbf{x},\Omega,\mathbf{V}) = \lim_{\tau\to 0}\frac{k(\mathbf{x},\Omega+\tau\mathbf{V}(\mathbf{x})) - k(\mathbf{x},\Omega)}{\tau} \tag{2.19}$$

From (2.14), a relation between the determinant of the transformation and the divergence of the velocity vector $\mathbf{V}$ can be obtained such that

$$\lim_{\tau\to 0}\frac{\det J_\tau(\mathbf{x}) - 1}{\tau} = \mathrm{div}(\mathbf{V}(\mathbf{x})) \tag{2.20}$$

If we apply the limits to (2.16) on both sides, then substitute Equations (2.17) and (2.18) to the first integral and (2.20) to the second integral on the RHS, we get

$$\lim_{\tau\to 0}\frac{m(\Omega(\tau))-m(\Omega)}{\tau}=\begin{cases}\iint_\Omega k_s(\mathbf{x},\Omega,\mathbf{V})\mathrm{d}\mathbf{x}+\iint_\Omega \nabla k(\mathbf{x},\Omega)\mathbf{V}(\mathbf{x})\mathrm{d}\mathbf{x}\\ \qquad\qquad\qquad\qquad +\iint_\Omega k(\mathbf{x},\Omega)\mathrm{div}\mathbf{V}\mathrm{d}\mathbf{x}\end{cases} \tag{2.21}$$

As in [Tamtsia 13a], if the functional k on the moment equation does not depend on the region itself, the shape derivative k_s in Equation (2.21) can be neglected. Then, Equation (2.21) can be further developed as

$$\lim_{\tau\to 0}\frac{m(\Omega(\tau))-m(\Omega)}{\tau}=\iint_\Omega \nabla k(\mathbf{x},\Omega)\mathbf{V}(\mathbf{x})\mathrm{d}\mathbf{x}+\iint_\Omega k(\mathbf{x},\Omega)\mathrm{div}\mathbf{V}\mathrm{d}\mathbf{x} \tag{2.22}$$

From (2.11), we know that the LHS of (2.22) is the derivative of m_{pq} and the integrals in the RHS can be combined to obtain

$$\mathrm{d}m_{pq}=\iint_\Omega \mathrm{div}(k\mathbf{V})\mathrm{d}\mathbf{x} \tag{2.23}$$

Further, if we consider that the intensity levels in the image are not varying (binary image assumption) just like in [Chaumette 04], then, $k(\mathbf{x},\Omega)=k(\mathbf{x})$ since $f(x,y)=1,\forall\mathbf{x}$. Therefore, we obtain the same result as in the previous section.

$$\dot{m}_{pq}=\iint_\Omega \mathrm{div}(k(x,y)\dot{\mathbf{x}})\mathrm{d}\mathbf{x} \tag{2.24}$$

Obtaining the final Interaction Matrix Irrespective of the method used for obtaining them, (2.5) or (2.24) can be further expanded as

$$\dot{m}_{pq}=\iint_\Omega\Big(\frac{\partial k}{\partial x}\dot{x}+\frac{\partial k}{\partial y}\dot{y}+k(x,y)\Big(\frac{\partial \dot{x}}{\partial x}+\frac{\partial \dot{y}}{\partial y}\Big)\Big)\mathrm{d}x\mathrm{d}y \tag{2.25}$$

We know $k=x^py^q$ and their partial derivatives $\frac{\partial k}{\partial x}=px^{p-1}y^q$ and $\frac{\partial k}{\partial y}=qx^py^{q-1}$ are straight-forward to obtain. Similarly, the variations of an image point with respect to camera motions are known analytically (see Equation 1.50).

$$\dot{x}=-\frac{v_x}{Z}+x\frac{v_z}{Z}+xy\omega_x-(1+x^2)\omega_y+y\omega_z \tag{2.26a}$$

$$\dot{y}=-\frac{v_y}{Z}+y\frac{v_z}{Z}-xy\omega_y+(1+y^2)\omega_x-x\omega_z \tag{2.26b}$$

Now, let us consider a planar object or an object with a planar limb surface. Then, we can express the depth of an image point as a function of its coordinates [Chaumette 04] :

$$\frac{1}{Z}=Ax+By+C \tag{2.27}$$

where A, B and C are the parameters related to the plane to which the imaged scene or planar limb surface belongs. This excludes the degenerate case of the camera optical center belonging to this plane. Substituting Equation(2.27) in Equations (2.26a) and (2.26b), the following partial derivatives can be obtained

$$\frac{\partial \dot{x}}{\partial x} = -Av_x + (2Ax + By + C)v_z + y\omega_x - 2x\omega_y \tag{2.28a}$$

$$\frac{\partial \dot{y}}{\partial y} = -Bv_y + (Ax + 2By + C)v_z + 2y\omega_x - x\omega_y \tag{2.28b}$$

After substitution of all the partial derivatives in closed-form, (2.24) can be assembled to the general form $\dot{s} = \mathbf{L}_s \mathbf{v}_c$ and hence $\mathbf{L}_{m_{pq}}$ be obtained from it. Then

$$\mathbf{L}_{m_{pq}} = \begin{bmatrix} \mathbf{L}^{\mathbf{v}}_{\mathtt{m_{pq}}} & \mathbf{L}^{\boldsymbol{\omega}}_{\mathtt{m_{pq}}} \end{bmatrix} = \begin{bmatrix} L^{v_x}_{\mathtt{m_{pq}}} & L^{v_y}_{\mathtt{m_{pq}}} & L^{v_z}_{\mathtt{m_{pq}}} & L^{\omega_x}_{\mathtt{m_{pq}}} & L^{\omega_y}_{\mathtt{m_{pq}}} & L^{\omega_z}_{\mathtt{m_{pq}}} \end{bmatrix}$$

where [Chaumette 04]

$$\left.\begin{aligned}
L^{v_x}_{\mathtt{m_{pq}}} &= -A(p+1)m_{pq} - Bpm_{p-1,q+1} - Cpm_{p-1,q} \\
L^{v_y}_{\mathtt{m_{pq}}} &= -Aqm_{p+1,q-1} - B(q+1)m_{p,q} - Cqm_{p,q-1} \\
L^{v_z}_{\mathtt{m_{pq}}} &= A\,(p+q+3)\,m_{p+1,q} + B(p+q+3)\,m_{p,q+1} + C(p+q+2)\,m_{pq} \\
L^{\omega_x}_{\mathtt{m_{pq}}} &= q\,m_{p,q-1} + (p+q+3)\,m_{p,q+1} \\
L^{\omega_y}_{\mathtt{m_{pq}}} &= -p\,m_{p-1,q} - (p+q+3)\,m_{p+1,q} \\
L^{\omega_z}_{\mathtt{m_{pq}}} &= p\,m_{p-1,q+1} - q\,m_{p+1,q-1}
\end{aligned}\right\} \tag{2.29}$$

Thus, an analytical form of the interaction matrix can be developed for image moments. From (2.29), for computing the interaction matrix of a moment of order $p+q$, moments of order upto $p+q+1$ are required. The portion of the interaction matrix related to the rotational velocities $\mathbf{L}^{\boldsymbol{\omega}}_{\mathtt{m_{pq}}}$ is free from the plane parameters A, B and C. This is expected since the rotational motions do not depend on the depth of the scene.

2.2.2.1 Interaction matrix for moments from a discrete set of points

Moments can also be computed from a set of discrete points available in the image. In the case of a set of 2D points $\{\mathbf{x}_i\} = \{x_i, y_i\}$, we have

$$m_{pq} = \sum_{i=1}^{N} x_i^p y_i^q \tag{2.30}$$

Differentiating Equation (2.30), we have

$$\dot{m}_{pq} = \sum_{i=1}^{N} \left(p x_i^{p-1} y_i^q \dot{x}_i + q x_i^p y_i^{q-1} \dot{y}_i \right) \tag{2.31}$$

The terms $\frac{\partial \dot{x}}{\partial x}$ and $\frac{\partial \dot{y}}{\partial y}$ present in Equation(2.25) do not appear anymore. Further, by substitution of Equations (2.26a), (2.26b) and (2.27) in Equation(2.31), we can obtain the

interaction matrix of the moments for the discrete case.

$$\left.\begin{aligned}
L_{\mathtt{m}_{pq}}^{v_x} &= -p(Apm_{pq} + Bm_{p-1,q+1} + Cm_{p-1,q}) - \delta Am_{pq} \\
L_{\mathtt{m}_{pq}}^{v_y} &= -q(Am_{p+1,q-1} + Bm_{p,q} + Cm_{p,q-1}) - \delta Bm_{pq} \\
L_{\mathtt{m}_{pq}}^{v_z} &= (p+q+3\delta)\,(Am_{p+1,q} + Bm_{p,q+1} + Cm_{pq}) - \delta Cm_{pq} \\
L_{\mathtt{m}_{pq}}^{\omega_x} &= qm_{p,q-1} + (p+q+3\delta)m_{p,q+1} \\
L_{\mathtt{m}_{pq}}^{\omega_y} &= -p\,m_{p-1,q} - (p+q+3\delta)\,m_{p+1,q} \\
L_{\mathtt{m}_{pq}}^{\omega_z} &= p\,m_{p-1,q+1} - q\,m_{p+1,q-1}
\end{aligned}\right\} \tag{2.32}$$

with $\delta = 1$. Thus, the interaction matrix for moments computed from a discrete set of points (discrete case) is different from the one for moments computed from a dense object defined by closed contours (dense case). It is to be noted that Equation (2.32) is general and applicable to moments computed both in the discrete and dense cases (with the appropriate δ). Indeed, when $\delta = 0$, the above set of equations are equal to Equation (2.29) obtained for the dense case. But the behaviour of the visual features developed from them will be similar [Tahri 05a], to be discussed in due course.

2.3 Moments-based Visual Servoing

Moments are not used per se in visual servoing since they present no interesting properties that are favourable for the control. In the following, we will discuss how invariant properties of specific functions of the moments can be exploited to design decoupled control laws. We will first see how lower (zeroth and first) order moments are made use of to control translational motions. In Section 1.1.3, we introduced formal definitions of a visual feature and how it is very interesting to have visual features that are related to a single DOF. Image moments present interesting invariance properties which can be utilized to obtain visual features that are related to motions along specific degrees of freedom.

2.3.1 Control of Translational Motions

In this section, we discuss how lower order moments are sufficient to derive visual features that can be used to control the spatial 3D translations. The interaction matrices for these features were introduced in [Chaumette 04].

2.3.1.1 Zeroth Order Moments

The zeroth order moment is the sum of the values of the distribution over the domain of integration.

$$m_{00} = \iint_{\Omega} f(x,y)\mathrm{d}x\mathrm{d}y \tag{2.33}$$

Its interaction matrix can be obtained by substituting $p = q = 0$ in Equation (2.32).

$$\mathbf{L}_{m_{00}} = \begin{bmatrix} -\delta Am_{00} & -\delta Bm_{00} & \delta m_{00}\Big(\dfrac{3}{Z_g} - C\Big) & 3\delta m_{00}y_g & -3\delta m_{00}x_g & 0 \end{bmatrix} \tag{2.34}$$

where x_g and y_g are the coordinates of the center of gravity $\mathbf{x}_g$ of the target object projected in the image (see Equation(2.37)).

If we consider the discrete case and set $\delta = 0$ in Equation (2.34), the interaction matrix becomes $\mathbf{0}$. This is not surprising since when moments are computed from a discrete point set, m_{00} is in fact the number of points N considered in Equation (2.30). The number of points in the image does not change with camera motion [1].

On the other hand, the feature is valid for the dense case $\delta = 1$. Further, when a binary object $f(x, y) = 1 \,\forall\, \Omega$ is considered, this moment is simply the area of the projection in the image of the target object. From Equation (2.33), we see that the zeroth order moment is mainly related to the translational motion along the optic axis. Most importantly, m_{00} is invariant to rotations around the optic axis at all configurations between the sensor and target planes. Further, when the camera and the object planes are parallel, we have $A = B = 0$ and hence this feature is invariant to planar translational motions in this case and the interaction matrix has the below form

$$\mathbf{L}^{||}_{m_{00}} = \begin{bmatrix} 0 & 0 & m_{00}\left(\frac{3}{Z_g} - C\right) & 3m_{00}y_g & -3m_{00}x_g & 0 \end{bmatrix} \tag{2.35}$$

It is to be noted that only the first 3 entries related to translational motions are affected and the interaction matrix related to rotational motions remain the same (since the plane parameters A, B or C do not appear in the rotational part). This holds true for also the features discussed next and we will refrain from restating this in the following sections. Further, when the target object is centred in the image such that the center of gravity is null $x_g = y_g = 0$, m_{00} becomes invariant to the rotational motions along x and y and we have an even more decoupled structure.

$$\mathbf{L}^{||}_{m_{00}|(\mathbf{x}_g=0)} = \begin{bmatrix} 0 & 0 & \delta m_{00}\left(\frac{3}{Z_g} - C\right) & 0 & 0 & 0 \end{bmatrix} \tag{2.36}$$

In this configuration, we can observe that the variation in m_{00} is directly linked to translation along the optic axis and does not change with respect to other motions. Hence, this feature is apt to control translational motion along the optic axis.

2.3.1.2 Moments of First Order

It has to be noted that the center of gravity in Equation (2.34) is not visually tracked. It is directly determined as simple ratios of the first order moments and the zeroth order moment.

$$\mathbf{x}_g = \quad (x_g, y_g) \quad = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{2.37}$$

The center of gravity is in fact used separately as visual features for controlling the planar translations. Their interaction matrix can be determined easily from Equation (2.32) by direct substitution of the moment order indexes p and q.

$$\mathbf{L}_{\mathbf{x}_g} = \begin{bmatrix} \mathbf{L}_{x_g} \\ \mathbf{L}_{y_g} \end{bmatrix} = \begin{bmatrix} -\frac{1}{Z_g} & 0 & L^{\nu_z}_{x_g} & L^{\omega_x}_{x_g} & L^{\omega_y}_{x_g} & y_g \\ 0 & -\frac{1}{Z_g} & L^{\nu_z}_{y_g} & L^{\omega_x}_{y_g} & L^{\omega_y}_{y_g} & -x_g \end{bmatrix} \tag{2.38}$$

[1] unless some of them get outside the field of view which is altogether a different problem

with

$$\begin{cases} L_{x_g}^{v_z} = \dfrac{x_g}{Z_g} + \epsilon(An_{20} + Bn_{11}) \\ L_{y_g}^{v_z} = \dfrac{y_g}{Z_g} + \epsilon(An_{11} + Bn_{02}) \\ L_{x_g}^{\omega_x} = -L_{y_g}^{\omega_y} = x_g y_g + \epsilon n_{11} \\ L_{x_g}^{\omega_y} = -(1 + x_g^2 + \epsilon n_{20}) \\ L_{y_g}^{\omega_x} = 1 + y_g^2 + \epsilon n_{02} \end{cases}$$

where $\epsilon = 4$ for the dense case and $\epsilon = 1$ for the discrete case, $n_{ij} = \frac{\mu_{ij}}{m_{00}}$ are the normalized moments of second order with

$$\begin{cases} \mu_{20} = m_{20} - m_{00} x_g^2 \\ \mu_{02} = m_{02} - m_{00} y_g^2 \\ \mu_{11} = m_{11} - m_{00} x_g y_g \end{cases} \tag{2.39}$$

Here, μ_{pq} are centred moments of order 2. Centred moments of a general order $p+q$ will be defined shortly in Section 2.3.2.1. It has to be noted that the interaction matrix in (2.38) is a generalization of the interaction matrix of an image point given in Equations 2.26a - 2.26b. The latter can be derived by taking $A = B = 0$ and $n_{20} = n_{02} = n_{11} = 0$. It is known that x_g does not change when the camera translates along the y axis and y_g does not change when the camera translates along the x axis. This invariance results in decoupling between the translations along x and y, i.e., $L_{x_g}^{\nu_y} = L_{y_g}^{\nu_x} = 0$. More generally, when $A = B = 0$ (where the camera and target planes are parallel, denoted by $||$), we have

$$\mathbf{L}_{\mathbf{x}_g}^{||} = \begin{bmatrix} -\frac{1}{Z_g} & 0 & \frac{x_g}{Z_g} & L_{x_g}^{\omega_x} & L_{x_g}^{\omega_y} & y_g \\ 0 & -\frac{1}{Z_g} & \frac{y_g}{Z_g} & L_{y_g}^{\omega_x} & L_{y_g}^{\omega_y} & -x_g \end{bmatrix} \tag{2.40}$$

Further, if the target object is perfectly centred in the image $\mathbf{x}_g = 0$, this feature becomes invariant to rotations around the optic axis as well. Then, we have the following decoupled interaction matrix for control of planar translations.

$$\mathbf{L}_{\mathbf{x}_g|\mathbf{x}_g=0}^{||} = \begin{bmatrix} -\frac{1}{Z_g} & 0 & 0 & \epsilon n_{11}^{'} & -(1+\epsilon n_{20}^{'}) & 0 \\ 0 & -\frac{1}{Z_g} & 0 & (1+\epsilon n_{02}^{'}) & -\epsilon n_{11}^{'} & 0 \end{bmatrix} \tag{2.41}$$

where $n_{ij}^{'} = \dfrac{m_{ij}}{m_{00}}$. The features $\mathbf{x}_g$ and m_{00} are used together for controlling the spatial (3D) translations. Though the centre of gravity feature can be computed from moments only upto the first order, its interaction matrix requires moments of upto second order to be computed. The same can also be inferred from Equation (2.29).

2.3.1.3 An Improved Feature Set for Control of Translational Motions

Let us consider the feature choice $(\mathbf{x}_g, m_{00})$ and analyse their interaction matrices (given by Equations (2.40) and (2.35)) with respect to only the translation motions when $A = B = 0$.

$$\begin{bmatrix} \mathbf{L}_{x_g}^{||\mathbf{v}} \\ \mathbf{L}_{y_g}^{||\mathbf{v}} \\ \mathbf{L}_{m_{00}}^{||\mathbf{v}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{Z_g} & 0 & \frac{x_g}{Z_g} \\ 0 & -\frac{1}{Z_g} & \frac{y_g}{Z_g} \\ 0 & 0 & m_{00}\left(\frac{3}{Z_g} - C\right) \end{bmatrix} \tag{2.42}$$

As noted before, for the centre of gravity feature, there is a decoupling of translational motion along x (resp.y) from the translational motion along y (resp.x). For feature in the third row, it is related to the translations along the optic z axis and decoupled from x and y motions. Also, we assume a planar target in which case all the object points including the centre of gravity are at the same uniform depth $Z_g = Z$. Further, since the target is situated parallel to the sensor plane, $A = B = 0$ and from the plane equation in (2.27), we have $\frac{1}{Z} = C$. So, the interaction matrix above can also be written more compactly as

$$\begin{bmatrix} \mathbf{L}_{x_g}^{\|\mathbf{v}} \\ \mathbf{L}_{y_g}^{\|\mathbf{v}} \\ \mathbf{L}_{m_{00}}^{\|\mathbf{v}} \end{bmatrix} = \begin{bmatrix} -C & 0 & Cx_g \\ 0 & -C & Cy_g \\ 0 & 0 & \frac{2m_{00}}{Z} \end{bmatrix} \tag{2.43}$$

It was shown in [Tahri 05a] that an adequate normalization applied to $(\mathbf{x}_g, m_{00})$ can result in an interaction matrix more closer to the ideal structure (discussed in Section 1.1.3) in Equation (1.43) than the interaction matrix developed for $(\mathbf{x}_g, m_{00})$.

Normalized Area Moment For notational simplicity, let us have $m_{00} = a$. Let us see how to get a normalized version of a such that its interaction matrix component with respect to translation motions along optic axis is -1. In other words, we search for a_n such that $L_{a_n}^{v_z} = -1$ when the sensor and target planes are parallel. If $a_n = a^l$, the derivative of this normalized feature is given by

$$\dot{a}_n = la^{l-1}\dot{a} \tag{2.44}$$

Since $\dot{a} = \mathbf{L}_a \mathbf{v}_c$, we can write

$$\dot{a}_n = la^{l-1}\mathbf{L}_a \mathbf{v}_c \tag{2.45}$$

That being said, we substitute for $\mathbf{L}_{m_{00}}$ from Equation (2.43) only the component related to linear velocity along the optic axis. Further substituting for C from Equation (2.27), we get

$$\dot{a}_n = la^l \frac{2}{Z} v_z \tag{2.46}$$

In perspective projection, we know that the size (area) of the object in the image is inversely proportional to the distance of the camera to the object. From the equations of perspective projection we can write

$$a = \frac{A}{Z^2} \implies \sqrt{\frac{a}{A}} = \frac{1}{Z} \tag{2.47}$$

where A is the area of the planar object. Substituting for $\frac{1}{Z}$ from Equation (2.47) into Equation (2.46),we have

$$\dot{a}_n = \frac{2la^{l+\frac{1}{2}}}{\sqrt{A}} v_z \tag{2.48}$$

A choice of $l = -\frac{1}{2}$ would give $\dot{a}_n = -\frac{v_z}{\sqrt{A}} = -\frac{v_z}{Z\sqrt{a}}$. [Tahri 05a] noted that $\sqrt{A} = Z\sqrt{a} = Z^*\sqrt{a}*$ and proposed the following normalized version of a.

$$a_n = Z^* \sqrt{\frac{a*}{a}} \tag{2.49}$$

It is easy to observe that at the desired pose, $a = a^*$ and $a_n = Z^*$. Its interaction matrix is then given by

$$\mathbf{L}_{a_n}^{||} = \begin{bmatrix} 0 & 0 & -1 & -a_n\epsilon_{31} & a_n\epsilon_{32} & 0 \end{bmatrix} \tag{2.50}$$

where $\epsilon_{31} = \frac{3}{2}y_g$ and $\epsilon_{32} = \frac{3}{2}x_g$. This feature now depends linearly on the depth.

Normalized Centre of Gravity Once a_n and $\mathbf{L}_{a_n}$ are available, the normalized centre of gravity $\mathbf{x}_n(x_n, y_n)$ can be obtained by solving $x_n = x_g a_n^l$ such that $L_{x_n}^{v_x} = -1$ and $y_n = y_g a_n^l$ such that $L_{y_n}^{v_y} = -1$. If the development steps as for a_n are followed, we will observe that we get $l = 1$ and we obtain the visual feature set proposed by [Tahri 05a].

$$x_n = x_g a_n \tag{2.51a}$$

$$y_n = y_g a_n \tag{2.51b}$$

The interaction matrix of the feature set $\mathbf{x}_n$ has the following structure

$$\begin{bmatrix} \mathbf{L}_{x_n}^{||} \\ \mathbf{L}_{y_n}^{||} \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & a_n\epsilon_{11} & -a_n(1+\epsilon_{12}) & y_n \\ 0 & -1 & 0 & a_n(1+\epsilon_{21}) & -a_n\epsilon_{22} & -x_n \end{bmatrix} \tag{2.52}$$

with $\epsilon_{11} = \epsilon_{22} = 4n_{11} - x_g y_g/2$, $\epsilon_{12} = 4n_{20} - x_g^2/2$, $\epsilon_{21} = 4n_{02} - y_g^2/2$. Now, if we use the set $\mathbf{x}_n$ and a_n for control, we note that the interaction matrix related to translations is an identity $-\mathbf{I}_3$. A control law using this gives rise to a nice behaviour of the translational trajectories.

2.3.2 Control of Rotational Motions around the optic axis

Existing visual features for the control of rotational degrees of freedom are based on the centred moments. A visual feature built from centred moments of second order is used for control of rotations around the optic axis (Z) while for the control of rotations around the x and y axes, visual features based on moment invariants are used [Chaumette 04] [Tahri 05a].

2.3.2.1 Centred Moments

The centre of gravity of the object projection in the image is known from Equation (2.37). With this knowledge, a new set of moments called centred moments can be obtained from it. A centred moment of order $p + q$ can be defined by the following equation.

$$\mu_{pq} = \iint_{\Omega} (x - x_g)^p (y - y_g)^q \, \mathrm{d}x\mathrm{d}y \tag{2.53}$$

The spatial coordinates are altered with respect to their distance from the centre of gravity. Translational invariance is therefore achieved by a coordinate transformation with respect to the centre of gravity. However, we will soon see that this invariance holds only when the sensor and target planes are parallel to each other. If the terms $(x - x_g)^p$ and $(y - y_g)^p$ in Equation (2.53) are expanded using the binomial theorem, we can directly obtain the relation between the raw moments and centred moments.

$$\mu_{pq} = \sum_{k=0}^{p} \sum_{l=0}^{q} (-1)^{p+q-k-l} \binom{p}{k} \binom{q}{l} (x_g)^{p-k} (y_g)^{q-l} m_{kl} \tag{2.54}$$

As proposed in [Chaumette 04], there are two ways to obtain the interaction matrix of the centred moments.

1. We can differentiate Equation (2.53) directly as follows.

$$\left.\begin{aligned}\dot{\mu}_{pq} = \iint_{\Omega}\Big[&p(x-x_g)^{p-1}(y-y_g)^{q}(\dot{x}-\dot{x}_g) + q(x-x_g)^{p}(y-y_g)^{q-1}(\dot{y}-\dot{y}_g)\\ &+ (x-x_g)^{p}(y-y_g)^{q}\Big(\frac{\partial \dot{x}}{\partial x}+\frac{\partial \dot{y}}{\partial y}\Big)\Big]\mathrm{d}x\mathrm{d}y\end{aligned}\right\} \tag{2.55}$$

   Then, by substitution from Equations (2.26a), (2.26b), (2.28a) and (2.28b) similar to the developments at the end of Section 2.2, we can obtain the interaction matrix in closed form.

2. Alternatively, the interaction matrix of the centred moments can be obtained by differentiation of Equation (2.54).

$$\left.\begin{aligned}\dot{\mu}_{pq} = \sum_{k=0}^{p}\sum_{l=0}^{q}(-1)^{p+q-k-l}\binom{p}{k}\binom{q}{l}\Big[m_{kl}\Big((p-k)(x_g)^{p-k-1}y_g\dot{x}_g\\ + (q-l)x_g(y_g)^{q-l-1}\dot{y}_g\Big) + (x_g)^{p-k}(y_g)^{q-l}\dot{m}_{kl}\Big]\end{aligned}\right\} \tag{2.56}$$

   Subsequently, since $\dot{\mu}_{pq} = \mathbf{L}_{\mu_{pq}}\mathbf{v}_c$, we can write

$$\left.\begin{aligned}\mathbf{L}_{\mu_{pq}} = \sum_{k=0}^{p}\sum_{l=0}^{q}(-1)^{p+q-k-l}\binom{p}{k}\binom{q}{l}\Big[m_{kl}\Big((p-k)(x_g)^{p-k-1}y_g\mathbf{L}_{x_g}\\ + (q-l)x_g(y_g)^{q-l-1}\mathbf{L}_{y_g}\Big) + (x_g)^{p-k}(y_g)^{q-l}\mathbf{L}_{m_{kl}}\Big]\end{aligned}\right\} \tag{2.57}$$

With either choice, substitution of the interaction matrix $\mathbf{L}_{\mathbf{x}_g}$ and the interaction matrix of raw moments (2.32), we can obtain the interaction matrix of the centred moments in analytical form. It is given by [Tahri 05a]:

$$\mathbf{L}_{\mu_{pq}} = \begin{bmatrix} L^{\nu_x}_{\mu_{\mathrm{pq}}} & L^{\nu_y}_{\mu_{\mathrm{pq}}} & L^{\nu_z}_{\mu_{\mathrm{pq}}} & L^{\omega_x}_{\mu_{\mathrm{pq}}} & L^{\omega_y}_{\mu_{\mathrm{pq}}} & L^{\omega_z}_{\mu_{\mathrm{pq}}}\end{bmatrix} \tag{2.58}$$

where

$$\left\{\begin{aligned}
L^{\nu_x}_{\mu_{\mathrm{pq}}} &= -A(p+\delta)\mu_{pq} - B\,p\,\mu_{p-1,q+1}\\
L^{\nu_y}_{\mu_{\mathrm{pq}}} &= -A\,q\,\mu_{p+1,q-1} - B(q+\delta)\mu_{p,q}\\
L^{v_z}_{\mu_{\mathrm{pq}}} &= -A\,L^{\omega_y}_{\mu_{\mathrm{pq}}} + B\,L^{\omega_x}_{\mu_{\mathrm{pq}}} + C(p+q+2\delta)\,\mu_{pq}\\
L^{\omega_x}_{\mu_{\mathrm{pq}}} &= (p+q+3\delta)\mu_{p,q+1} + p\,x_g\,\mu_{p-1,q+1} + (p+2q+3\delta)\,y_g\,\mu_{p,q}\\
&\quad - \epsilon\,p\,n_{11}\,\mu_{p-1,q} - \epsilon\,q\,n_{02}\,\mu_{p,q-1}\\
L^{\omega_y}_{\mu_{\mathrm{pq}}} &= -(p+q+3\delta)\mu_{p+1,q} - (2p+q+3)\,x_g\,\mu_{p,q} - q\,y_g\,\mu_{p+1,q-1}\\
&\quad + 4\,p\,n_{20}\,\mu_{p-1,q} + 4\,q\,n_{11}\,\mu_{p,q-1}\\
L^{\omega_z}_{\mu_{\mathrm{pq}}} &= p\,\mu_{p-1,q+1} - q\,\mu_{p+1,q-1}
\end{aligned}\right.$$

This interaction matrix is general and valid for both the dense and discrete cases. Indeed in the dense case $\delta = 1$ and it reduces to the case in [Chaumette 04]. From Equation (2.58), we observe that if $A = B = 0$, we have $L^{\nu_x}_{\mu_{pq}} = L^{\nu_y}_{\mu_{\mathrm{pq}}} = 0$. So, the centred moments are invariant to planar translations only if the sensor and target planes are parallel to each other. Just like in Equation (2.29), the interaction matrix for control of rotational motions is devoid of the plane parameters A, B or C.

2.3.2.2 Orientation Feature

The orientation feature is the angle between the horizontal image axis and the principal axis of inertia of the projection of the target object in the image (see Figure 2.2). This feature serves to regulate the error in planar orientation in the image. This orientation can be derived from the relation between the second order moments and the principal moments of inertia of the image projection [Mukundan 98].[2] It is given by

$$\alpha = \frac{1}{2}\arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{2.59}$$

By differentiating Equation (2.59), we have

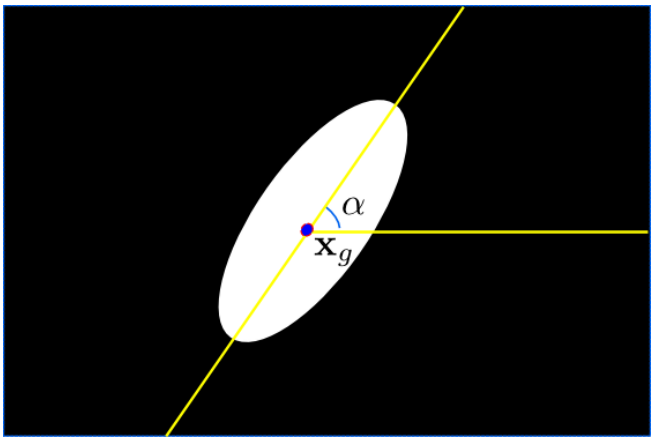


Figure 2.2 – The orientation of the target projection in the image can be represented as an ellipse

$$\mathbf{L}_\alpha = (\mu_{20} - \mu_{02})\mathbf{L}_{\mu_{11}} - \mu_{11}\frac{(\mathbf{L}_{\mu_{20}} - \mathbf{L}_{\mu_{02}})}{\Delta} \tag{2.60}$$

where $\Delta = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2$. The interaction matrices $\mathbf{L}_{\mu_{11}}$, $\mathbf{L}_{\mu_{20}}$ and $\mathbf{L}_{\mu_{02}}$ can be directly obtained from (2.58). For the dense case, $\mathbf{L}_\alpha$ was developed analytically in [Chaumette 04] and its invariance properties can be easily analysed.

$$\mathbf{L}_\alpha = \begin{bmatrix} \mathrm{L}^{\nu_x}_\alpha & \mathrm{L}^{\nu_y}_\alpha & \mathrm{L}^{\nu_z}_\alpha & \mathrm{L}^{\omega_x}_\alpha & \mathrm{L}^{\omega_y}_\alpha & -1 \end{bmatrix} \tag{2.61}$$

[2] A supplementary reference is [Hu 62] which obtained α using the theory of algebraic invariants.

where

$$\begin{cases} \mathrm{L}_\alpha^{\nu_x} = aA + bB & a = \mu_{11}\dfrac{(\mu_{20}+\mu_{02})}{\Delta} \\ \mathrm{L}_\alpha^{\nu_y} = -cA - aB & b = \dfrac{2\mu_{11}^2 + \mu_{02}(\mu_{02}-\mu_{20})}{\Delta} \\ \mathrm{L}_\alpha^{\nu_z} = -A\mathrm{L}_\alpha^{\omega_y} + B\mathrm{L}_\alpha^{\omega_x} & c = \dfrac{2\mu_{11}^2 + \mu_{20}(\mu_{20}-\mu_{02})}{\Delta} \\ \mathrm{L}_\alpha^{\omega_x} = -bx_g + ay_g + d & d = \dfrac{5[\mu_{12}(\mu_{20}-\mu_{02}) + \mu_{11}(\mu_{03}-\mu_{21})]}{\Delta} \\ \mathrm{L}_\alpha^{\omega_y} = ax_g - cy_g + e & e = \dfrac{5[\mu_{21}(\mu_{02}-\mu_{20}) + \mu_{11}(\mu_{30}-\mu_{12})]}{\Delta} \end{cases}$$

From (2.61), we observe that $\mathrm{L}_\alpha^{\omega_z} = -1$ for all configurations between the sensor and target planes. That is why this feature is suitable for controlling this rotational degree of freedom. Furthermore, when the sensor and target planes are parallel to each other ($A = B = 0$), the orientation feature is invariant to 3D translations. We have the following decoupled structure:

$$\mathbf{L}_\alpha^{||} = \begin{bmatrix} 0 & 0 & 0 & \mathrm{L}_\alpha^{\omega_x} & \mathrm{L}_\alpha^{\omega_y} & -1 \end{bmatrix} \tag{2.62}$$

where $\mathrm{L}_\alpha^{\omega_x}$ and $\mathrm{L}_\alpha^{\omega_y}$ are given in closed form in [Tahri 05a]:

$$\begin{cases} \mathrm{L}_\alpha^{\omega_x} = (\beta[\mu_{12}(\mu_{20}-\mu_{02}) + \mu_{11}(\mu_{03}-\mu_{21})] + \gamma x_g[\mu_{12}(\mu_{20}-\mu_{02}) - 2\mu_{11}^2] \\ \quad + \gamma y_g \mu_{11}[\mu_{20}+\mu_{02}])/\Delta \\ \mathrm{L}_\alpha^{\omega_y} = (\beta[\mu_{21}(\mu_{20}-\mu_{02}) + \mu_{11}(\mu_{30}-\mu_{12})] + \gamma y_g[\mu_{21}(\mu_{02}-\mu_{20}) - 2\mu_{11}^2] \\ \quad + \gamma x_g \mu_{11}[\mu_{20}+\mu_{02}])/\Delta \end{cases}$$

with $\beta = 5$ and $\gamma = 1$ for dense objects and $\beta = 4$ and $\gamma = 2$ for the discrete case. Interestingly, the above direct relation of this feature to optic axis rotation and decoupling properties are true for both the dense and discrete cases. Control of the other two rotational degrees of freedom is non-trivial however. In the following, we will describe moment invariants and various methods for obtaining them.

2.3.3 Theory behind Moment Invariants

Moment invariants are functions of the moments that are invariant to specific transformations of the spatial coordinates in the image. Consider $\mathcal{F}$ as a general mapping from $\mathbf{x}$ to $\mathbf{x}'$. An invariant to this mapping, denoted $\Upsilon_\mathcal{F}$ satisfies

$$\Upsilon_\mathcal{F}(\mathbf{x}) = \Upsilon(\mathcal{F}(\mathbf{x})) \ \forall \mathcal{F} \tag{2.63}$$

The general transformation that describes the formation of a monocular image is the perspective projection transform. It can be written as

$$\mathcal{P}: \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \frac{\begin{bmatrix} a_1 & a_2 & t_x \\ b_1 & b_2 & t_y \\ c_1 & c_2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}{\begin{bmatrix} c_1 & c_2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}} \tag{2.64}$$

The perspective transform is a non-linear transformation of the spatial coordinates. When the target object is smaller in size compared to the distance of the camera to the object, c_1 and c_2 tend to 0 and the perspective transformation reduces to an affine transform. This can be represented as

$$\mathcal{A} : \begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{A} \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{t} \quad \text{where } \mathbf{A} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \text{ and } \mathbf{t} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{2.65}$$

The affine transform can be used to describe all geometrical transformations of the image coordinates on a plane and hence also called general linear transform. If we restrict the transformation $\mathbf{A}$ such that it represents planar rotation of a rigid object, affine transformations are reduced to a TRS transformation. By enforcing the constraints $a_1 = b_2$ and $a_2 = -b_1$, $\mathbf{A}$ becomes a rotation matrix. In addition, if we scale the spatial coordinates by a factor of s, these transformations are together called the similarity transformation.

$$\mathcal{S} : \begin{bmatrix} x' \\ y' \end{bmatrix} = s \times \mathbf{R}(\alpha) \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{t} \quad \text{where } \mathbf{R}(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \tag{2.66}$$

is a rotation matrix that can be parametrized by an angle that represents the amount of rotation in the image plane. In visual servoing methods based on image moments, visual features that are invariant to the above translations $\mathbf{t}$, rotations $\mathbf{R}$ and scale s are used in controlling the x and y rotational degrees of freedom non-orthogonal to the optic axis. We shall next see how invariances to the above transformations have been achieved.

2.3.3.1 Invariance to rotation

Invariants to rotationally transformed images were first introduced in the pattern recognition community. A set of 7 moment polynomials invariant with respect to in-plane rotations around the origin were proposed in [Hu 62].

$$\left.\begin{aligned}
h_1 &= \mu_{20} + \mu_{02} \\
h_2 &= (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \\
h_3 &= (\mu_{30} - 3\mu_{12})^2 + (\mu_{03} - 3\mu_{21})^2 \\
h_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{03} + \mu_{21})^2 \\
h_5 &= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) \left[(\mu_{30} + \mu_{12})^2 - 3(\mu_{03} + \mu_{21})^2\right] \\
&\quad + (3\mu_{21} - \mu_{03})(\mu_{03} + \mu_{21}).\left[3(\mu_{30} + \mu_{12})^2 - (\mu_{03} + \mu_{21})^2\right] \\
h_6 &= (\mu_{20} - \mu_{02}) \left[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right] \\
&\quad + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \\
h_7 &= (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})((\mu_{30} + \mu_{12})^2 - 3(\mu_{03} + \mu_{21})^2) \\
&\quad - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})(3(\mu_{30} + \mu_{12})^2 - (\mu_{03} + \mu_{21})^2)
\end{aligned}\right\} \tag{2.67}$$

In this classic work, the invariants were derived using the fundamental theorem of moment invariants (FTMI). A generalized version of this theorem is presented below. The preliminaries required for understanding this theorem is the excellent text on the theory of algebraic invariants [Hilbert 93] [3].

[3] published posthumously 50 years after the death of the legendary mathematician David Hilbert from his lectures at the University of Gottingen

Theory of Algebraic Invariants A binary form is a homogeneous polynomial in two variables $F(x_1, x_2) = a_{20}x_1^2 + a_{11}x_1x_2 + a_{02}x_2^2$. The weight of the binary form is the degree of the terms in the polynomial (For F, it is 2). From its coefficients $\mathbf{a} = [a_{20}, a_{11}, a_{02}]$, we observe the order of F is also 2. Similarly, $\mathbf{m}_p$ is a vector of moments of order p. For instance, for the second order, we have $\mathbf{m}_2 = [m_{20}, m_{11}, m_{02}]$. The Revised Fundamental Theorem of Moment Invariants (RFTMI) [Reiss 91] can be stated as under.

Theorem 2.3.1

If a binary form of order p has an algebraic invariant of weight w and order k,

$$I(\mathbf{a}') = \Delta^w I(\mathbf{a}) \tag{2.68}$$

then moments of order p have the same invariant but also an included additional factor $|J|^k$.

$$I(\boldsymbol{m}_p') = \Delta^w |J|^k I(\boldsymbol{m}_p) \tag{2.69}$$

where $J = \Delta$ is the Jacobian of the linear transformation to which $I(\mathbf{a})$ is invariant.

If $k = 1$ or $|J| = 1$ in Equation (2.69), the RTFMI reduces to the fundamental theorem of moment invariants (FTMI) stated by [Hu 62]. That is, the FTMI and RFTMI are one and the same with respect to scaling $(k = 1)$ and rotations $(|J| = 1)$. Since any polynomial can be rendered homogeneous by simple addition of extra variables, the above theorem is applicable also to non-homogeneous polynomials. A formal proof of the above theorem using Fourier Transform is given in [Flusser 09] Appendix 3.A.

On careful observation of this theorem, we can infer that to find moment invariants, knowledge of the algebraic invariants of binary forms is needed. Unfortunately, this theorem does not deal with how to find these algebraic invariants in the first place. Later, based on [VanGool 95], a new procedure for generating moment invariants based on Lie Group theory and valid for n dimensional space was given in [Tahri 04]. A detailed discussion of this method follows in Frame 1. Below is a listing of the moment invariants[4] presented

[4]Some typographical mistakes in I_5, I_6, I_8 and I_{11} in the original version have been corrected

in [Tahri 05a].

$$
\left.
\begin{aligned}
I_1 &= -\mu_{20}\mu_{02} + \mu_{11}^2 \\
I_2 &= (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \\
I_3 &= (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \\
I_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \\
I_5 &= -\mu_{30}^2\mu_{03}^2 + 6\mu_{30}\mu_{21}\mu_{03}\mu_{12} - 4\mu_{30}\mu_{12}^3 - 4\mu_{21}^3\mu_{03} + 3\mu_{21}^2\mu_{12}^2 \\
I_6 &= 3\mu_{30}^2\mu_{12}^2 + 2\mu_{30}^2\mu_{03}^2 - 6\mu_{30}\mu_{21}^2\mu_{12} - 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 2\mu_{30}\mu_{12}^3 \\
&\quad + 3\mu_{21}^4 + 2\mu_{21}^3\mu_{03} + 3\mu_{21}^2\mu_{03}^2 - 6\mu_{21}\mu_{12}^2\mu_{03} + 3\mu_{12}^4 \\
I_7 &= -\mu_{30}^3\mu_{03} + 3\mu_{30}^2\mu_{21}\mu_{12} - 2\mu_{30}\mu_{21}^3 - 3\mu_{30}\mu_{21}^2\mu_{03} \\
&\quad + 6\mu_{30}\mu_{21}\mu_{12}^2 + 3\mu_{30}\mu_{12}^2\mu_{03} + \mu_{30}\mu_{03}^3 - 3\mu_{21}^3\mu_{12} \\
&\quad - 6\mu_{21}^2\mu_{12}\mu_{03} + 3\mu_{21}\mu_{12}^3 - 3\mu_{21}\mu_{12}\mu_{03}^2 + 2\mu_{12}^3\mu_{03} \\
I_8 &= -\mu_{30}^3\mu_{12} + \mu_{30}^2\mu_{21}^2 - \mu_{30}^2\mu_{21}\mu_{03} - 2\mu_{30}^2\mu_{12}^2 + 3\mu_{30}\mu_{21}^2\mu_{12} \\
&\quad - 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 3\mu_{30}\mu_{12}^3 - \mu_{30}\mu_{12}\mu_{03}^2 + 3\mu_{21}^3\mu_{03} \\
&\quad - 2\mu_{21}^2\mu_{03}^2 + 3\mu_{21}\mu_{12}^2\mu_{03} - \mu_{21}\mu_{03}^3 + \mu_{12}^2\mu_{03}^2 \\
I_9 &= \mu_{30}^4 + 6\mu_{30}^3\mu_{12} + 6\mu_{30}^2\mu_{21}\mu_{03} + 9\mu_{30}^2\mu_{12}^2 + 2\mu_{30}^2\mu_{03}^2 \\
&\quad + 18\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 6\mu_{30}\mu_{12}\mu_{03}^2 + 9\mu_{21}^2\mu_{03}^2 + 6\mu_{21}\mu_{03}^3 + \mu_{03}^4 \\
I_{10} &= \mu_{40}\mu_{04} - 4\mu_{31}\mu_{13} + 3\mu_{22}^2 \\
I_{11} &= -3\mu_{40}\mu_{22} - 2\mu_{40}\mu_{04} + 3\mu_{31}^2 + 2\mu_{31}\mu_{13} - 3\mu_{22}\mu_{04} + 3\mu_{13}^2 \\
I_{12} &= 3\mu_{40}^2 + 12\mu_{40}\mu_{22} + 2\mu_{40}\mu_{04} + 16\mu_{31}\mu_{13} + 12\mu_{22}\mu_{04} + 3\mu_{04}^2 \\
I_{13} &= (\mu_{50} + 2\mu_{32} + \mu_{14})^2 + (\mu_{05} + 2\mu_{23} + \mu_{41})^2 \\
I_{14} &= (\mu_{50} - 2\mu_{32} - 3\mu_{14})^2 + (\mu_{05} - 2\mu_{23} - 3\mu_{41})^2 \\
I_{15} &= (\mu_{50} - 10\mu_{32} + 5\mu_{14})^2 + (\mu_{05} - 10\mu_{23} + 5\mu_{41})^2
\end{aligned}
\right\} \tag{2.70}
$$

These 15 invariants are developed following the method in Frame 1. Let us note that I_2, I_3, I_4 are classic Hu invariants (see Equation (2.67)). There exist several methods for deriving moments-based invariants. For example, the image intensity function was expressed in polar coordinates $I(r, \theta)$ in [Reddi 81] and radial and angular moments were computed from it. He showed that the Hu invariants developed using algebraic invariant theory can be expressed solely as simple functions of the radial and angular moments. A set of rotational invariants using complex moments was developed by [Ghorbel 06] and [Flusser 09].

2.3.3.2 Invariance to translation

It was shown in Section 2.3.2.1 that centred moments are invariant to planar translation motions because the moments are defined with respect to the centroid as the origin. If the rotation invariants discussed above are computed from the centred moments, then we can obtain invariance to both rotation and translations. The coefficient vector $\mathbf{m}_p$ in Equation (2.69) can be replaced by centred moments $\boldsymbol{\mu}_p$ (which are already translation invariant) to obtain invariance to both translational and rotational motions.

Frame 1 Rotational Moment Invariants [VanGool 95] [Tahri 04]

Let us consider moments in n-dimensional space $m_{p1,\ldots,pn}$ whose spatial domain has undergone an orthogonal transformation (determinant = 1) defined by $\mathbf{x}'(t) = \mathbf{R}\,\mathbf{x}(t)$. Since $\mathbf{R}\mathbf{R}^\top = \mathbf{I_k}$, its time derivative can be obtained as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{X}(t) \tag{2.71}$$

where $\mathbf{A} = \mathbf{R}^\top\dot{\mathbf{R}}$. That being said, the variation in the moments after an orthogonal transformation are given by the following relation [Tahri 04]:

$$\dot{m}_{p1\ldots pn} = \sum_{i=1}^{n}\int\cdots\int_{\Omega} p_i x_i^{p_i-1}\dot{x}_i \prod_{j=1,j\neq i}^{n} x_j^{p_j} f(\mathbf{x})\mathrm{d}\mathbf{x} + \int\cdots\int_{\Omega} x_i^{p_i}\ldots x_n^{p_n}\dot{f}(\mathbf{x})\mathrm{d}\mathbf{x} \tag{2.72}$$

Under the hypothesis that the distribution function does not change, $\dot{f}(\mathbf{x}) = 0$ and hence we have

$$\dot{m}_{p1\ldots pn} = \sum_{i=1}^{n}\int\cdots\int_{\Omega} p_i x_i^{p_i-1}\dot{x}_i \prod_{j=1,j\neq i}^{n} x_j^{p_j} f(\mathbf{x})\mathrm{d}\mathbf{x} \tag{2.73}$$

As a simple case, let us consider $\mathbf{m}_p$, a vector with a collection of individual moments of a certain order p. Then, for each element in $\mathbf{m}_p$, we can use equation (2.73) to determine its time variation in terms of other elements in $\mathbf{m}_p$. These relationships can be assembled into a matrix form as $\dot{\mathbf{m}}_p = \mathbf{M}(\mathbf{A})\,\mathbf{m}_p$ where $\mathbf{M}(\mathbf{A})$ is a square matrix of dimension $l = dim(\mathbf{m}_p)$. More generally, the elements of $\mathbf{m}_p$ can consist of moment monomials of weight d with p^{th} order moments considered earlier. A vector of such monomials of moments of order p and weight d can be represented by $\mathbf{m}_p^d$ (For example, a polynomial with first order moments $p = 1$ and weight $d = 2$ is $\mathbf{m}_1^2 = \left[m_{10}^2, m_{10}m_{01}, m_{01}^2\right]$). The following general relationship can then be established.

$$\dot{\mathbf{m}}_p^d = \mathbf{M}(\mathbf{A})\,\mathbf{m}_p^d \tag{2.74}$$

A more detailed treatment with proof can be consulted from [Tahri 04].

Problem The problem of finding invariants to a certain transformation is then a search for polynomials $z(\mathbf{m}_p^d) = \boldsymbol{\alpha}^\top(\mathbf{m}_p^d)$ invariant to that transformation such that the following invariance postulate is satisfied.

$$\dot{z}(\mathbf{m}_p^d) = \boldsymbol{\alpha}^\top(\dot{\mathbf{m}}_p^d) = 0 \tag{2.75}$$

where each possible column of $\boldsymbol{\alpha}^\top$ represents the coefficients of a polynomial. It is in fact a tensor quantity. Here, we seek invariants to rotation which is an orthogonal transformation. So, by direct substitution from Equation (2.74), we have

$$\dot{z}(\mathbf{m}_p^d) = \boldsymbol{\alpha}^\top\,\mathbf{M}(\mathbf{A})\,\mathbf{m}_p^d = 0 \tag{2.76}$$

The problem of finding invariants is now distilled to determination of $\boldsymbol{\alpha}$ pertaining to the kernel of the matrix $\mathbf{M}(\mathbf{A})$. The number of independent invariants that can be obtained is equal to the dimension of this kernel. It is not guaranteed that $\dim\left(\ker(\mathbf{M}(\mathbf{A}))\right) \neq 0$ for all p and d. Nevertheless, about a dozen invariants upto fourth order moments have been obtained in [Tahri 05a] (and listed in Section 2.3.3.1 as well).

2.3.3.3 Invariance to uniform scaling

In visual servoing, scaled ratios of the moment polynomials established earlier are normally employed. This is in order to obtain invariance with respect to translation along the optic axis. To perform this scaling, it is necessary to study the effect of scaling transformation on the moments.
We recall that the centred moments of a 2D spatial distribution $f(x, y)$ are given by

$$\mu_{pq} = \iint (x - x_g)^p (y - y_g)^q \, f(x, y) \mathrm{d}x \mathrm{d}y \tag{2.77}$$

A uniform scaling transformation by s is governed by the following equation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.78}$$

Then, the centred moments after scaling can be written as

$$\mu'_{pq} = \iint s^p (x - x_g)^p s^q (y - y_g)^q \, f'(x, y) \, \mathrm{d}x' \mathrm{d}y' \tag{2.79}$$

Knowing that $\mathrm{d}x' = s \, \mathrm{d}x$ and $\mathrm{d}y' = s \, \mathrm{d}y$ and with the standard assumption that the value of the distribution itself does not change $f'(x, y) = f(x, y)$, we get the relation between the scaled and unscaled centred moments [Mukundan 98] [Flusser 09].

$$\mu'_{pq} = s^{p+q+2} \mu_{pq} \tag{2.80}$$

The above relation is indispensable when scaling moment functions that are typically composed of rotation moment invariants. Let β be the visual feature which is a ratio of $z_1(\boldsymbol{\mu}_{p1}^{d1})$ and $z_2(\boldsymbol{\mu}_{p2}^{d2})$ be two such functions. Then, to establish scale invariance $\beta' = \beta$, we need to determine the factor w such that the below invariance relation is always true.

$$\frac{z'_1(\boldsymbol{\mu}_{p1}^{d1})}{z'_2(\boldsymbol{\mu}_{p2}^{d2})} = \frac{z_1(\boldsymbol{\mu}_{p1}^{d1})}{[z_2(\boldsymbol{\mu}_{p2}^{d2})]^w} \tag{2.81}$$

where z'_1 and z'_2 are the moment polynomials after an unknown scaling transformation. Thus, what is sought is the scale invariance of simple functions of the moments to be employed as visual features. It is trivial to solve Equation (2.81) to obtain w, in the process of which s will actually be eliminated. It has to be noted that knowledge of the scaling factor s is not required to establish invariance to scale.

Thus, we dealt with the theory of moment invariants and procedures to obtain invariance to translations, rotations and scale (TRS). So far, we have introduced only the visual feature α, for controlling the rotational motions along the optic axis. In the following section, we will introduce the visual features used to control the remaining 2 dof, namely rotations around the x and y axes.

2.3.4 Visual Features based on Moment Invariants

[Chaumette 04] proposed the following 2 visual features for control of rotations around x and y axis respectively.

$$P_x = \frac{h_2}{h_1^2} \tag{2.82a}$$

$$P_y = \frac{\mu_{00}\, h_3}{h_1^3} \tag{2.82b}$$

where

$$h_1 = \mu_{20} + \mu_{02} \tag{2.83a}$$

$$h_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \tag{2.83b}$$

$$h_3 = (\mu_{30} - 3\mu_{12})^2 + (\mu_{03} - 3\mu_{21})^2 \tag{2.83c}$$

where the h_i are in fact a subset of the Hu's rotational invariants [Hu 62] (See Section 2.3.3.1). The interaction matrices of P_x and P_y are a function of the interaction matrix of the centred moments of second and third order. These can be computed from Equation (2.58) in a straightforward manner. If the target and camera planes are parallel, we have the following decoupled structure for the interaction matrix.

$$\begin{bmatrix} \mathbf{L}_{P_x}^{\parallel} \\ \mathbf{L}_{P_y}^{\parallel} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \mathtt{L}_{P_x}^{\omega_x} & \mathtt{L}_{P_x}^{\omega_y} & 0 \\ 0 & 0 & 0 & \mathtt{L}_{P_y}^{\omega_x} & \mathtt{L}_{P_y}^{\omega_y} & 0 \end{bmatrix} \tag{2.84}$$

When a symmetrical[5] target object is observed by the camera and its projection is perfectly centred in the image, moments of odd order vanish ($\mu_{30} = \mu_{21} = \mu_{12} = \mu_{03} = 0$) and the above interaction matrix becomes singular. That is, $\mathtt{L}_{P_x}^{\omega_x} = \mathtt{L}_{P_x}^{\omega_y} = 0$ and the rotational errors around x and y axes cannot be regulated. Obviously, this loss of rank of the interaction matrix is unfavourable for visual servoing. Thus, the possibility of encountering an *isolated singularity* exists and in fact can occur for any of the Hu invariants [Chaumette 04]. Therefore, these visual features can be used only with non-symmetrical targets.

Based on the previously introduced theory of moment invariants, a set of about 15 different invariants were developed in [Tahri 04]. (See Equation (2.70)). Then, the following are a set of 10 visual features which are simply ratios formed from the invariants introduced in Section 2.3.3.1.

$$\left.\begin{array}{lllll} c_1 = \dfrac{I_1}{I_2} & c_2 = \dfrac{I_3}{I_4} & c_3 = \dfrac{I_5}{I_6} & c_4 = \dfrac{I_7}{I_6} & c_5 = \dfrac{I_8}{I_6} \\ c_6 = \dfrac{I_9}{I_6} & c_7 = \dfrac{I_{11}}{I_{10}} & c_8 = \dfrac{I_{12}}{I_{10}} & c_9 = \dfrac{I_{13}}{I_{15}} & c_{10} = \dfrac{I_{14}}{I_{15}} \end{array}\right\} \tag{2.85}$$

So, the property of invariance to optic axis rotations should be retained in this case. The interaction matrix of these invariants exhibited the following form

$$\mathbf{L}_{I_i}^{\parallel} = \begin{bmatrix} 0 & 0 & L_{I_i}^{v_x} & L_{I_i}^{\omega_x} & L_{I_i}^{\omega_y} & 0 \end{bmatrix} \tag{2.86}$$

The interaction matrix of any visual feature $c_i = \dfrac{I_p}{I_q}$ can be easily developed as

$$\mathbf{L}_{c_i} = \frac{1}{I_q}\left(\mathbf{L}_{I_p} - c_i\, \mathbf{L}_{I_q}\right) \tag{2.87}$$

[5] symmetrical here refers to axial symmetry, i.e., about the coordinate axes

The interaction matrix of these features retain the same decoupled structure as in the case of P_x and P_y discussed previously. That is, for any c_i, when the target and the sensor planes are parallel, we have

$$\mathbf{L}_{c_i}^{||} = \begin{bmatrix} 0 & 0 & 0 & \mathrm{L}_{c_i}^{\omega_x} & \mathrm{L}_{c_i}^{\omega_y} & 0 \end{bmatrix} \tag{2.88}$$

When there is a symmetry however, $I_q = 0$ in Equation (2.87) (not for all c_i, for example, c_1 is an exception since $I_2 \neq 0$) and $\mathbf{L}_{c_i}$ in such a case is not defined. In these scenarios, the visual features proposed in Equation (2.85) will not be useful.

For symmetrical targets, [Chaumette 04] developed the following set of visual features. These features are reported to have been developed after extensive trials with various normalizations such that s_x is as independent as possible from ω_y and s_y is independent from ω_x.

$$s_x = \frac{\sqrt{m_{00}}\, I_x}{h_2 h_1^{3/2}} \tag{2.89a}$$

$$s_y = \frac{\sqrt{m_{00}}\, I_y}{h_2 h_1^{3/2}} \tag{2.89b}$$

where

$$I_x = (\mu_{03} - 3\mu_{21})[(\mu_{20} - \mu_{02})^2 - 4\mu_{11}^2] + (\mu_{30} - 3\mu_{12})4\mu_{11}(\mu_{20} - \mu_{02}) \tag{2.90a}$$

$$I_y = (\mu_{30} - 3\mu_{12})[(\mu_{20} - \mu_{02})^2 - 4\mu_{11}^2] - (\mu_{03} - 3\mu_{21})4\mu_{11}(\mu_{20} - \mu_{02}) \tag{2.90b}$$

Their interaction matrix is of the following structure

$$\begin{bmatrix} \mathbf{L}_{s_x}^{||} \\ \mathbf{L}_{s_y}^{||} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \mathrm{L}_{s_x}^{\omega_x} & \mathrm{L}_{s_x}^{\omega_y} & \mathrm{L}_{s_x}^{\omega_z} \\ 0 & 0 & 0 & \mathrm{L}_{s_y}^{\omega_x} & \mathrm{L}_{s_y}^{\omega_y} & \mathrm{L}_{s_y}^{\omega_z} \end{bmatrix} \tag{2.91}$$

In equation (2.91), $\mathrm{L}_{s_x}^{\omega_z} \neq 0$ and $\mathrm{L}_{s_x}^{\omega_z} \neq 0$. We see that these features are no more invariant to 2D rotations. This is not surprising since the moment polynomials are not from the Hu Invariants.

From the above discussions, it can be inferred that the problem of finding an appropriate set of visual features for controlling x and y rotations is not solved. A separate set of visual features needs to be employed depending on the symmetrical nature of the target object.

2.3.4.1 Selection Method

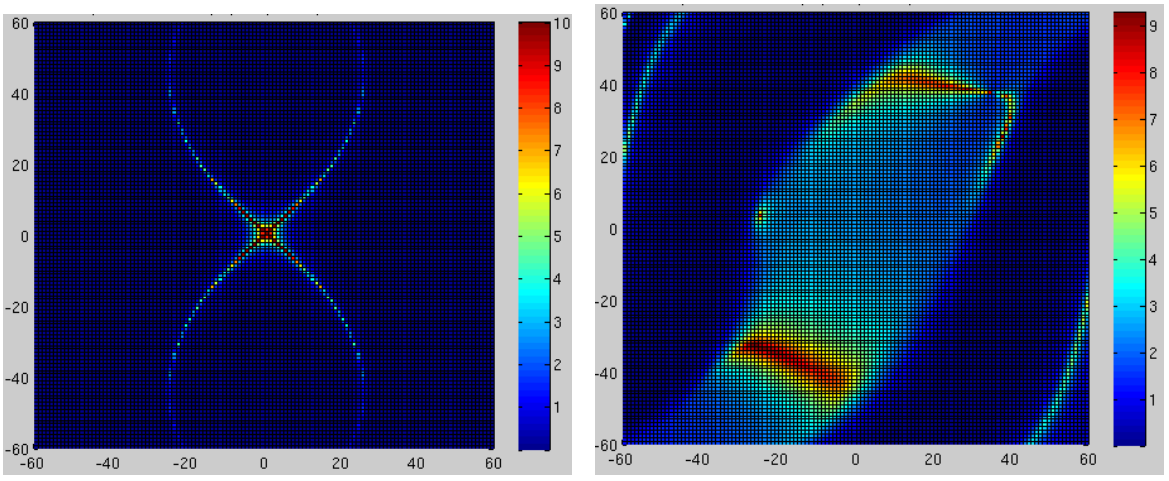
While various visual features have been proposed in literature, there are few works which explicitly deal with issues of selection from among a set of visual features. In [Tahri 05a], it is recommended to choose a pair of visual features after an analysis of their behaviour around the desired equilibrium state. To perform this analysis, the following error function is proposed.

$$e_r(\theta_x, \theta_y) = \sum_{i=1}^{n} [c_i(\theta_x, \theta_y) - c_i^*]^2 \tag{2.92}$$

where θ_x and θ_y are the rotation angles around the x and y axes. For any visual feature pair (c_i, c_j), e_r is evaluated for a finite range of angles ($\theta_x, \theta_y \in [-\pi/3, \pi/3]$ for instance) around the desired equilibrium with a step size of required granularity (1° increments or less). The error function can also be easily visualised from

$$f(e_r) = \frac{1}{0.1 + e_r} \tag{2.93}$$

The above evaluations are performed for various potential visual feature pairs. Then, the feature pair for which e_r presents a globally minimal value and the maximum influence zone is then selected. What exactly constitutes the influence zone is not quantified. However, manual selection is possible through visualisation of $f(e_r)$. Example visualisations of $f(e_r)$ for the visual feature pair (s_x, s_y) (see Equation (2.89)) computed for a rectangular region and for a pentagonal region are shown in Figure 2.3. The cost function in Figure 2.3(a) is considered better for visual servoing than the one in Figure 2.3(a) since the optimum for the former occurs at the desired rotation angle configuration (namely $0, 0$) and exhibits higher values at all states other than the solution. For a set consisting of



(a) $f(e_r)$ for (s_x, s_y) for a $0.4m \times 0.2m$ rectangle

(b) $f(e_r)$ for (s_x, s_y) for a pentagon

Figure 2.3 – Visualization of $f(e_r)$ evaluated for $\theta_x, \theta_y \in [-\pi/3, \pi/3]$

10 visual features, this analysis has to be performed to select from among ${}^{10}C_2$ pairs, that is for 45 possible choices of the visual features. The problem is that this manual selection is exhausting and could easily become error-prone.

Problems in current state of the art pertaining to visual features to control rotations around the x and y axes is situated at the intersection of two different levels. The first is at the design level : cooking of appropriate visual features which can be used regardless of the shape and/or symmetrical nature of the target. The second is concerned with an appropriate procedure for visual feature selection when several options are available.

The following section deals with the idea of shifted moments which was aimed to find visual features that can work irrespective of the shape of the target.

2.4 Tunable Visual Features

The theory of moment invariants and a general procedure for obtaining moment-based functions invariant to planar translations, rotation and scale (TRS) was introduced in Section 2.3. Then, existing visual features in the state of the art were introduced followed a brief overview of a method used for selecting them. This context is necessary for a clear understanding of tunable visual features, our contribution. As mentioned before, a challenging aspect in visual servoing is the simultaneous control of the rotational motions around the x and y camera axes, when all 6dof of the robot have to be controlled. This contribution is a simple possible direction for dealing with this non-trivial problem.

2.4.1 Shifted Moments

In [Tamtsia 13b], a new type of image moments called shifted moments were proposed. A shifted moment of order $p+q$ is defined as follows:

$$\tilde{\mu}_{pq} = \iint (x - x_g + x_{sh})^p (y - y_g + y_{sh})^q \, \mathrm{d}x \, \mathrm{d}y \tag{2.94}$$

As seen from Equation (2.94), the moments are computed with reference to the shift parameters x_{sh} and y_{sh}. This can be be interpreted as the centred moments subjected to a shift specified by (x_{sh}, y_{sh}). Indeed, if $x_{sh} = y_{sh} = 0$, Equation (2.94) reduces to the case of centred moments, as in Equation (2.53). The tuple (x_{sh}, y_{sh}) can be regarded as the coordinates of an image point, which we refer to as the shift point (See Figure 2.4). Its coordinates resemble the following form

$$x_{sh} = x_g + \tilde{x} \tag{2.95a}$$

$$y_{sh} = y_g + \tilde{y} \tag{2.95b}$$

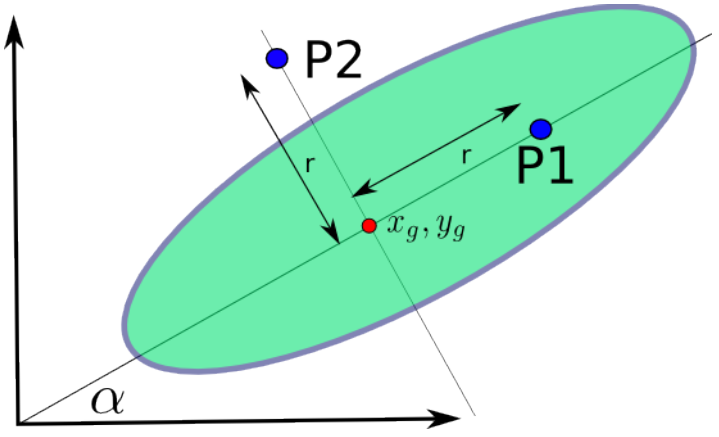


Figure 2.4 – Shift points $\mathbf{P}_1(x_g + \tilde{x})$ and $\mathbf{P}_2(y_g + \tilde{y})$ are defined on orthogonal axes. Shifted moments are computed with reference to these shift points

In [Tamtsia 13b], moments are computed with reference to two unique shift points $\mathbf{P}_1(x_{sh1}, y_{sh1})$ and $\mathbf{P}_2(x_{sh2}, y_{sh2})$ (see Fig 2.4). For the first shift point, we have

$$x_{sh1} = r\cos(\alpha) \tag{2.96a}$$

$$y_{sh1} = r\sin(\alpha) \tag{2.96b}$$

where r is a scalar given by

$$r = (\mu_{20} + \mu_{02})^{\frac{1}{4}} \tag{2.97}$$

and α is the orientation as in Equation (2.59). Similarly, the coordinates of the second shift point are chosen along an axis perpendicular to the axis along which $\mathbf{P}_1$ is defined.

$$x_{sh2} = r\cos(\alpha + \frac{\pi}{2}) \tag{2.98a}$$

$$y_{sh2} = r\sin(\alpha + \frac{\pi}{2}) \tag{2.98b}$$

With this choice of orientation, the two shift points can be considered to be along the major and minor orientation axes of the orientation ellipse. In the same work, alternate choices are also proposed for r by choosing the points at different distances r_1 and r_2.

$$r_1 = \sqrt{\frac{2}{\mu_{00}}\left(\mu_{20} + \mu_{02} + \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}\right)} \tag{2.99}$$

$$r_2 = \sqrt{\frac{2}{\mu_{00}}\left(\mu_{20} + \mu_{02} - \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}\right)} \tag{2.100}$$

The rationale behind the choice of r is that it should be rotationally invariant as well as vary consistently with the scaling undergone by the target object in the image. Equation (2.94) can be further expanded using the binomial theorem as follows:

$$\tilde{\mu}_{pq} = \iint \left[\sum_{k=0}^{p} \binom{p}{k} (x - x_g)^{p-k}\, x_{sh}^{k} \sum_{l=0}^{q} \binom{q}{l} (y - y_g)^{q-l}\, y_{sh}^{l}\right] \mathrm{d}x\, \mathrm{d}y \tag{2.101}$$

Further, using the relation in Equation (2.54), we can write

$$\tilde{\mu}_{pq} = \sum_{k=0}^{p}\sum_{l=0}^{q} \binom{p}{k}\binom{q}{l} x_{sh}^{k}\, y_{sh}^{l}\, \mu_{p-k,q-l} \tag{2.102}$$

The relation between the shifted moments and the centred moments is thus obtained in equation (2.102). Differentiating this equation (similar to developments in Section 2.3.2.1), we can obtain the interaction matrix of the shifted moments.

$$\left.\begin{aligned} \mathbf{L}_{\tilde{\mu}_{pq}} = {} & \mathbf{L}_{x_{sh}} \left(\sum_{k=0}^{p}\sum_{l=0}^{q} \binom{p}{k}\binom{q}{l} k x_{sh}^{k-1} y_{sh}^{l} \mu_{p-k,q-l}\right) \\ & + \mathbf{L}_{y_{sh}} \left(\sum_{k=0}^{p}\sum_{l=0}^{q} \binom{p}{k}\binom{q}{l} l x_{sh}^{p} y_{sh}^{l-1} \mu_{p-k,q-l}\right) \\ & + \mathbf{L}_{\mu_{p-k,q-l}} \left(\sum_{k=0}^{p}\sum_{l=0}^{q} \binom{p}{k}\binom{q}{l} x_{sh}^{k} y_{sh}^{l}\right) \end{aligned}\right\} \tag{2.103}$$

where $\mathbf{L}_{x_{sh}}$ and $\mathbf{L}_{y_{sh}}$ are based on how the shift parameters are defined and straightforward to obtain. For the shift point defined by Equations (2.96), we have

$$\mathbf{L}_{x_{sh1}} = \mathbf{L}_r \cos\alpha - r\mathbf{L}_\alpha \sin\alpha \tag{2.104a}$$

$$\mathbf{L}_{y_{sh1}} = \mathbf{L}_r \sin\alpha + r\mathbf{L}_\alpha \cos\alpha \tag{2.104b}$$

Similarly, for the second shift point , we have

$$\mathbf{L}_{x_{sh2}} = \mathbf{L}_r \cos(\alpha + \pi/2) - r\mathbf{L}_\alpha \sin(\alpha + \pi/2) \tag{2.105a}$$

$$\mathbf{L}_{y_{sh2}} = \mathbf{L}_r \sin(\alpha + \pi/2) + r\mathbf{L}_\alpha \cos(\alpha + \pi/2) \tag{2.105b}$$

where, if r is defined by Equation 2.97,

$$\mathbf{L}_r = \frac{1}{4}(\mu_{20} + \mu_{02})^{-\frac{3}{4}}(\mathbf{L}_{\mu_{20}} + \mathbf{L}_{\mu_{02}}) \tag{2.106}$$

and $\mathbf{L}_\alpha$ is the interaction matrix of the orientation feature discussed previously. The work in [Tamtsia 13b] breathed a fresh perspective to the problem of control of rotational degrees of freedom perpendicular to the optic axis. *Most importantly, all the invariant properties of the moment invariants introduced in Section 2.3.3 are still valid when those invariants are computed with the shifted moments. In other words, substitution of the centred moments by shifted moments does not alter their invariance properties.*

2.4.1.1 Visual Feature based on Shifted Moments

Once the analytical forms for the shifted moments upto a specific order (Equation (2.94)) and their interaction matrices (Equation (2.103)) are available, the theory of moment invariants can be applied to obtain visual features invariant to planar translations, rotations and scale (TRS). The following visual features were used in [Tamtsia 13b] for visual servoing using moments computed from a set of discrete points.

$$rs_1 = \frac{I_{s2}}{I_{s1}^{5/4}} \tag{2.107}$$

$$rs_2 = \frac{I_{s3}}{I_{s1}^{5/4}} \tag{2.108}$$

$$rs_3 = \frac{I_{s3}}{I_{s2}} \tag{2.109}$$

where

$$I_{s1} = \tilde{\mu}_{20}\,\tilde{\mu}_{02} - \tilde{\mu}_{11}^2 \tag{2.110}$$

$$I_{s2} = -\tilde{\mu}_{30}\,\tilde{\mu}_{12} + \tilde{\mu}_{21}^2 - \tilde{\mu}_{03}\,\tilde{\mu}_{21} + \tilde{\mu}_{12}^2 \tag{2.111}$$

$$I_{s3} = 3\tilde{\mu}_{30}\,\tilde{\mu}_{12} + \tilde{\mu}_{30}^2 + 3\tilde{\mu}_{03}\tilde{\mu}_{21} + \tilde{\mu}_{03}^2 \tag{2.112}$$

The polynomials I_{s1}, I_{s2} and I_{s3} are invariant to TRS transformations and were developed in [Tahri 04] (in Appendix A.1). Let us note that the fundamental principles/methodology for obtaining these polynomials are the same as those listed in Equation (2.70). Similarly, the visual features rs_i above are simple ratios of the invariant polynomials, identical to how features in Equation (2.85) were constructed from Equation (2.70). The difference now is that shifted moments are now used rather than the centred moments. Further, these are not null even for symmetrical objects, thanks to the shifted moments.

- The translational invariance of rs_1, rs_2 and rs_3 result from the fact that shifted moments are obtained directly in terms of the centred moments.

- Clearly, both I_{s2} and I_{s3} are built from shifted moments of third order. So, they are affected by scale change in the same way and naturally rs_3 is invariant to scale. On the other hand, rs_1 is obtained by scaling I_{s1} appropriately as discussed in Section 2.3.3.3.

The interaction matrices of rs_1, rs_2 and rs_3 are of the following form when the sensor and target planes are parallel to each other.

$$\mathbf{L}^{||}_{rs_i} = \begin{bmatrix} 0 & 0 & 0 & \mathrm{L}^{\omega_x}_{rs_i} & \mathrm{L}^{\omega_y}_{rs_i} & 0 \end{bmatrix} \tag{2.113}$$

Existing state of the art was described thus far which was deemed necessary for a clear understanding of the contributions and ideas in this thesis. Henceforth, our contributions would be presented. First, an improved version of the shifted moments which allows to tune the visual servoing characteristics is introduced.

2.4.2 Generalized Shifted Moments

In the previous section, we saw that the shifted moments were computed along vectors oriented at 0 and $\pi/2$ with respect to the orientation of the target object in the image. We will see in the following that this is not in all cases the best choice. A bad choice of these shift angles might lead to instability, singularities in the interaction matrix and even failure of the control law. A more generalized version of the shifted moments is presented in this section. These moments are computed with respect to tunable shift points. Visual features customized to satisfy specific criteria can be built from these generalized shifted moments. It will be shown that a careful choice of these shift angles augurs well for stability of the visual servoing.

Two shift points of the following form are proposed.

`Tunable shift point` $\mathbf{P}_1$

$$x_{sh1} = \sqrt{m_{00}}\,\cos(\alpha + \Delta_1) \tag{2.114a}$$

$$y_{sh1} = \sqrt{m_{00}}\,\sin(\alpha + \Delta_1) \tag{2.114b}$$

`Tunable shift point` $\mathbf{P}_2$

$$x_{sh2} = \sqrt{m_{00}}\,\cos(\alpha + \Delta_2) \tag{2.115a}$$

$$y_{sh2} = \sqrt{m_{00}}\,\sin(\alpha + \Delta_2) \tag{2.115b}$$

where $\Delta_1 \in \mathbb{R}$ and $\Delta_2 \in \mathbb{R}$ are shift angles to be chosen in an appropriate manner. Therefore, the shift points are not necessarily orthogonal to each other. First, we note that, $r = \sqrt{m_{00}}$ instead of the second order moments as used previously in Equation (2.97). In fact, $r \in \mathbb{R}$ is a scalar defined such that it respects the following two simple conditions.

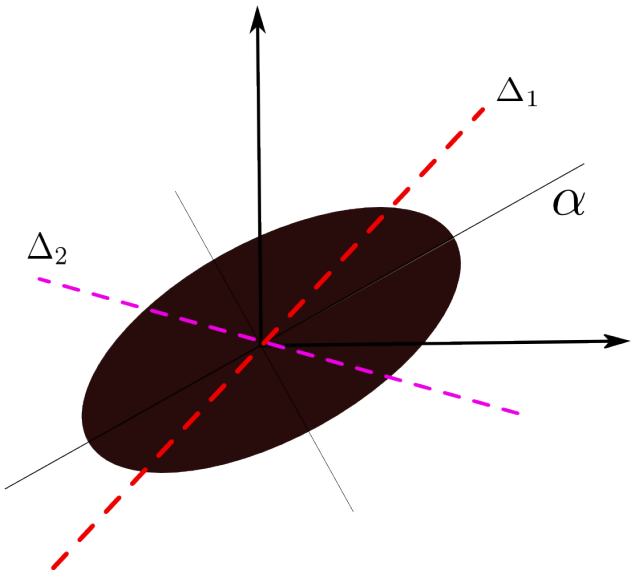


Figure 2.5 – Illustration for shifted moments. α is along the major axis of the orientation ellipse. Δ_1 and Δ_2 are shifts in orientation that define the actual shift points

1. r is invariant to rotational transformations in the plane. That is, $r(\mathbf{I}(\mathbf{x}^{'})) = r(\mathbf{I}(\mathbf{x}))$ when the image coordinates have undergone a planar rotation as in $\mathbf{x}^{'} = \mathbf{R}\mathbf{x}$, where $\mathbf{R} \in \mathbb{SO}(2)$. If r is rotationally invariant, then the coordinates of the shift point (See for instance Equation (2.96) related to shift point $\mathbf{P}_1$) varies in direct proportion to the amount of planar rotation (captured by α).

2. r should vary in direct proportion to scaling undergone by the image. If the image coordinates are scaled by λ, that is, $\mathbf{x}^{'} = \lambda\mathbf{x}$, then $r' = \lambda r$.

Clearly, $(\mu_{20} + \mu_{02})$ used in Equation (2.97) is invariant to 2D rotational transformations. Further, from equation (2.80), we have $(\mu^{'}_{20} + \mu^{'}_{02}) = \lambda^4(\mu_{20} + \mu_{02})$. Raising both sides of this equality to power $1/4$, we have $(\mu^{'}_{20} + \mu^{'}_{02})^{\frac{1}{4}} = \lambda(\mu_{20} + \mu_{02})^{\frac{1}{4}}$. This explains the choice of $r = (\mu_{20} + \mu_{02})^{\frac{1}{4}}$, which satisfies the conditions stated above. So, different choices are indeed possible for r. In a similar way, when choosing the zeroth order moment, we have the following relation between the scaled and unscaled moment: $m^{'}_{00} = \lambda^2 m_{00}$. Taking square root on both sides, we have $r = \sqrt{m_{00}}$. Choosing r based on the area, r will inherit the invariance properties of the area moment. In general, when having to choose between a higher and lower order moment, it is always advisable to stick to moments of lower order. The first obvious reason is because earlier research proved the susceptibility of higher order moments to noise [Teh 88]. Second, computation of the interaction matrix of a moment of order $p+q$ requires moments upto order $p+q+1$ (See Equation (2.29). A judicious choice with appropriate visual features has positive implications for the stability of the system.

Next, if we differentiate Equations 2.114, the interaction matrix of the first generalized shift point can be obtained.

$$\mathbf{L}_{x_{sh1}} = \mathbf{L}_r \cos(\alpha + \Delta_1) - r\mathbf{L}_\alpha \sin(\alpha + \Delta_1) \tag{2.116a}$$

$$\mathbf{L}_{y_{sh1}} = \mathbf{L}_r \sin(\alpha + \Delta_1) + r\mathbf{L}_\alpha \cos(\alpha + \Delta_1) \tag{2.116b}$$

Similarly, from the second generalized shift point in 2.115, we have

$$\mathbf{L}_{x_{sh2}} = \mathbf{L}_r \cos(\alpha + \Delta_2) - r\mathbf{L}_\alpha \sin(\alpha + \Delta_2) \tag{2.117a}$$
$$\mathbf{L}_{y_{sh2}} = \mathbf{L}_r \sin(\alpha + \Delta_2) + r\mathbf{L}_\alpha \cos(\alpha + \Delta_2) \tag{2.117b}$$

where the interaction matrix of r is given by

$$\mathbf{L}_r = \frac{1}{2}(m_{00})^{-\frac{1}{2}}\mathbf{L}_{m_{00}} \tag{2.118}$$

Substitution from 2.33 and considering that we are using a dense case with $\delta = 1$, we get

$$\mathbf{L}_r = \begin{bmatrix} -A\frac{\sqrt{m_{00}}}{2} & -B\frac{\sqrt{m_{00}}}{2} & \frac{\sqrt{m_{00}}}{2}\left(\frac{3}{Z_g} - C\right) & \frac{3\sqrt{m_{00}}}{2}y_g & -\frac{3\sqrt{m_{00}}}{2}x_g & 0 \end{bmatrix} \tag{2.119}$$

As noted previously, r is rotation invariant. Further, when the camera and target planes are parallel, we further obtain invariance to planar translations.

$$\mathbf{L}_r^{||} = \begin{bmatrix} 0 & 0 & \frac{\sqrt{m_{00}}}{2}\left(\frac{3}{Z_g} - C\right) & \frac{3\sqrt{m_{00}}}{2}y_g & -\frac{3\sqrt{m_{00}}}{2}x_g & 0 \end{bmatrix} \tag{2.120}$$

Then, using Equations (2.116) and (2.117) pertaining to the generalized shift points into Equation (2.103), the interaction matrix for the shifted moments with respect to $\mathbf{P}_1$ and $\mathbf{P}_2$ can be obtained.

2.4.3 Selection Methods for Generalized Shifted Moments

In moments-based visual servoing, there exist a set of visual features based on the centre of gravity and area which are chosen primarily to control the 3D translations. Similarly, the orientation feature based on the second order centred moments is intended to control the rotation motions around the optic axis. It is of great interest to find such a standard visual feature set to control the rotations around the x and y axes. The generalized shifted moments are proposed with this motivation. Let us recall that these moments and their corresponding interaction matrices are a function of the moments computed with respect to tunable shift points $\mathbf{P}_1$ and $\mathbf{P}_2$ in Equations (2.114) and (2.115). Next, we present a few methods for their selection such that they are optimal with respect to specific criteria. The selection methods are performed using the image learnt from the desired robot pose.

2.4.3.1 Selection of Δ_1

Let us consider the interaction matrix for a visual feature developed from moment invariants using the generalized shifted moments. This interaction matrix is of the following general form[6]:

$$\mathbf{L}_i = \begin{bmatrix} L^{v_x}_{\mathtt{p}_1} & L^{v_y}_{\mathtt{p}_1} & L^{v_z}_{\mathtt{p}_1} & L^{\omega_x}_{\mathtt{p}_1} & L^{\omega_y}_{\mathtt{p}_1} & 0 \\ L^{v_x}_{\mathtt{p}_2} & L^{v_y}_{\mathtt{p}_2} & L^{v_z}_{\mathtt{p}_2} & L^{\omega_x}_{\mathtt{p}_2} & L^{\omega_y}_{\mathtt{p}_2} & 0 \end{bmatrix} \tag{2.121}$$

[6]In each entry of $\mathbf{L}$, the subscript signifies the shift point p_i at which the invariant is evaluated. The superscript v refers to the translational velocity and ω to the rotational component of the camera velocity screw, in either of x, y or z axes

Each entry in the interaction matrix of a visual feature represents the responsiveness of that visual feature i to a particular spatial motion. Conversely, in the interaction matrix inverse, this responsiveness is towards errors in the feature intended for controlling that particular motion. Also, when the camera and target planes are parallel to each other, we have the following excellent decoupled structure:

$$\mathbf{L}_i^{||} = \begin{bmatrix} 0 & 0 & 0 & L_{\mathrm{p}_1}^{\omega_x} & L_{\mathrm{p}_1}^{\omega_y} & 0 \\ 0 & 0 & 0 & L_{\mathrm{p}_2}^{\omega_x} & L_{\mathrm{p}_2}^{\omega_y} & 0 \end{bmatrix} \tag{2.122}$$

By selecting an optimal Δ_1 connected to the first shift point, we can obtain an interaction matrix with optimal responsiveness with respect to ω_x and ω_y. This can be formally stated as the following optimization problem:

$$\Delta_1^* = \max_{\Delta_1} \quad \mathcal{F}_1(\Delta_1) \tag{2.123}$$

where

$$\mathcal{F}_1(\Delta_1) = \left\| \mathbf{L}_{\mathrm{p}_1}^{\omega} \right\| = \left\| \begin{bmatrix} L_{\mathrm{p}_1}^{\omega_x} & L_{\mathrm{p}_1}^{\omega_y} \end{bmatrix} \right\| \tag{2.124}$$

where the operator $\|.\|$ stands for l_2 norm. Indeed, the solution to this problem depends on the image content. When the image of the target observed by the camera is non-symmetrical[7], a unique solution to this optimization problem was found. In other cases speaking generally, multiple solutions are possible. This is indeed dependent upon both the visual feature selected and also the shape of the object. In such cases, one among these solutions can be chosen as the optimal value. Once this optimal value for Δ_1 is obtained, the first shift point can be held constant at this value $\Delta_1 = \Delta_1^*$ and the method described next can be used to obtain the second shift angle.

2.4.3.2 Selection of Δ_2

From Equation (2.121), let us consider only the portion of the interaction matrix that is related to the x and y rotational motions. This 2×2 partial interaction matrix can be written as

$$\mathbf{L}_{\omega_{x,y}} = \begin{bmatrix} L_{\mathrm{p}_1}^{\omega_x} & L_{\mathrm{p}_1}^{\omega_y} \\ L_{\mathrm{p}_2}^{\omega_x} & L_{\mathrm{p}_2}^{\omega_y} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{\mathrm{p}_1} \\ \mathbf{L}_{\mathrm{p}_2} \end{bmatrix} \tag{2.125}$$

The selection of Δ_2 should be such that the simultaneous control of both the rotations is optimal. We propose to choose Δ_2 such that the factors responsible for the x and y rotational velocities are perpendicular to each other. That is, Δ_2^* should belong to the solution set $\mathcal{D}$ that satisfies the orthogonality of the vectors $\mathbf{L}_{\mathrm{p}_1}$ and $\mathbf{L}_{\mathrm{p}_2}$.

$$\Delta_2^* \in \mathbb{R} = \{ d : \forall d \in \mathcal{D},\ \mathcal{F}_2(d) \simeq 0 \} \tag{2.126}$$

where for any possible angle $\Delta_2 \in \mathbb{R}$, the objective function is

$$\mathcal{F}_2(\Delta_2) = \kappa_{dn}(\mathbf{L}_{\omega_{x,y}}) = \frac{\mathbf{L}_{\mathrm{p}_1}^{\top} . \mathbf{L}_{\mathrm{p}_2}}{\|\mathbf{L}_{\mathrm{p}_2}\|} \tag{2.127}$$

[7] either due to the target itself being non-symmetrical or a symmetrical target observed from a non-parallel desired configuration

where κ_{dn} is a function of the partial interaction matrix defined in Equation (2.125) and represents the angle between the two vectors defined by $\mathbf{L}_{\mathrm{p}_1}$ and $\mathbf{L}_{\mathrm{p}_2}$. Clearly, these two vectors are a function of the respective visual feature chosen to control the x and y rotations. With this optimization, these vectors are optimized in such a way that their directions are perpendicular to each other. For every choice of Δ_1 as obtained above, there could be multiple choices for Δ_2, all of which satisfy the orthogonality criterion proposed in Equation (2.127). In this case, one among those values, the one closest to 0 can be selected.

2.4.3.3 Joint Selection of Δ_1 and Δ_2

In Sections 2.4.3.1 and 2.4.3.2, we saw that the shift points can be selected such that each of them satisfies separate objectives. Alternatively, these shift angles can be selected such that they are jointly optimal with respect to a common objective. The following objective function is proposed to select the shift angles that result in the best possible conditioning of the interaction matrix.

$$\mathcal{J}(\Delta_1, \Delta_2) = \frac{1}{\kappa\left(\mathbf{L}_s(\Delta_1, \Delta_2)\right)} \tag{2.128}$$

where κ is the condition number of the interaction matrix. With this objective function, the influence of the shift angles on the system conditioning can be studied. Hence, the choice of these angles is vital for the stability and robustness of the visual servoing. The condition number of the interaction matrix was used in [Feddema 91].

We recall to the reader that the visual features are now calculated using the tunable shift points (2.114),(2.115) introduced in this paper. Finally, it is be noted that the selection criteria presented above are applicable to both symmetrical and non-symmetrical objects observed in the image, as shown in the case studies.

We have thus proposed some metrics for visual feature selection that are tied to responsiveness with respect to ω_x and ω_y, orthogonality in components of partial interaction matrix and to the conditioning of the system. In previous literature, measures such as resolvability[Nelson 95] and motion perceptibility[Sharma 97] were also based on the interaction matrix. To the best of our knowledge, this is the first work on moments-based visual servoing where the moments are defined based on objective characteristics that would result from the interaction matrix. We would also like to point out that the criteria proposed above are not the best in all cases and for all visual features. They can be adopted according to the requirements envisaged. For instance, when it is expected that there are strong perturbations might affect the system, it is best to choose the interaction matrix with optimal conditioning. What is most interesting are the wide possibilities that are opened by adopting this methodology for defining the moments.

2.4.4 Application to Tunable Visual Features

In this section, we present studies where the proposed selection criteria are applied to existing shifted moment-based visual features defined in Equations (2.114) and (2.115). These were described in Section 2.4.1.1, referred to by rs_1, rs_2 and rs_3. The studies have been done first with a symmetrical target and later with a non-symmetrical target. Also, we chose binary moments for this study to serve as a validation step before moving on to

the photometric moments in Chapter 3. The first desired pose ${}^{c*}\mathbf{M}_o$ is chosen such that the camera and the target planes are parallel at a depth of 1.0m above the target. That is ${}^{c*}\mathbf{M}_o = [0.0, 0.0, 1m, 0.0°, 0.0°, 0.0°]$. The second desired pose is a non-parallel configuration where there is a non-trivial rotation between the sensor and target orientations, given by ${}^{c*}\mathbf{M}_o = [20.0cm, -20.0cm, 280cm, -20.0°, 20.0°, 20.0°]$. Typically, there is a very large displacement in depth. These are designated as the parallel desired (PD) and non-parallel desired (NPD) configurations in the following discussion. We assume a calibrated camera that acquires images of size 640×480. The camera simulator was developed by interfacing to the ViSP software library [Marchand 05]. For each visual feature, the selection metrics proposed in Equations (2.123), (2.127) and (2.128) were used for Δ_1 and Δ_2 respectively.

(a) In parallel desired configuration (b) In non-parallel desired configuration

Figure 2.6 – Images acquired from symmetrical object

Symmetrical Case

For the symmetrical case, the object observed by the camera is a simple white rectangle. In the case where the sensor and the object planes are parallel, the acquired image is shown in Fig 2.6(a). The possible design space with respect to each of the proposed metrics is shown in Figures 2.8 to 2.11. For all the visual features considered, the maximum responsiveness criterion in Equation (2.123) can be achieved with more than one solution. For rs_1 and rs_3, there are 2 solutions at $0°$ and $180°$ as shown in Figs 2.8(a) and 2.11(a). In the case of rs_2, there are 4 solutions at $62°$, $118°$, $242°$ and $298°$ for the first shift angle. There is a symmetry at $180°$ for all these features, provided the rectangle is perfectly centred in the image. Corresponding to each of these solutions of Δ_1, a solution exists for the second shift angle Δ_2 such that these together satisfy the orthogonality criterion. Let us consider first rs_2. For each of the 4 choices of Δ_1 for rs_2, there are two choices of Δ_2 such that orthogonality will result. These are shown in Figs. 2.9(b) - 2.9(e). For rs_1 and rs_3, as shown in Figs 2.8(b) and 2.11(b), there are six solutions from $30°$ to $330°$ at $60°$ intervals. Due to the symmetrical shape of the object, it is logical to observe that these solutions for orthogonality are the same for either of the choices $\Delta_1 = 0°$ or $\Delta_1 = 180°$. Further, let us note that the default choice of $\Delta_1 = 0°$ and $\Delta_2 = 90°$ made

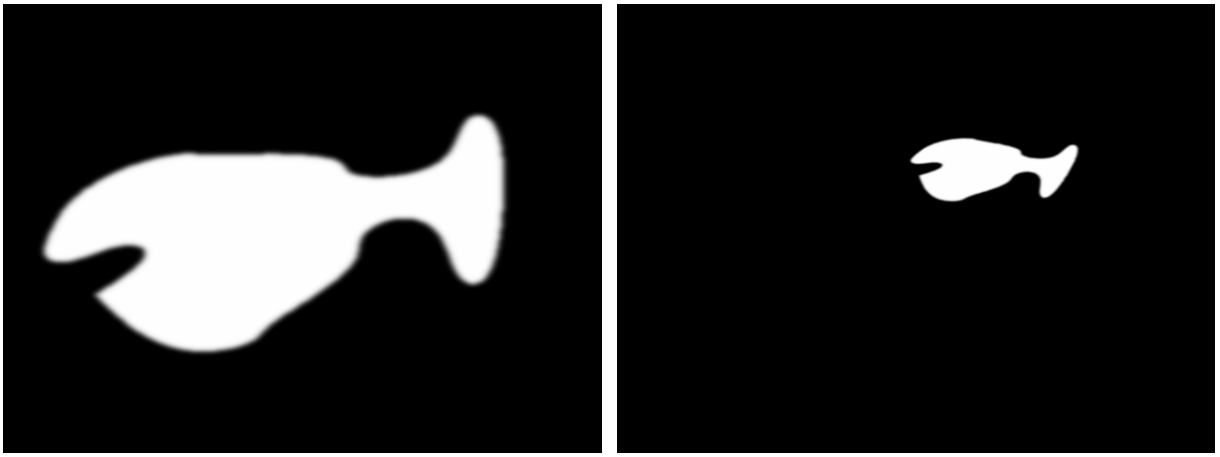
in [Tamtsia 13b] is therefore only 1 among the different possible combinations that can guarantee orthogonality. Finally, a beautiful symmetry manifests itself at $180°$. In case of rs_2, no such symmetry is exhibited. Further, we see here that the default choice of $0°$ and $90°$ for the shift angles guarantees neither a maximal responsiveness nor an orthogonality. We thus see that there are less intuitive findings but not surprising because the proposed criteria are empirical and based on the interaction matrix which is a function of not only on the image but indeed also on the visual feature used.

The effect of these angles on the system conditioning for rs_1 can be seen from Figure 2.8(c), rs_2 from 2.10 and rs_3 from 2.11(c). This study is interesting in the first place because it reveals the influence of the shift angles on the system conditioning. One of the first clear observations is that the two angles should never be on the diagonal. In that case, since $\Delta_1 = \Delta_2$, the corresponding points are the same $P_1 = P_2$ and so $\mathbf{L}_{\mathbf{p}_1} = \mathbf{L}_{\mathbf{p}_2}$, which is indeed a singular configuration for the system. There are multiple pairs of Δ_1, Δ_2 where the system conditioning is optimal. Also, the conditioning graph is symmetrical along this diagonal. *We point out that the default choice* $\Delta_1 = 0°$ *and* $\Delta_2 = 90°$ *is not the most optimal one as far as system conditioning is concerned.* This does not necessarily mean that the default choice is always a bad one in all cases. Considering rs_3 for example in Fig 2.11(c), we see that several choices of the shift angles can result in a satisfactory system conditioning.

These criteria have also been applied to the NPD configuration where the sensor and object plane orientations are not parallel to each other. The image observed is not symmetrical with respect to the coordinate axes, as observed from Figure 2.6(b). The results for this image are shown from Figures 2.12 to 2.14. A unique solution for Δ_1 exists for the visual features rs_1, rs_2 and rs_3 to achieve criterion (2.123). Corresponding to this solution, 4 solutions for Δ_2 exist that can satisfy the orthogonality condition. Next, let us observe the behaviour of the system conditioning for joint selection of Δ_1 and Δ_2. From Figures 2.12(c), 2.13(c) and 2.14(c). There are far fewer solutions for optimal conditioning when compared to the PD configuration seen previously. From the large blue areas in Fig 2.13(c), we also observe that for rs_2, there are more states (meaning shift angle pairs (Δ_1, Δ_2)) than in the case of rs_1 or rs_3 for which the system conditioning is not satisfactory.

Non-symmetrical case

For this case, we used images with a whale whose intensity distribution is evidently non-symmetrical. The objective of this case study was to show that the method can be applied without regards to the symmetrical nature of the target. Like for the symmetrical case, the selection criteria were applied to the visual features rs_1, rs_2 and rs_3 in PD and NPD configurations. This object as seen from the PD and NPD configurations are shown in Figs.2.7(a) and 2.7(b). The corresponding results are shown in Figures 2.15 to 2.17. For the individual selection of Δ_1 and Δ_2, a unique Δ_1 for the maximum responsiveness criterion and 4 corresponding solutions for Δ_2 for the orthogonality criterion are obtained. This is for all visual features rs_1, rs_2 and rs_3. From Fig 2.16(c) and 2.19(c), there are far more states of angles in the case of rs_2 where the conditioning might not be satisfactory. In fact, these are true also in the case of the non-symmetrical object being observed from a NPD configuration. Thus, results identical to the symmetrical case observed from a NPD

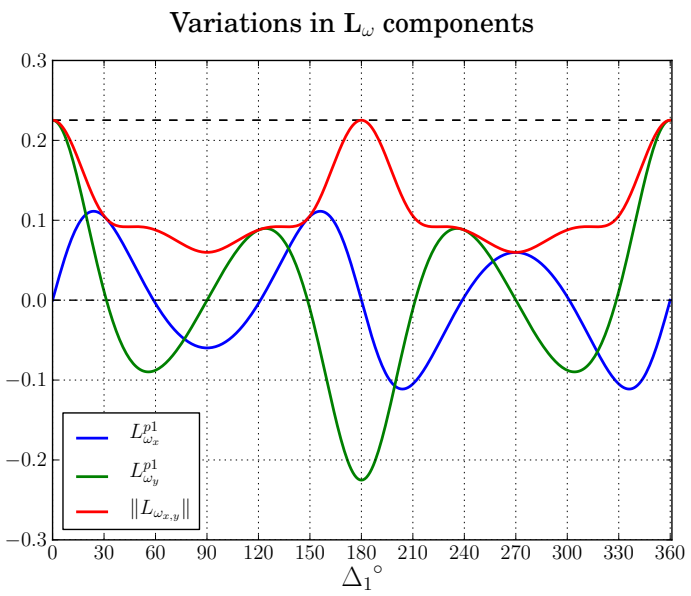


(a) In parallel desired configuration (b) In non-parallel desired configuration

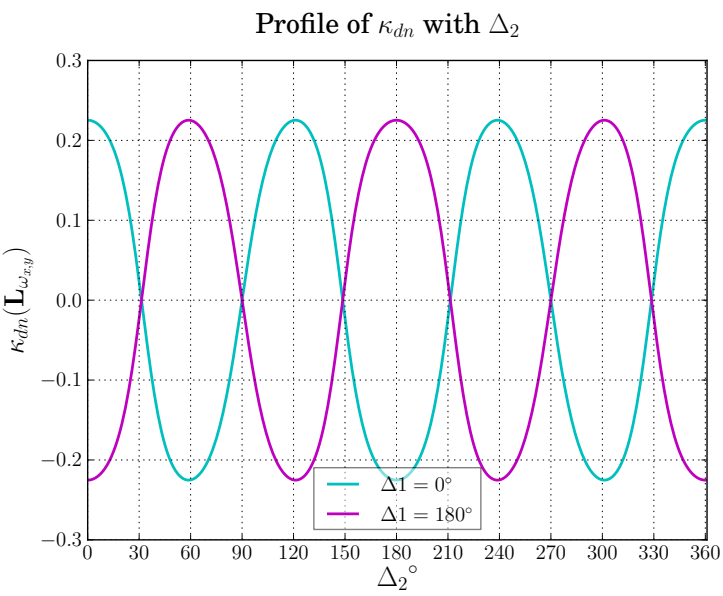
Figure 2.7 – Images acquired from non-symmetrical object

configuration are obtained. Let us note that these identical results are not surprising since the proposed criteria are all based on image measures. The camera images, whether of a symmetrical object observed from a NPD configuration or a non-symmetrical object observed from the PD or NPD configuration are identical and do not present any interesting symmetries along the coordinate axes.

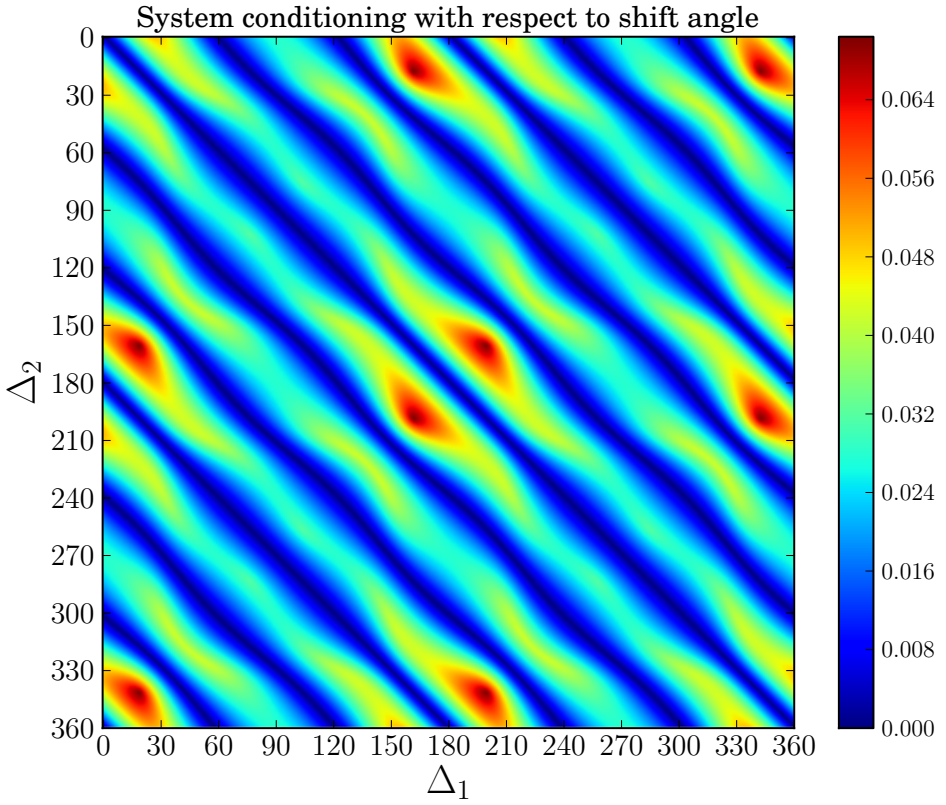
In the following section, we will showcase the usefulness of this selection by showing how this selection can be exploited to bring improvements to the visual servoing.



(a) Selection of Δ_1

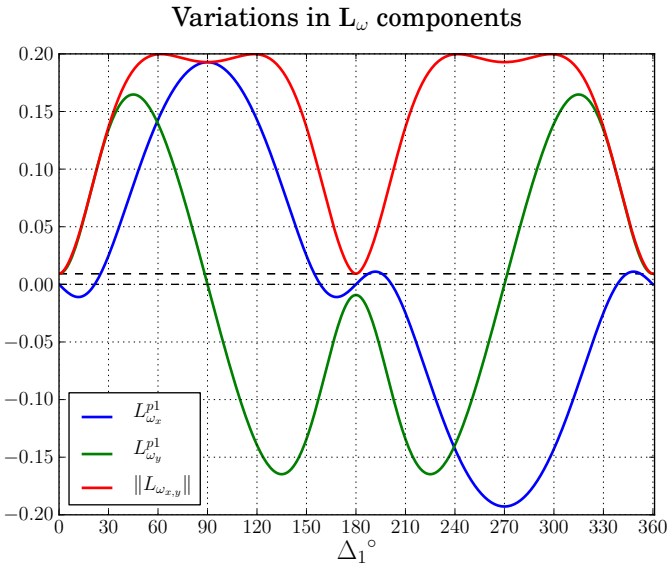


(b) Selection of Δ_2

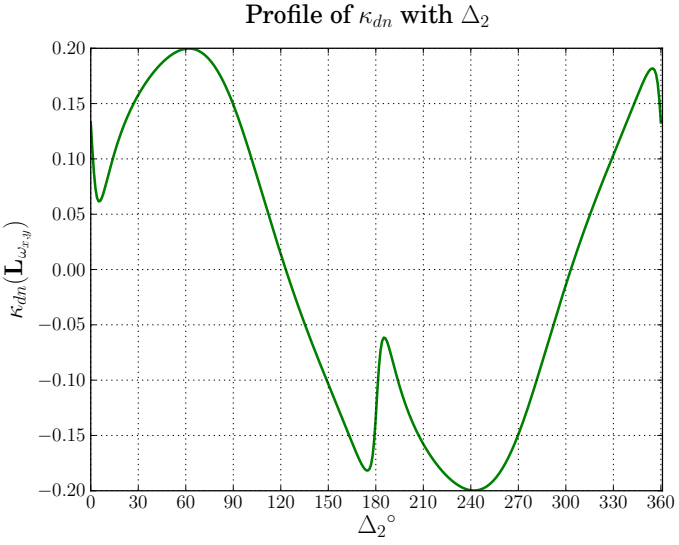


(c) Joint selection of Δ_1 and Δ_2

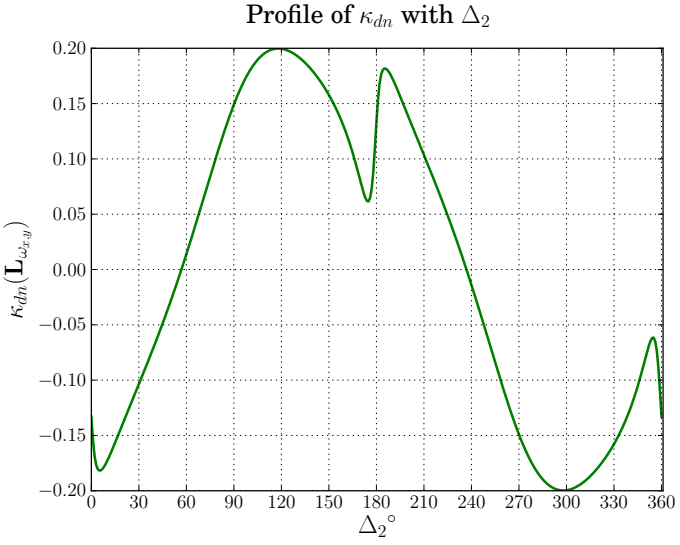
Figure 2.8 – Selection criteria applied to rs_1 with symmetrical object in PD configuration



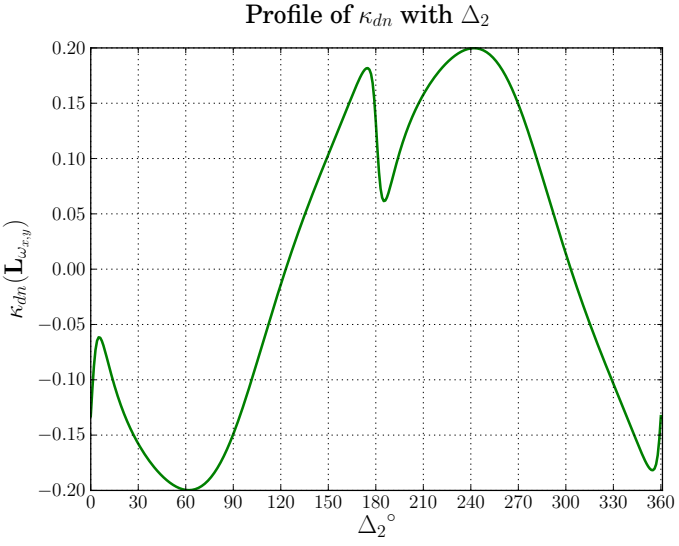
(a) Selection of Δ_1



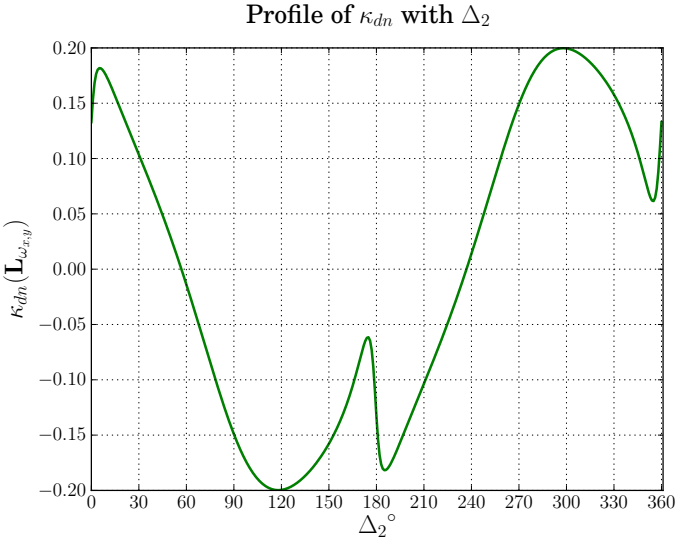
(b) Selection of Δ_2 with $\Delta_1 = 62°$



(c) Selection of Δ_2 with $\Delta_1 = 118°$



(d) Selection of Δ_2 with $\Delta_1 = 242°$



(e) Selection of Δ_2 with $\Delta_1 = 298°$

Figure 2.9 – Selection criteria applied to rs_2 with symmetrical object in PD configuration

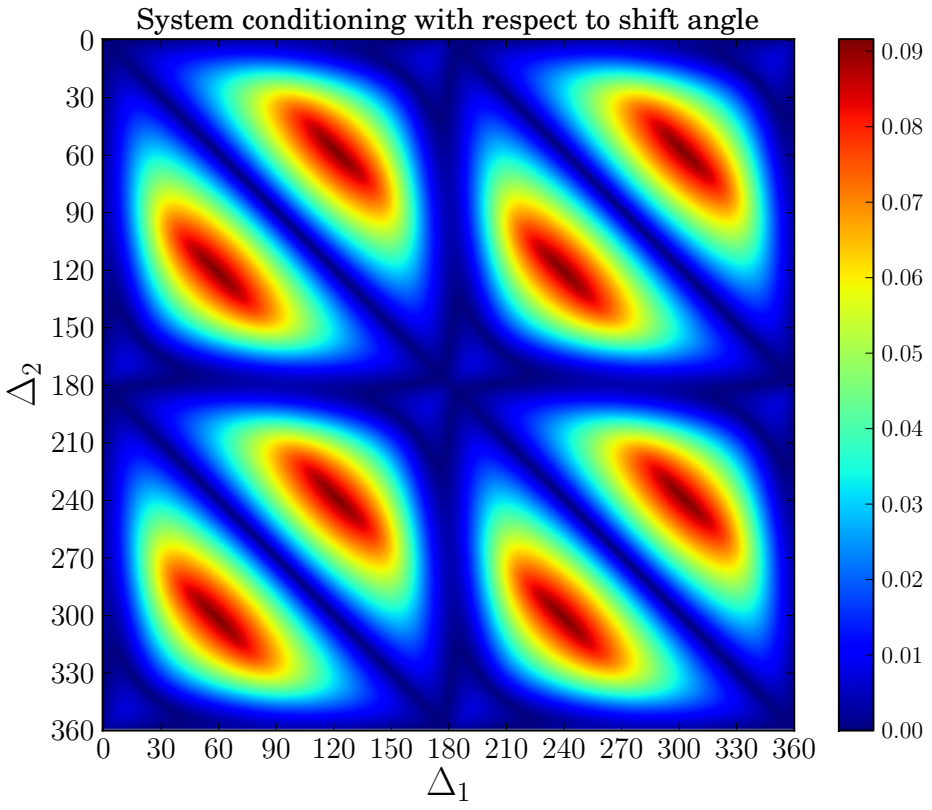
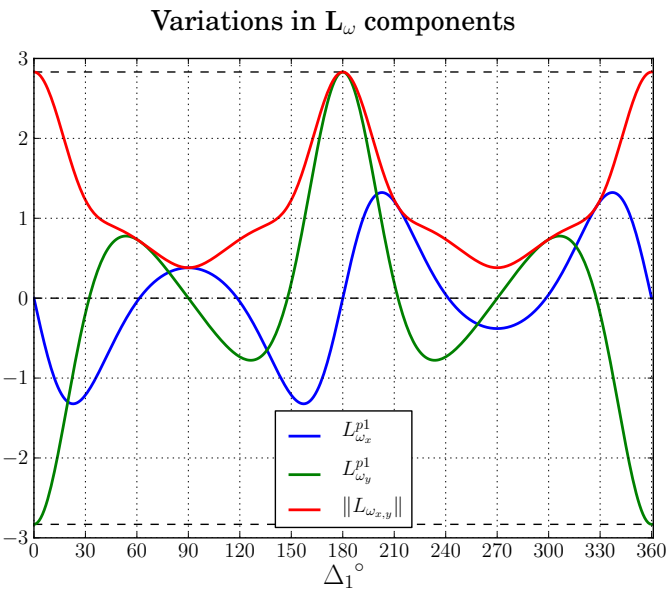
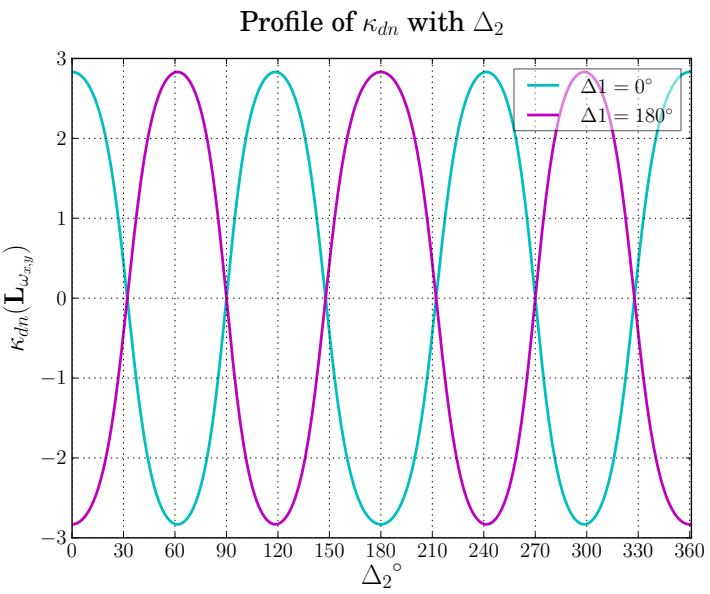


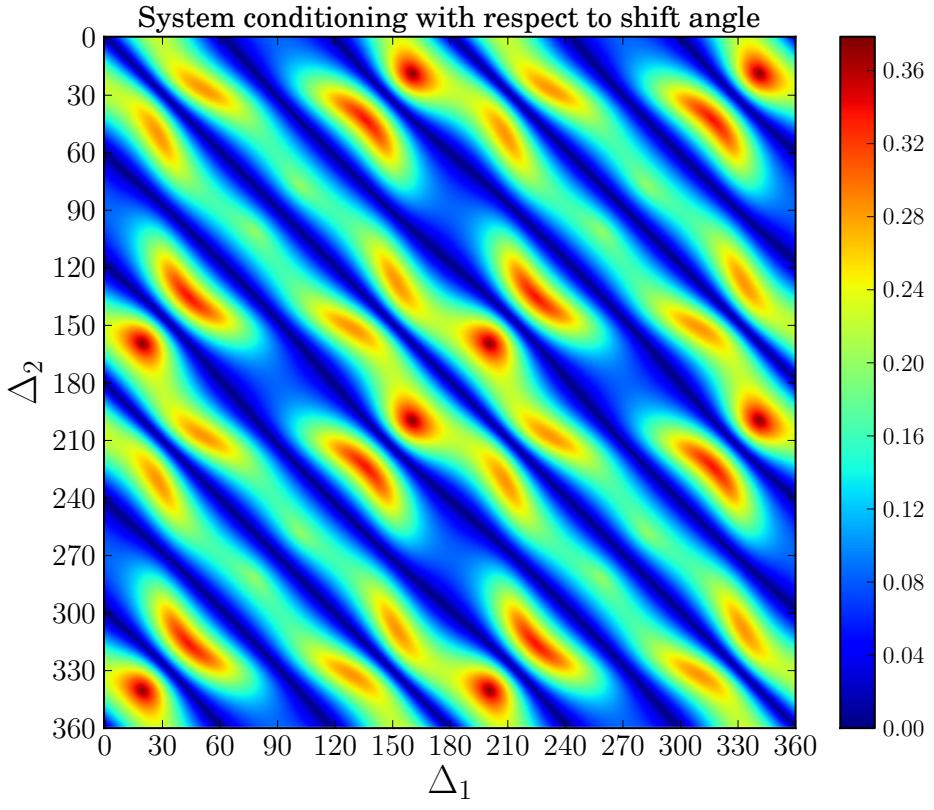
Figure 2.10 – Joint selection of Δ_1 and Δ_2 applied to rs_2 in PD configuration



(a) Selection of Δ_1

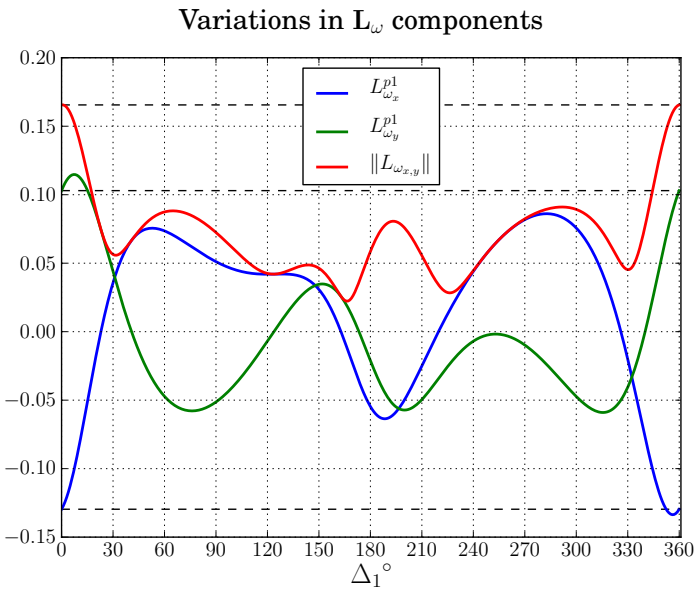


(b) Selection of Δ_2

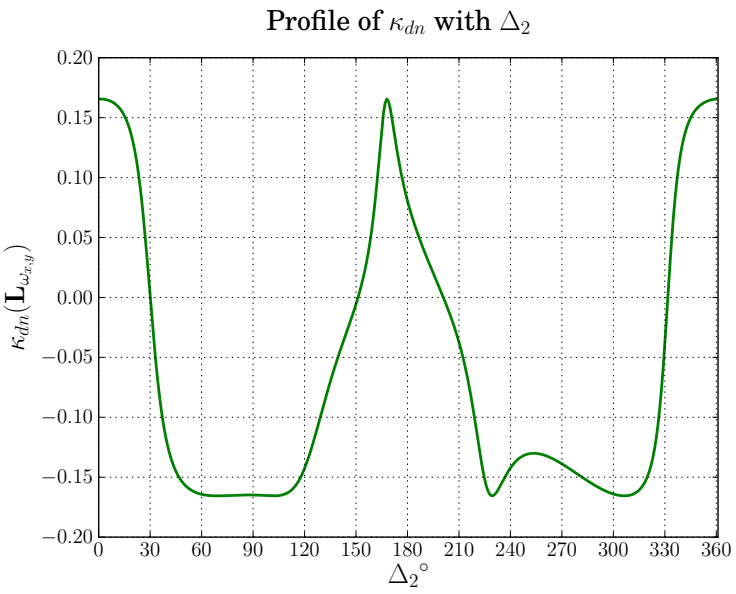


(c) Joint selection of Δ_1 and Δ_2

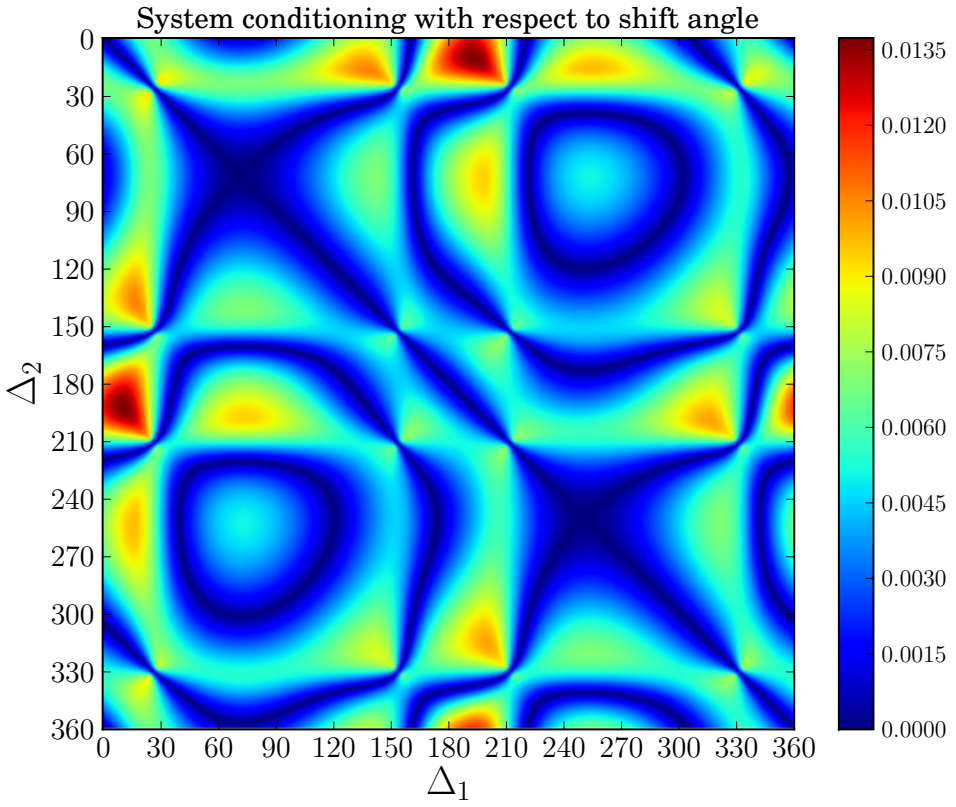
Figure 2.11 – Selection criteria to rs_3 with symmetrical object in PD configuration



(a) Selection of Δ_1

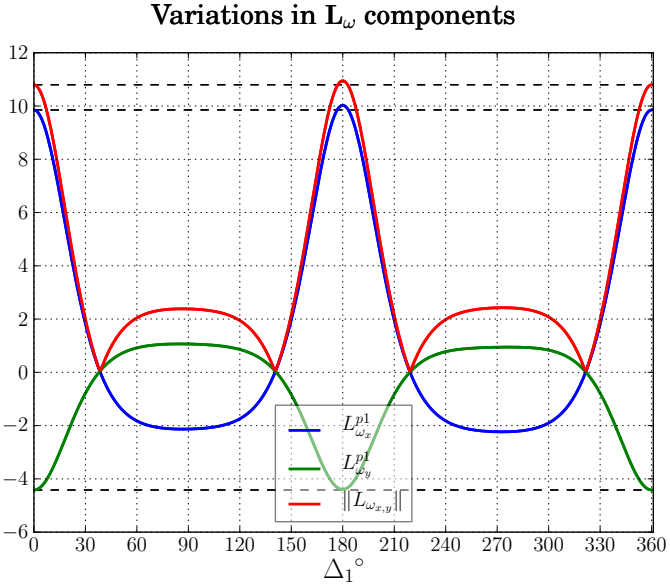


(b) Selection of Δ_2

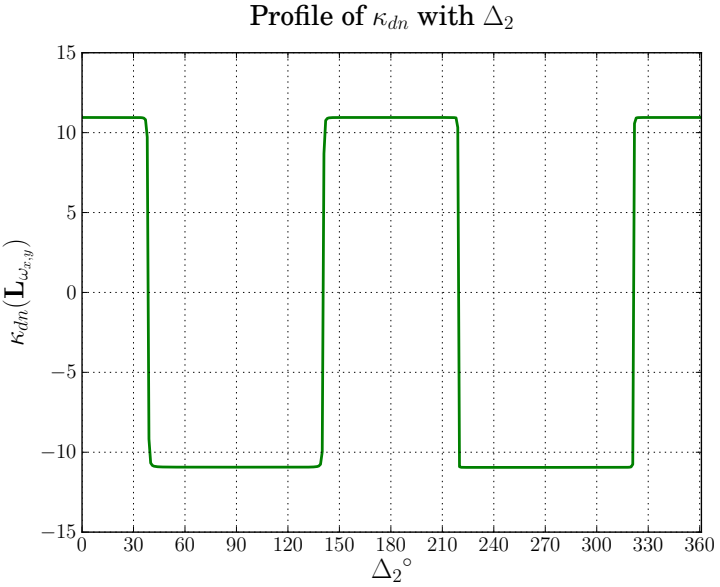


(c) Joint selection of Δ_1 and Δ_2

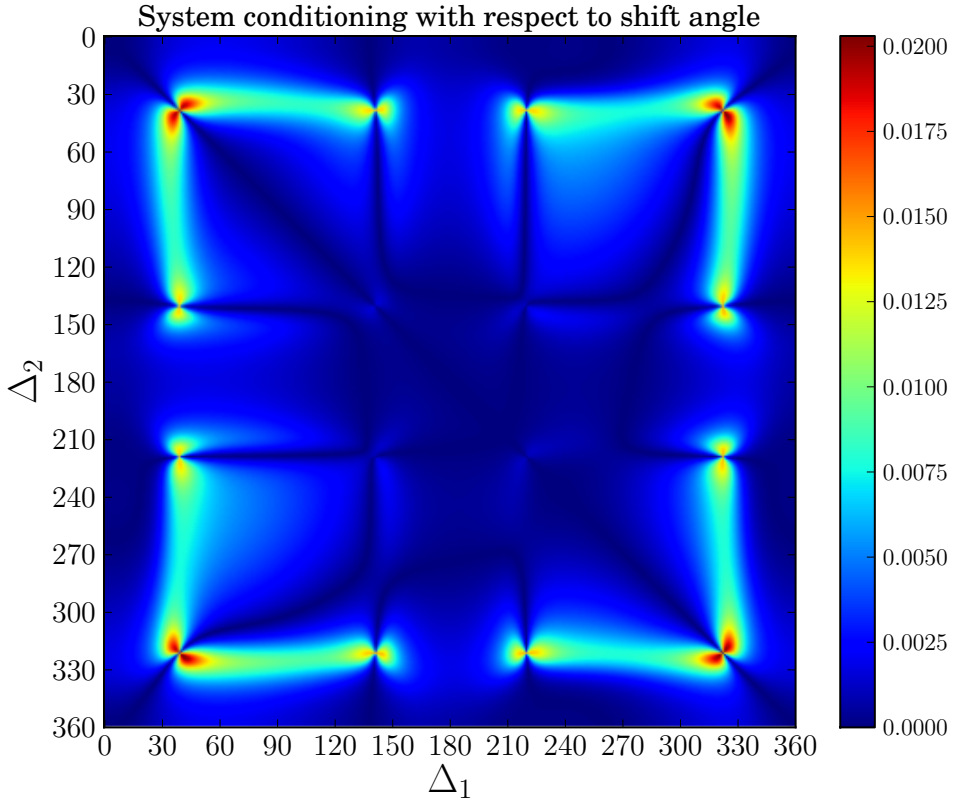
Figure 2.12 – Selection criteria applied to rs_1 with symmetrical object in NPD configuration



(a) Selection of Δ_1

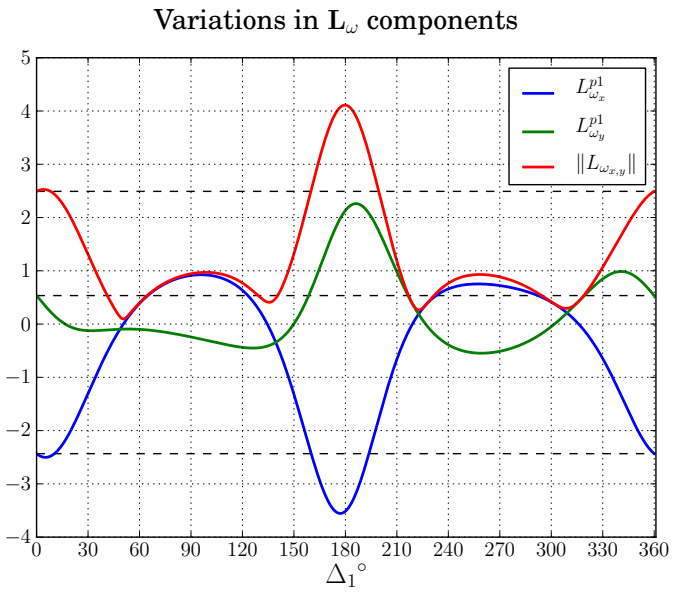


(b) Selection of Δ_2

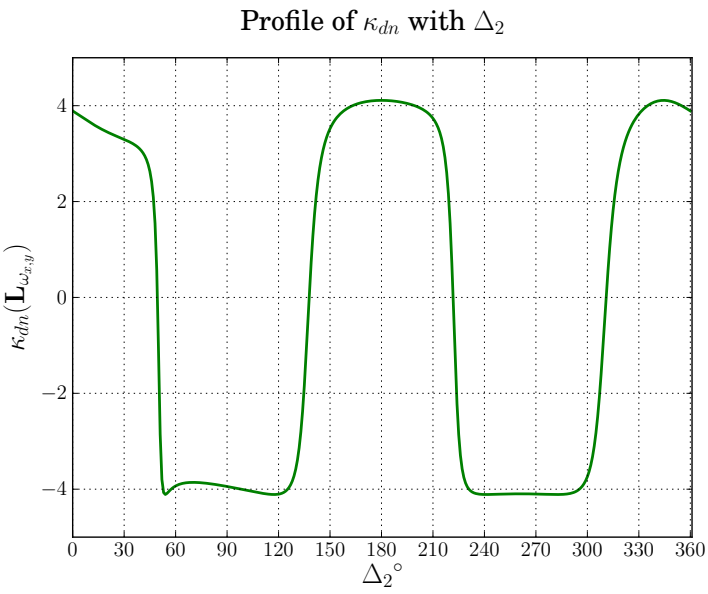


(c) Joint selection of Δ_1 and Δ_2

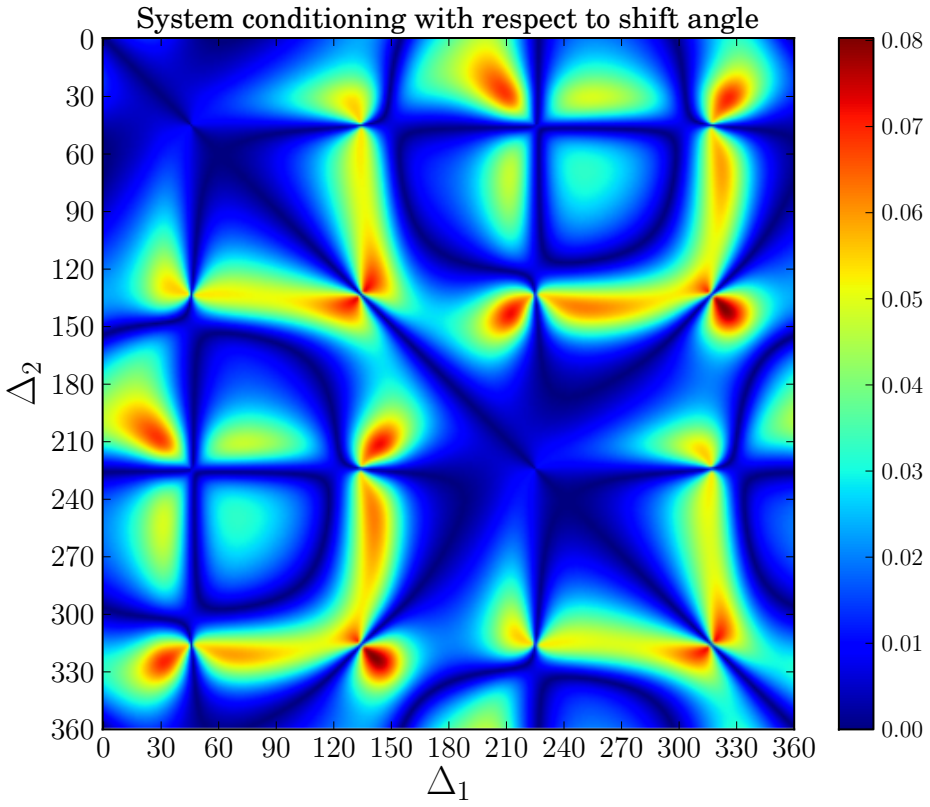
Figure 2.13 – Selection criteria applied to rs_2 with symmetrical object in NPD configuration



(a) Selection of Δ_1

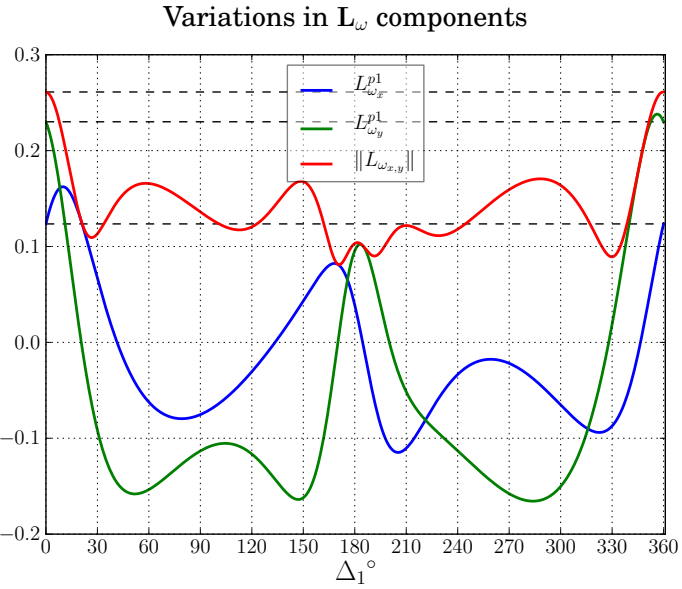


(b) Selection of Δ_2

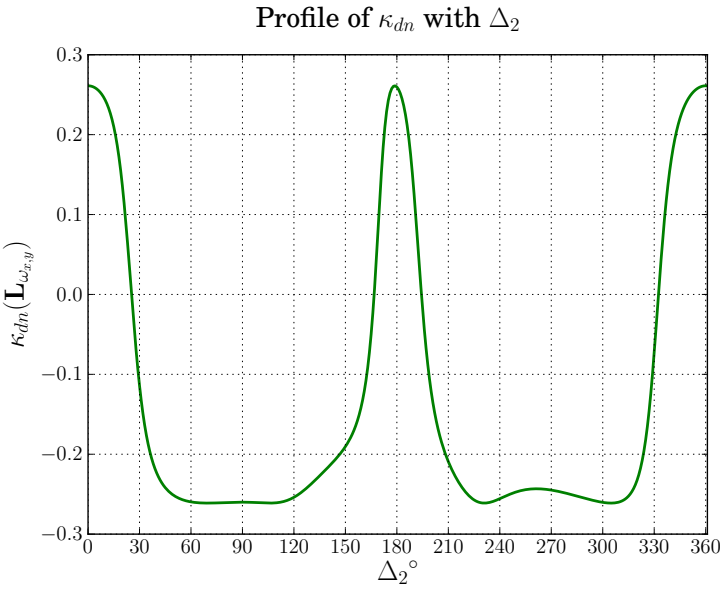


(c) Joint selection of Δ_1 and Δ_2

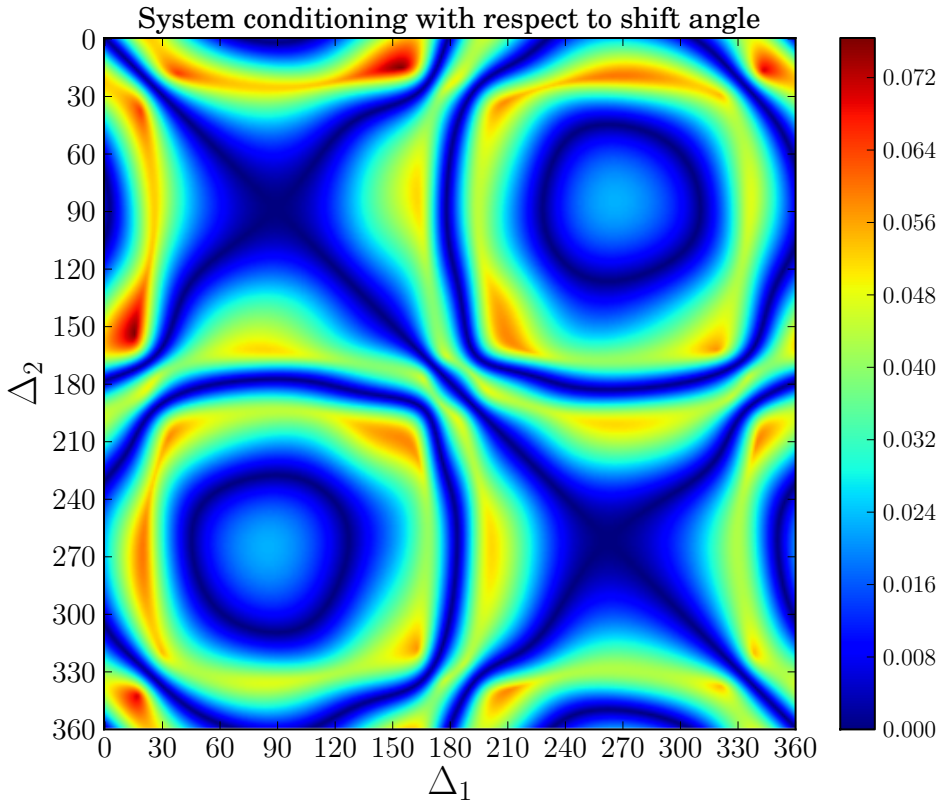
Figure 2.14 – Selection criteria applied to rs_3 with symmetrical object in NPD configuration



(a) Selection of Δ_1

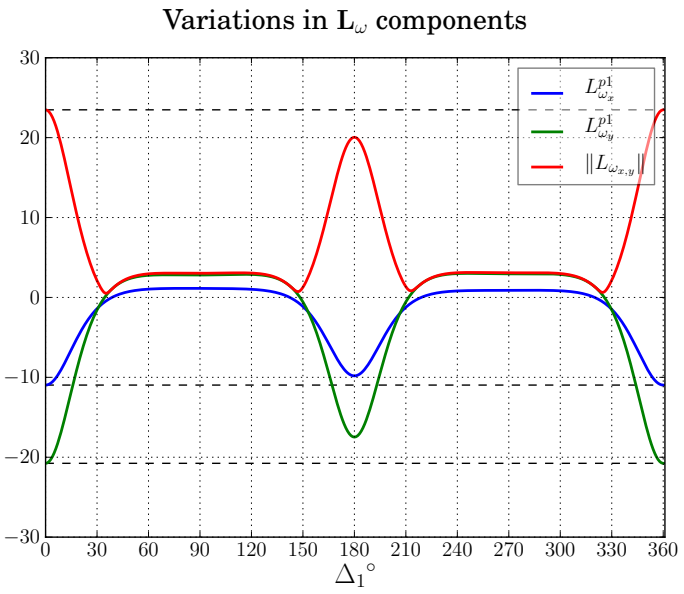


(b) Selection of Δ_2

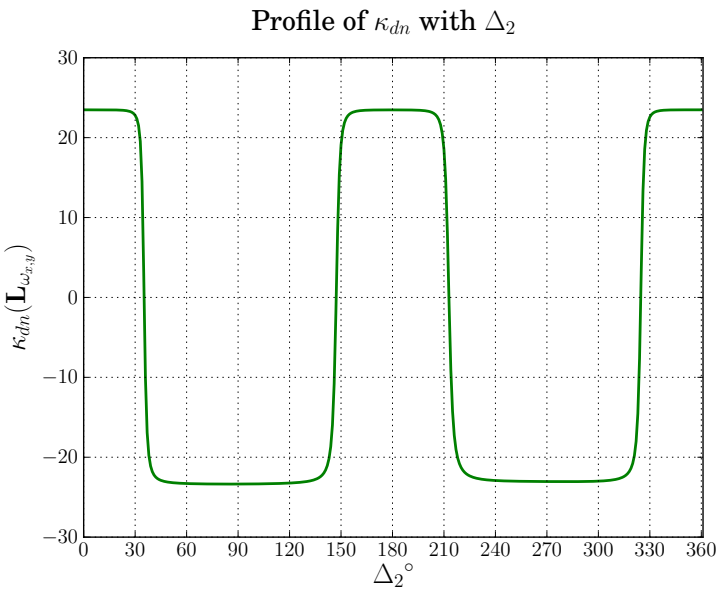


(c) Joint selection of Δ_1 and Δ_2

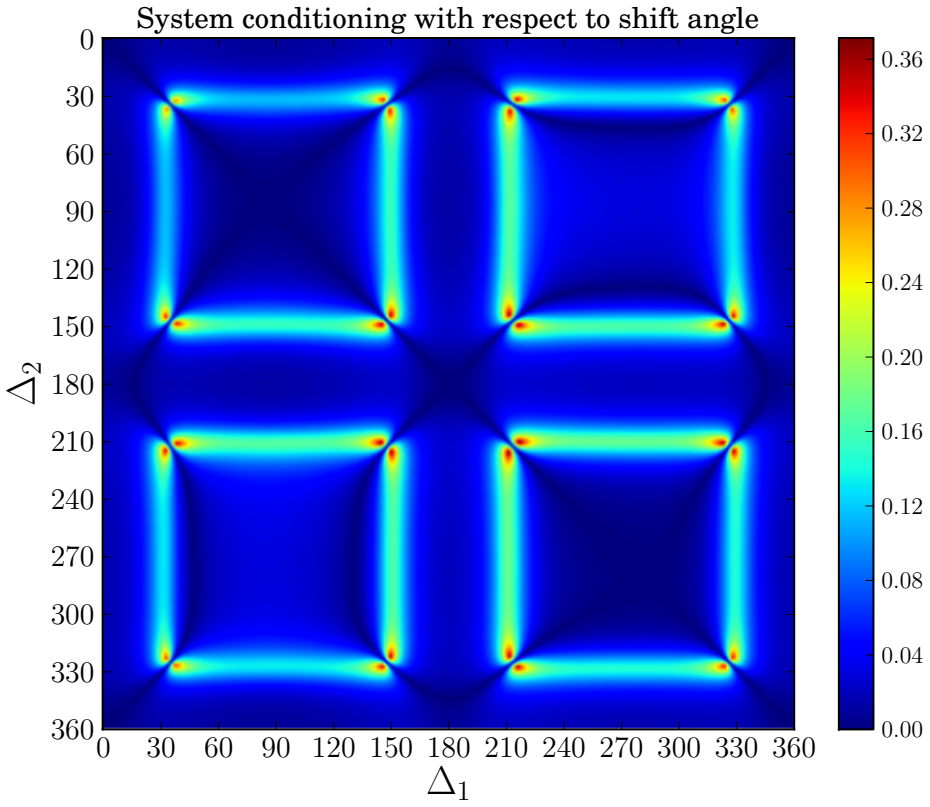
Figure 2.15 – Selection criteria applied to rs_1 with non-symmetrical object in PD configuration



(a) Selection of Δ_1

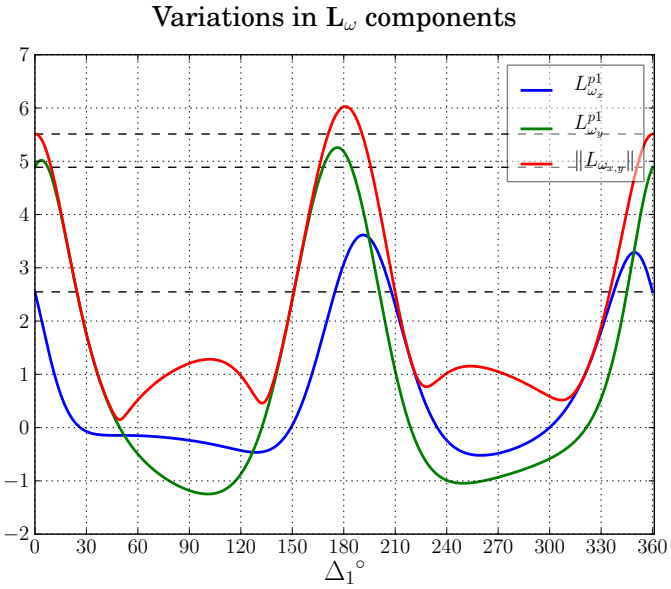


(b) Selection of Δ_2

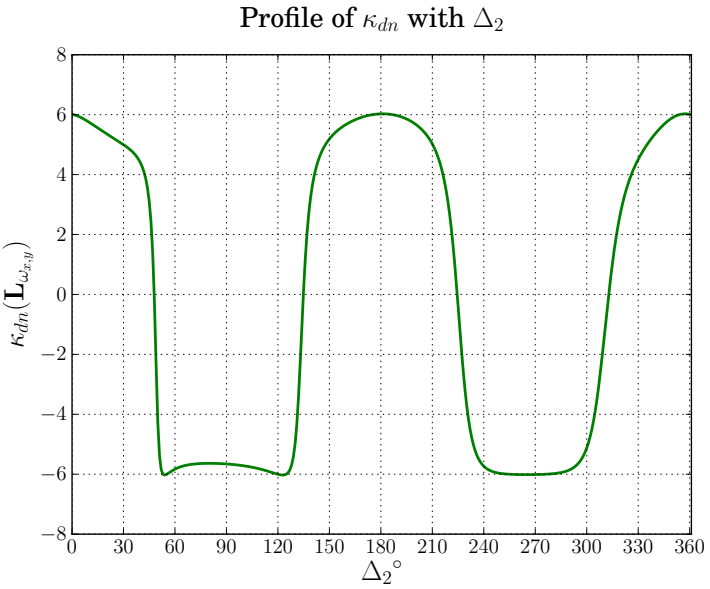


(c) Joint selection of Δ_1 and Δ_2

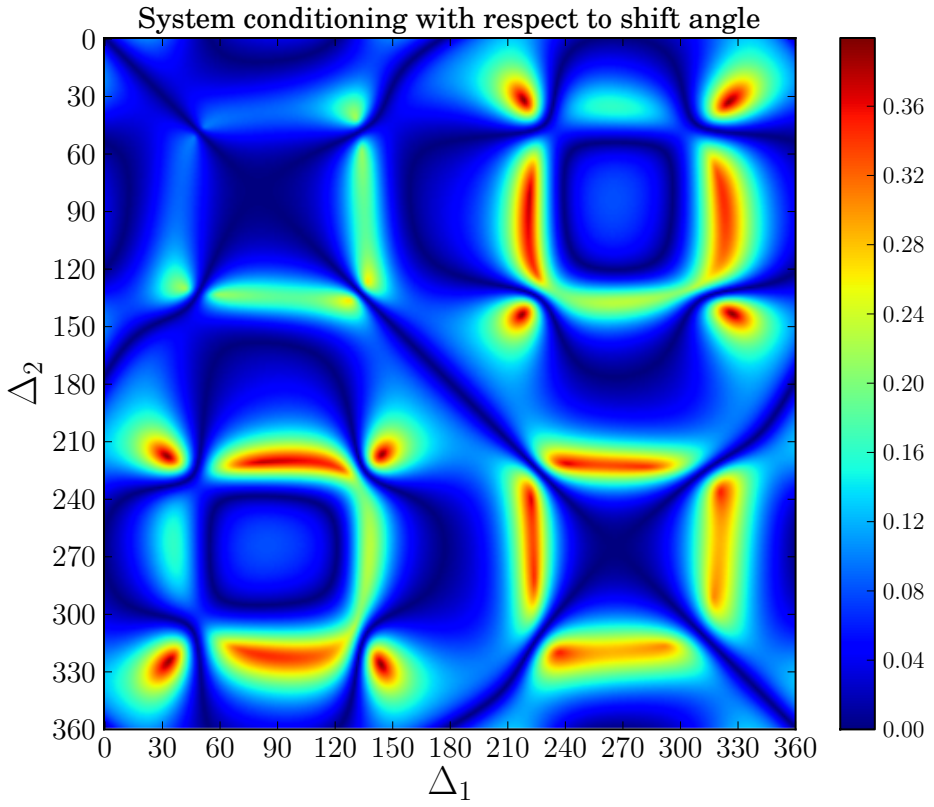
Figure 2.16 – Selection criteria applied to rs_2 with non-symmetrical object in PD configuration



(a) Selection of Δ_1

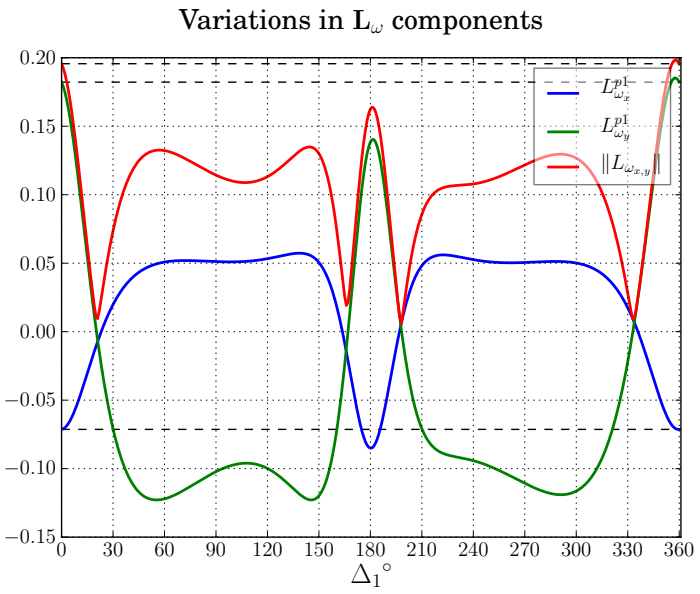


(b) Selection of Δ_2

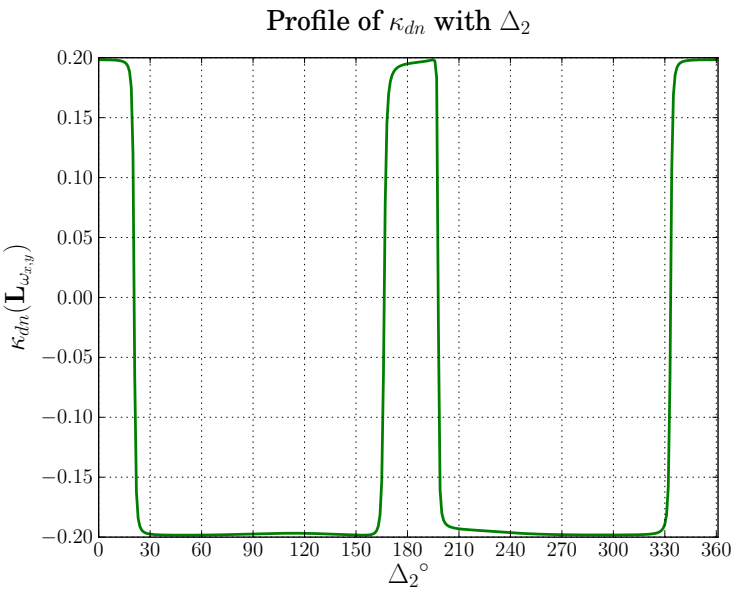


(c) Joint selection of Δ_1 and Δ_2

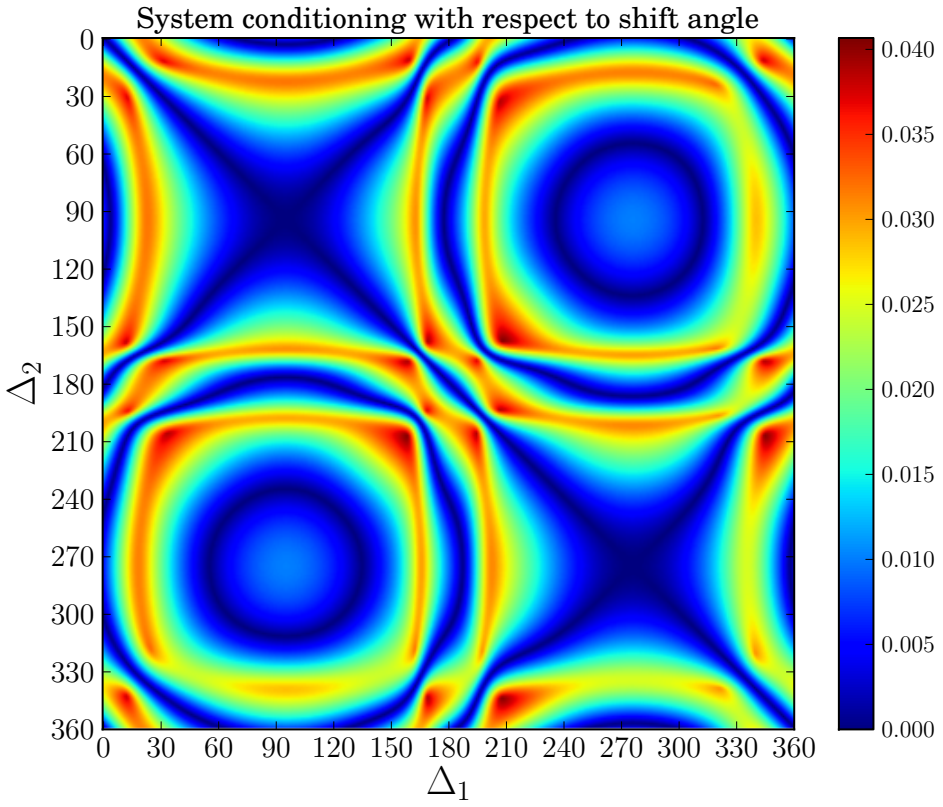
Figure 2.17 – Selection criteria applied to rs_3 with non-symmetrical object in PD configuration



(a) Selection of Δ_1

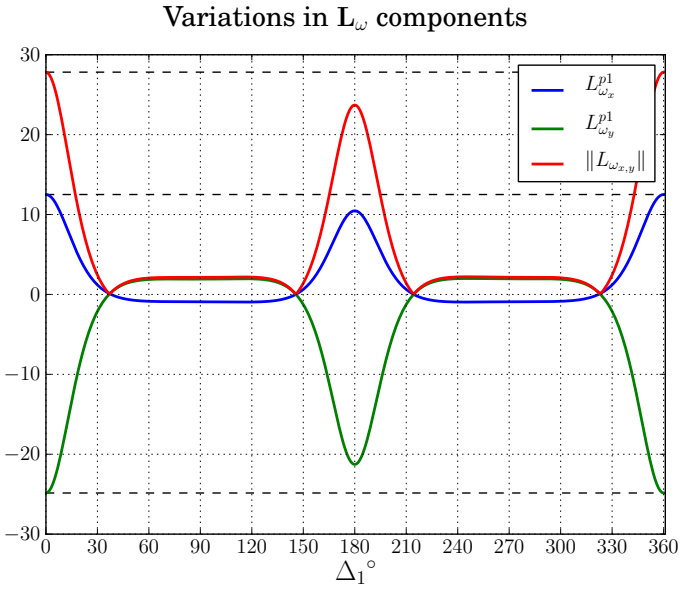


(b) Selection of Δ_2

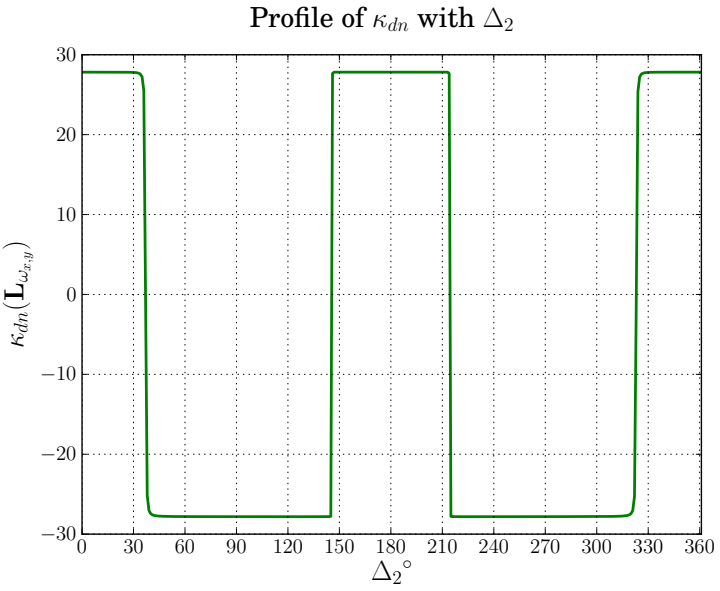


(c) Joint selection of Δ_1 and Δ_2

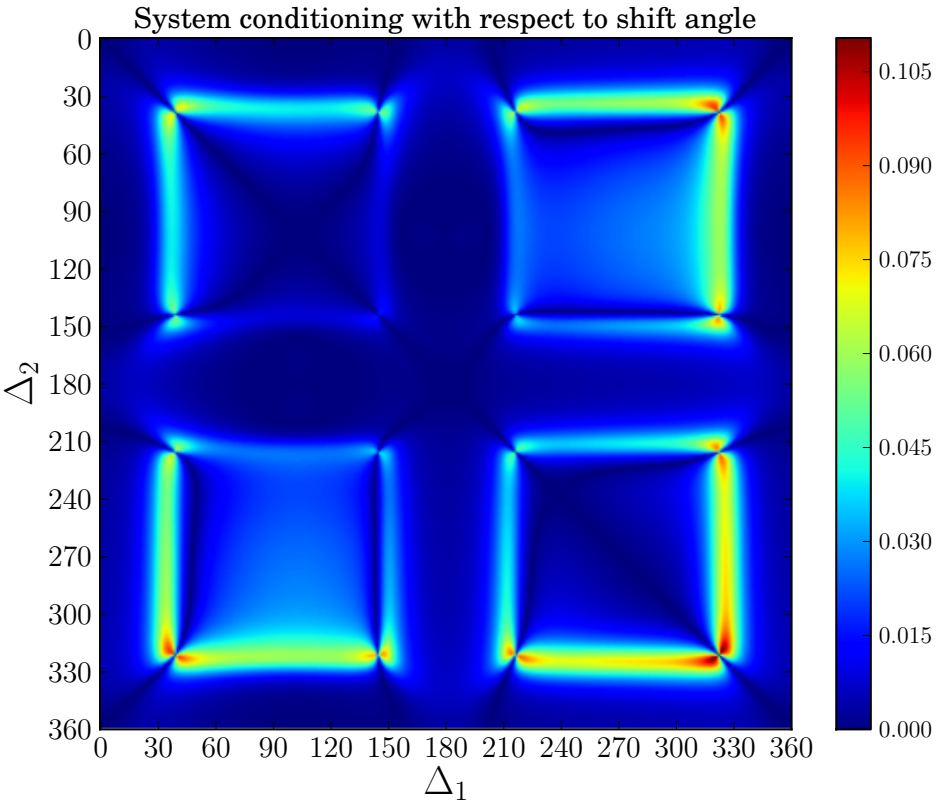
Figure 2.18 – Selection criteria for visual feature rs_1 with non-symmetrical object



(a) Selection of Δ_1

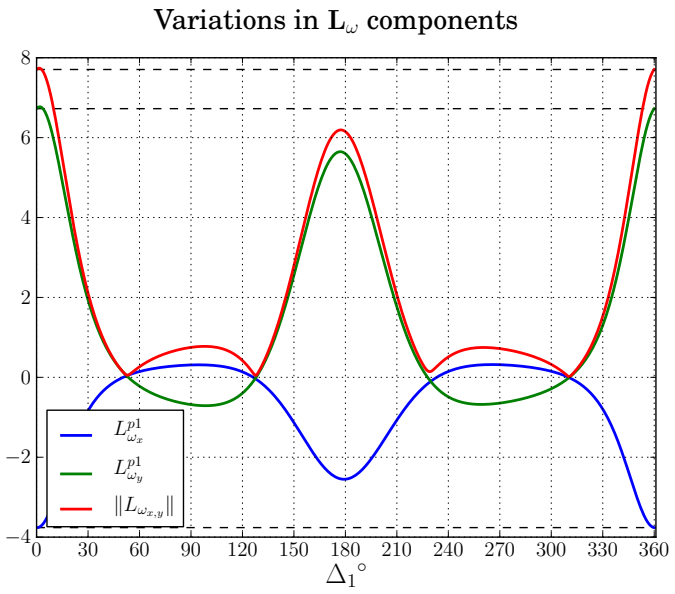


(b) Selection of Δ_2

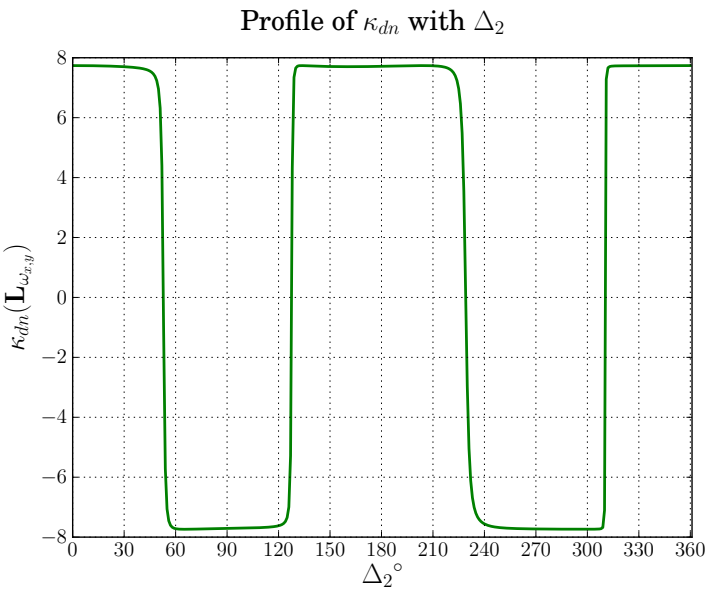


(c) Joint selection of Δ_1 and Δ_2

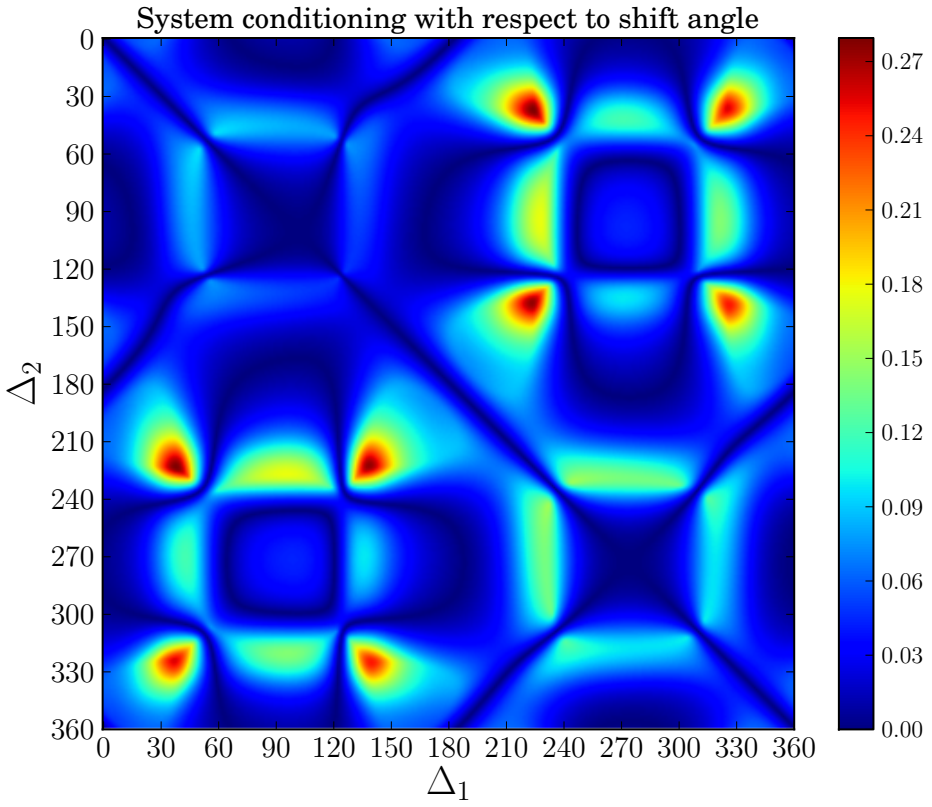
Figure 2.19 – Selection criteria for visual feature rs_2 with non-symmetrical object



(a) Selection of Δ_1



(b) Selection of Δ_2



(c) Joint selection of Δ_1 and Δ_2

Figure 2.20 – Selection criteria for visual feature rs_3 with non-symmetrical object

2.5 Visual Servoing with Tunable Visual Features

2.5.1 Methodology for 6DOF visual servoing experiments

For full 6DOF visual servoing, the following control law is used.

$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{-1}(\mathbf{s} - \mathbf{s}^*) \tag{2.129}$$

where $\mathbf{v}_c = (v_x, v_y, v_z, \omega_x, \omega_y, \omega_z)$ is the camera velocity screw. $\widehat{\mathbf{L}}_{\mathbf{s}}^{-1}$ is the inverse of the interaction matrix. In this part, the interaction matrix at the desired configuration was preferred with the depth approximated by its value at the desired pose $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \widehat{Z^*})$. With this simple choice, as explained in Chapter 1, a pose estimation would not be required. This estimation indeed is not convenient in practice in all scenarios, especially for dense methods. Many works including the pure photometric visual servoing [Collewet 11] use this configuration. Further, problems such as isolated singularities can be circumvented and focus can be laid on demonstration of the intended concepts. The work in [Michel 93] studied singular configurations that might occur when a set of 3 image points are used as visual features. For the case of image moments, to the best of our knowledge, this is an open problem, very complex and important at the same time. $\mathbf{s}$ is the following set of six visual features:

$$\mathbf{s} = (x_n, y_n, a_n, \phi_{p1}, \phi_{p2}, \alpha) \tag{2.130}$$

where $x_n = x_g a_n$, $y_n = y_g a_n$, $a_n = z^*\sqrt{a^*/a}$, $a = m_{00}$ being the area. $x_g = m_{10}/m_{00}$ and $y_g = m_{01}/m_{00}$ are the centre of gravity along the coordinate axes. These were introduced in Section 2.3.1.3. α is the orientation feature introduced in Section 2.3.2.2. ϕ_{p1} and ϕ_{p2} denote the visual features used to control the x and y rotational degrees of freedom. These features are computed with respect to two different shift points p_1 and p_2. We recall that ϕ_1 and ϕ_2 are based on TRS invariants computed from the shifted moments and three possible choices were shown in Section 2.4.1.1. Which of these choices is selected will be reported systematically in the listed experiments. This methodology will be referred to in the visual servoing experiments presented in the subsequent chapters.

Numerous simulations were performed with combinations of the visual features rs_1, rs_2 and rs_3 and using both symmetric and non-symmetric shaped objects. In this section, we showcase results that further support our assertion that existing choices made for the shifted moments in [Tamtsia 13b] are not always the best choice for every case. These choices sometimes lead to suboptimal spatial trajectories and in the worst case a divergence of the control law. In these cases, the selection procedures introduced in the previous section can be used to tune the visual features and this can help to mitigate the aforesaid issues. This is the motivation behind the experiments to be presented next. We recall that the selection criteria are applied on the image obtained from the desired pose.

2.5.2 Experiment I

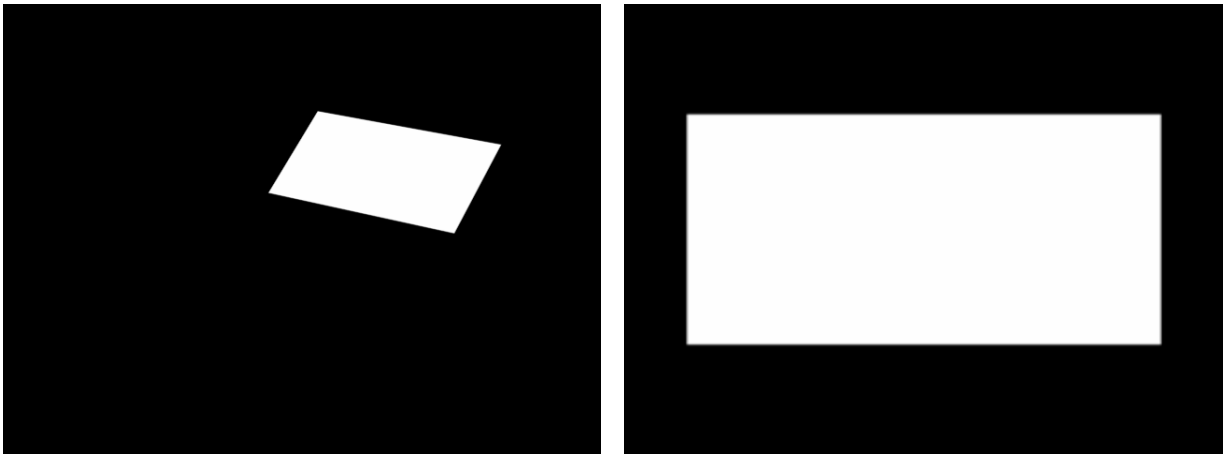
First, a simple experiment done with the symmetric object is presented. The desired pose is ${}^{c*}\mathbf{M}_o = [0.0, 0.0, 1.2m, 0.0°, 0.0°, 0.0°]$ from which the image in Figure 2.21(b) is learnt. The initial pose is chosen at ${}^{c}\mathbf{M}_o = [20.0cm, -20.0cm, 280cm, -20.0°, 20.0°, 20.0°]$. The initial image is shown in 2.21(a). When the visual features rs_1 was used with the de-

fault shift angles of $\Delta_1 = 0°$ and $\Delta_1 = 90°$ presented in [Tamtsia 13b], the visual servo began to diverge after a few iterations and this is shown in Fig 2.22(a). The final difference image is shown in Fig 2.22(e). Then, the visual feature rs_2 was used with the default choice and a failure again resulted. The system is driven away from the desired pose causing the object to leave the field of view in very few iterations as seen from 2.22(e). The resulting oscillatory behaviour is shown in Figure 2.22(c).

Using joint selection : Then, the same visual feature but this time using shifted moments from our joint selection method was used. As we pointed out earlier, there are numerous possibilities offered by the proposed selection methods. Here, a joint selection of Δ_1 and Δ_2 based on the system conditioning from Section 2.4.3.3 was used (see 2.128). For rs_1, the values $\Delta_1 = 161°$ and $\Delta_2 = 199°$ (shown in Figure 2.8(c)) was used. The visual servoing then converged for the same task as shown in Fig.2.22(b). Similarly for rs_2, the angles $\Delta_1 = 121°$ and $\Delta_2 = 239°$ were obtained using the joint selection method (See Fig 2.10). After these selections, the control law resulted in a satisfactory convergence of the errors as shown in Fig 2.22(d). The final difference image shown in Fig 2.22(g) confirms the errors were regulated to 0.

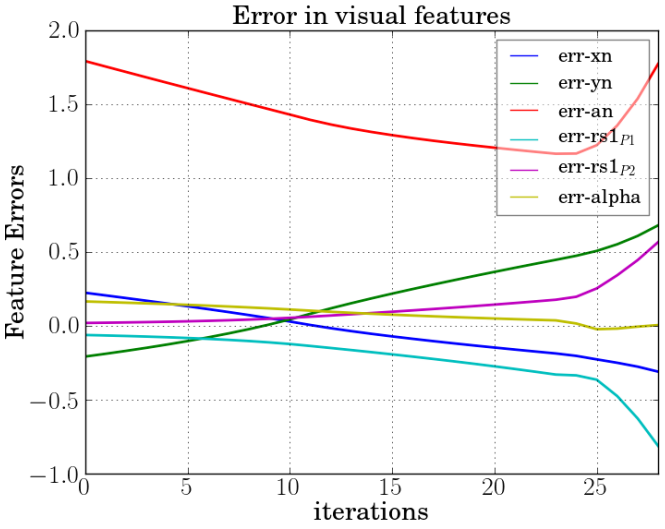
Selecting for orthogonality : Next, we would like also to demonstrate that this selection need not always be confined to the joint selection method. For example, let us consider the same experiment and the same visual feature rs_2 from shifted moments, but selected this time using criteria (2.123) and (2.127). Evaluation of these criteria with our desired image presents us the choices shown in Figures 2.9(a) and 2.9(d). From this, let us choose for instance $\Delta_1 = 242°$ and $\Delta_2 = 123°$. This choice yielded the following interaction matrix:

$$\begin{bmatrix} -1 & 0 & 0 & 0.0000 & -1.2713 & 0 \\ 0 & -1 & 0 & 1.2298 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.1243 & -0.1108 & 0 \\ 0 & 0 & 0 & 0.1107 & -0.1234 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1.0000 \end{bmatrix} \tag{2.131}$$

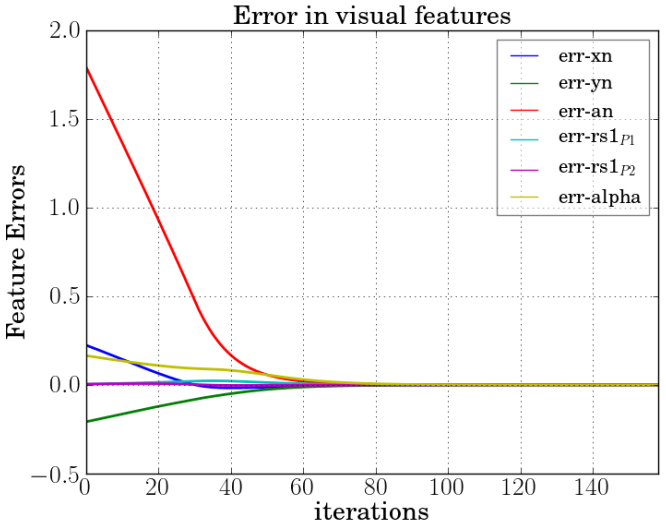


(a) Image learnt from initial pose (b) Image at desired pose

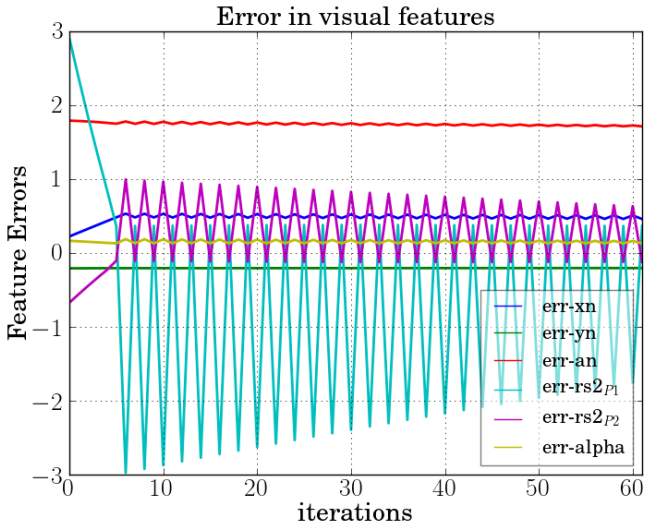
Figure 2.21 – Testing tunable visual features with symmetrical objects



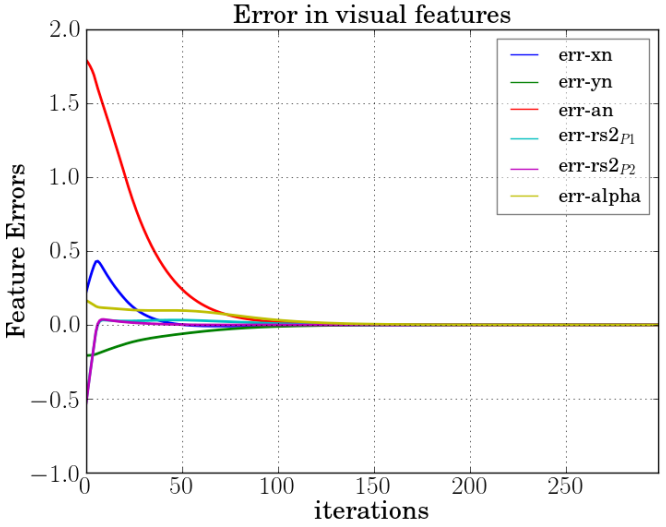
(a) rs_1 with default shifted moments



(b) rs_1 with joint selection



(c) rs_2 with default shifted moments



(d) rs_2 with joint selection

(e) Difference image using rs_1 from default shifted moments

(f) Difference image using rs_2 from default shifted moments

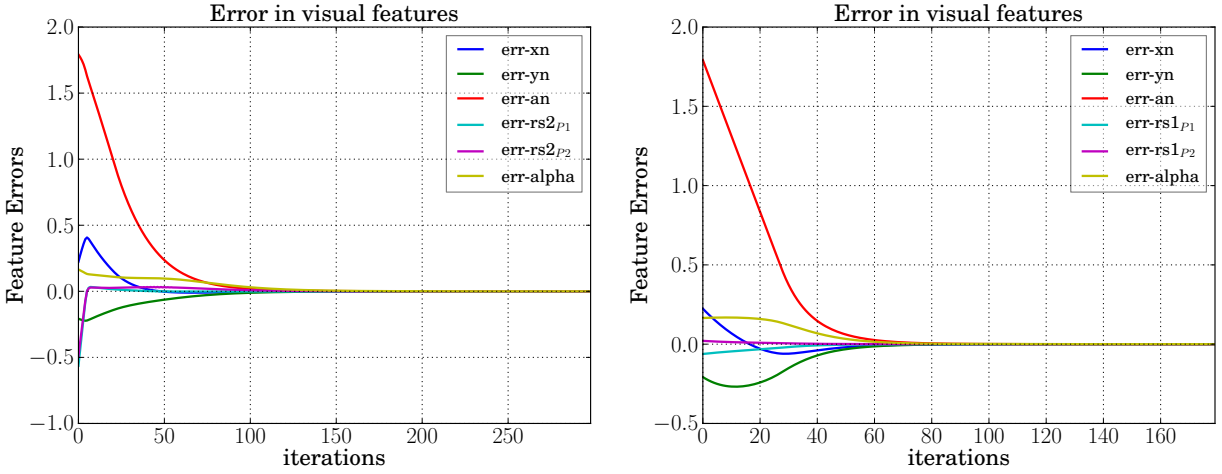
(g) Difference image using rs_1/rs_2 after joint selection

Figure 2.22 – Tunable visual features - Results for Experiment 2.5.2

The empirical dot product of the components in question is $-8.7290e^{-5}$. This validates our proposal about achieving orthogonality in the interaction matrix components related to the x and y rotations discussed in 2.4.3.2. Visual servoing results from this choice are shown in Figure 2.23(a). Similarly, for rs_1, with the shift angles of $270°$ and $0°$ obtained with criteria (2.127) and (2.123), the following excellent decoupled structure was obtained in the interaction matrix.

$$\begin{bmatrix} -1 & 0 & 0 & 0.0000 & -1.2713 & 0 \\ 0 & -1 & 0 & 1.2298 & -0.0000 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0499 & 0.0000 & 0 \\ 0 & 0 & 0 & 0 & 0.1877 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1.0000 \end{bmatrix} \tag{2.132}$$

The visual servoing converged with a satisfactory decrease of the errors as shown in Figure 2.23(b). So, we see clearly that the proposed criteria have enabled obtaining an attractive decoupled interaction matrix. The selection criteria proposed are of interest because they present such interesting possibilities in IBVS. We also note that we used the criteria (2.123) and (2.127) for the results in [Bakthavatchalam 14]. We argue that the default



(a) Feature errors for rs_2 using (2.131)

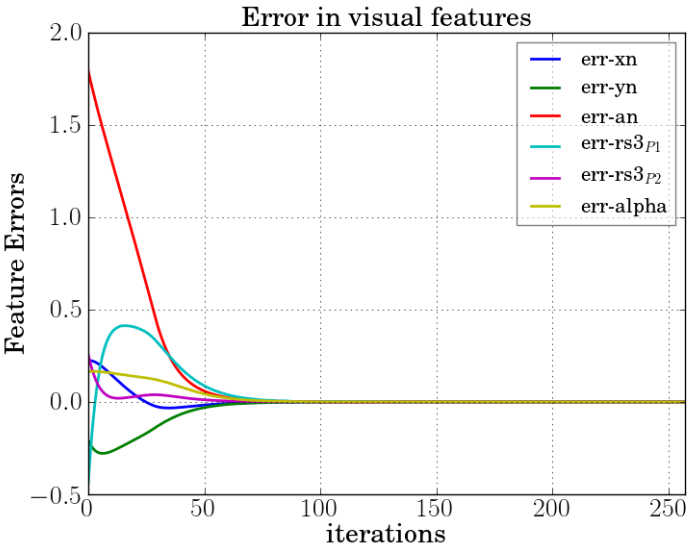
(b) Feature errors for rs_1 using using (2.132)

Figure 2.23 – Behaviour with rs_1 and rs_2 for Experiment 2.5.2 with orthogonality criterion

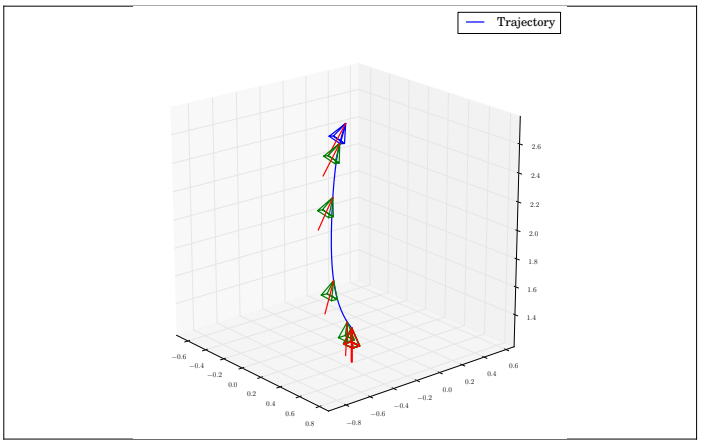
selection is not always the best choice. In this study for example, the servoing using two different visual features failed when the moments were computed using default values for the shift angles. However, thanks to the selection procedures introduced for the shift points in this thesis, it was possible to obtain a satisfactory behaviour for exactly the same conditions under which the default shifted moments failed. The selection of the visual features for these cases rescued the control from failure and ensured satisfactory convergence.

However, the default choice does not always cause a failure in the control law. For the same task, the visual feature rs_3 with the default shift angles $\Delta_1 = 0°$ and $\Delta_2 = 90°$ converged to the desired pose. The interaction matrix in this case had a condition number

of 13.78 and the results were satisfactory as shown in Figure 2.24. In this case, let us note that if required, the conditioning can be optimized even more using our strategy.



(a) Feature errors



(b) Camera velocities

Figure 2.24 – Behaviour with rs_3 for Experiment 2.5.2

In the next experiment, we present another case where the default choice does not result in a satisfactory Cartesian behaviour.

2.5.3 Experiment II

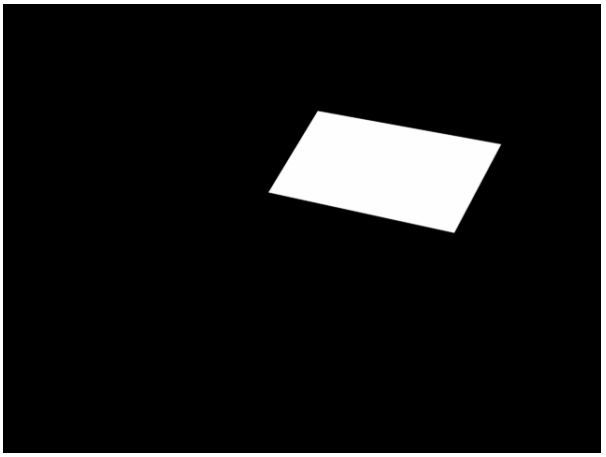
This experiment was conducted with the initial and desired poses from the previous Experiment 2.5.2 interchanged. The initial image is shown in Fig 2.25(a). The system should converge to the desired pose which is non-parallel to the object plane (NPD configuration). The image learnt from this pose is shown in 2.25(b). Different from the previous case, this experiment was performed with the visual feature rs_3 first with the default values for the shift angles. During the visual servo, part of the image briefly leaves the field

of view, introducing a perturbation in the profile of the visual feature errors as shown in Fig 2.25(c). However, the system attains the expected desired pose at the end of the servo. The deviant Cartesian trajectory is shown in blue in Figure 2.25(e). Later, a shift angle pair obtained using the joint selection criterion was used. For this experiment, we used $\Delta_1 = 141^\circ$ and $\Delta_2 = 322^\circ$ from among the options presented in Fig. 2.14(c)). After this selection, the same control law produced a better behaviour and the image did not leave the field of view. The satisfactory decrease in errors can be observed from 2.25(d) and the spatial trajectories are shown in green in 2.25(e).

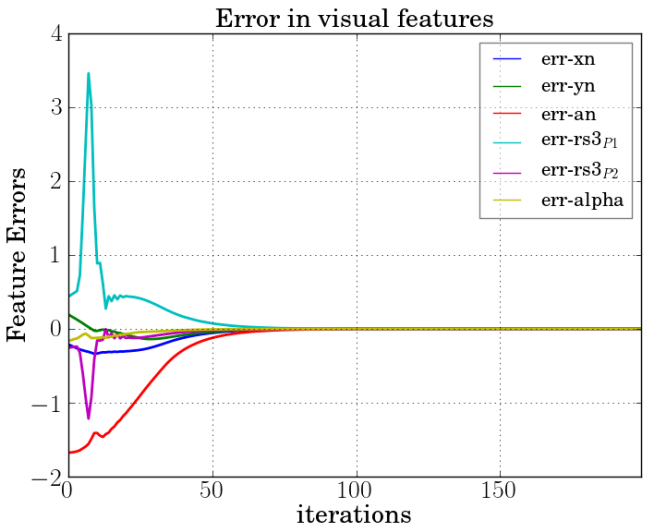
2.5.4 Experiment III

We report here another experiment conducted with a non-symmetrical object. The initial image is shown in Fig 2.26(a) and image learnt from the desired pose is shown in Fig 2.26(b). The visual feature rs_2 was employed in this case first with the default shifted moments. The decrease in one of the error components which is related to the first shift angle Δ_1 is very unsatisfactory. Similar to the previous two cases, we rerun the same task but now with the visual feature computed from the joint selection of the two shift angles. Specifically, from Figure 2.16(c), the pair $\Delta_1 = 144^\circ$ and $\Delta_2 = 211^\circ$ was chosen and the experiment repeated. From the results in Fig 2.26(d), we find that the decrease is much more satisfactory in this case. This is reflected also certainly in the excellent spatial trajectory shown in green in Fig 2.26(e).

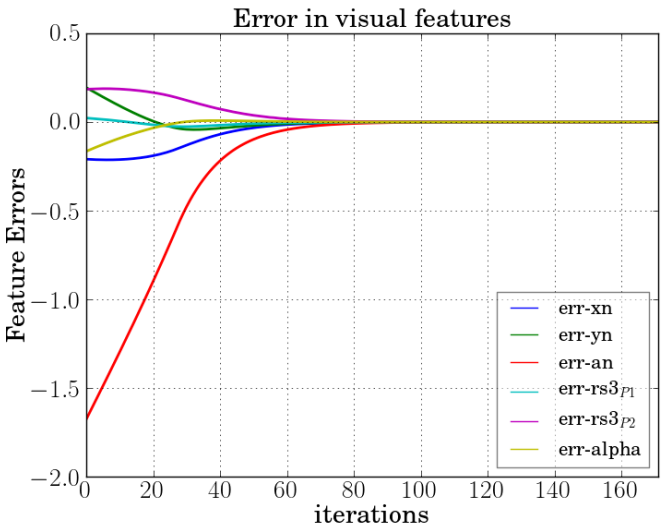
(a) Image at initial pose



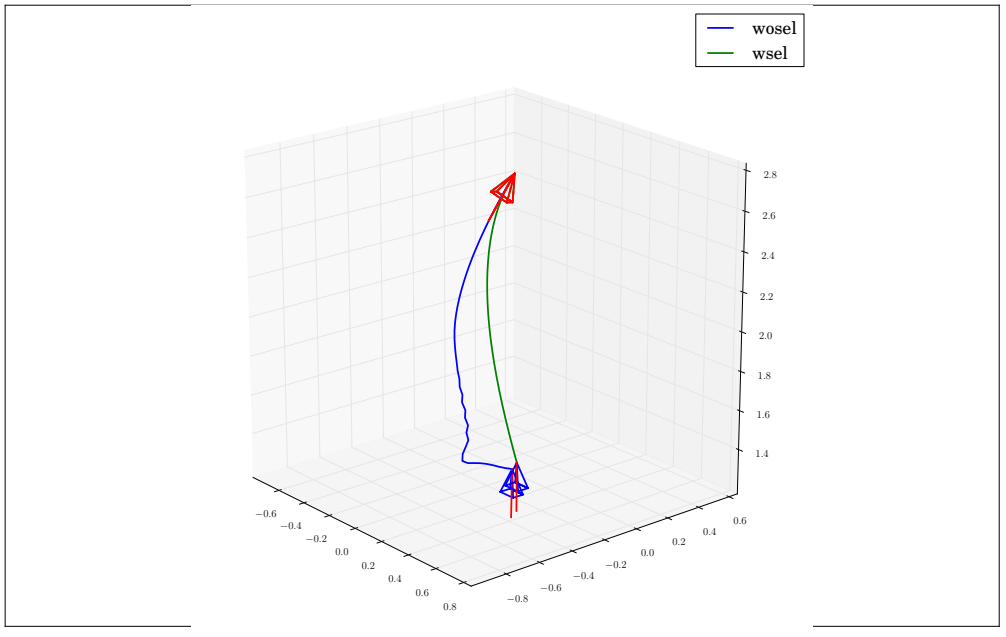
(b) Image learnt from desired pose



(c) rs_3 with default shifted moments

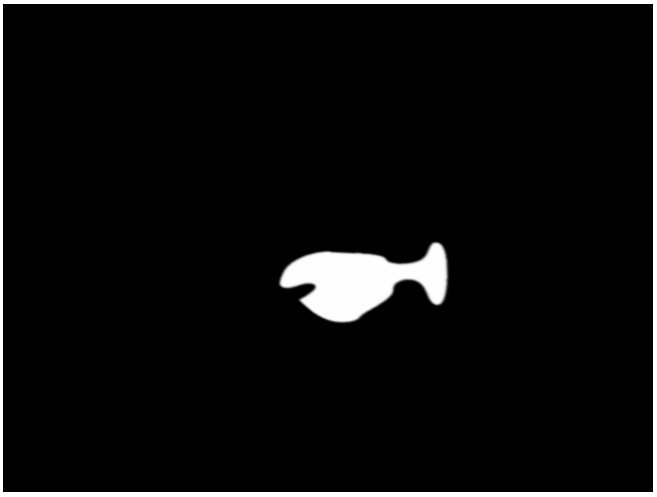


(d) rs_3 with selection



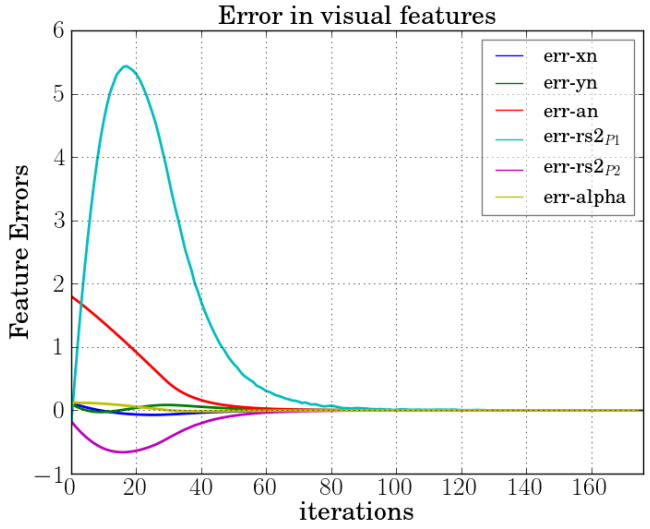
(e) Trajectory comparison with default shifted moments and with joint selection

Figure 2.25 – Tunable visual features - Results for Experiment 2.5.3

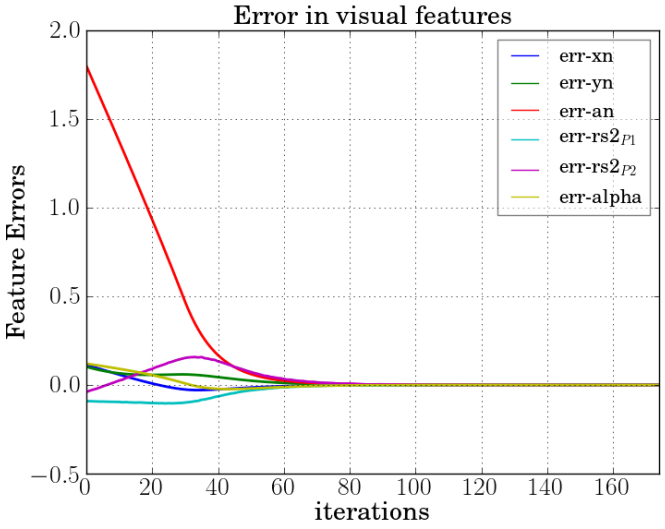


(a) Image at initial pose

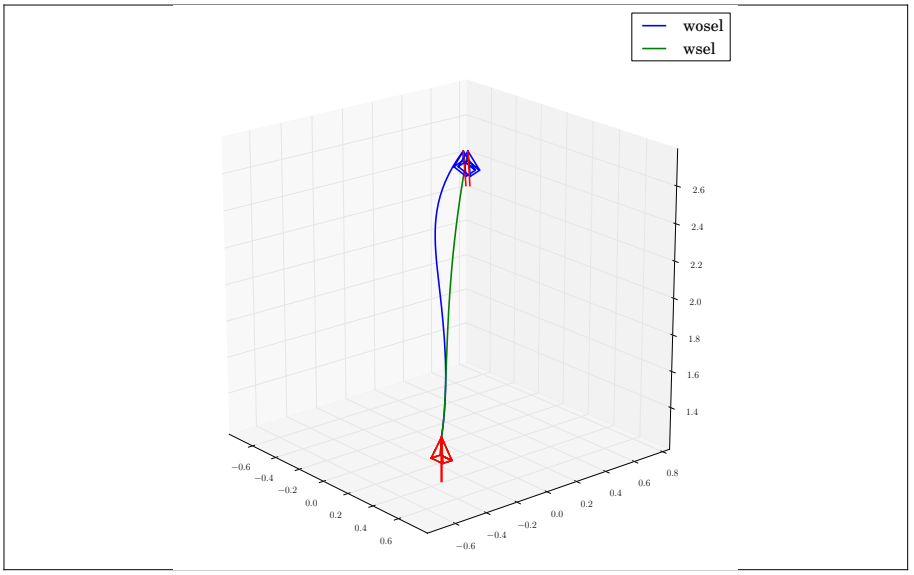
(b) Image learnt from desired pose



(c) rs_2 with default shifted moments



(d) rs_2 with selection



(e) Trajectories comparison

Figure 2.26 – Tunable visual features - Results for Experiment 2.5.4

2.5.5 On the choice of a moment-invariant based visual feature

This section is concerned with the choice of visual features that are mainly related to the control of rotational motions around the x and y axes in visual servoing.

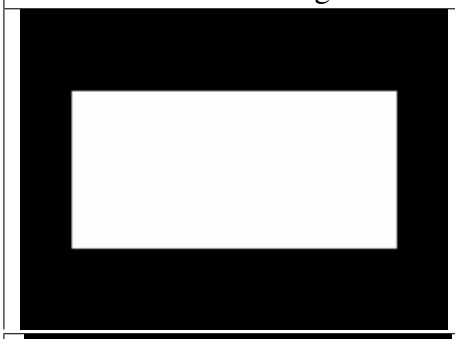
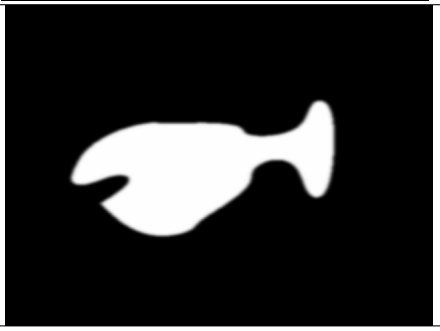
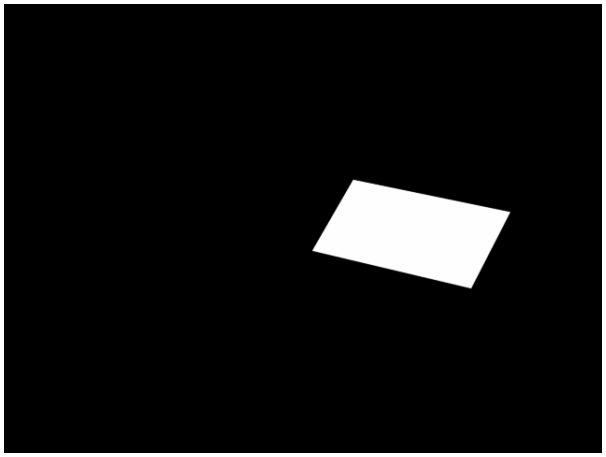
Camera Image	rs_1	rs_2	rs_3
	291.44	125.49	29.04
	56.34	373.3	14.407

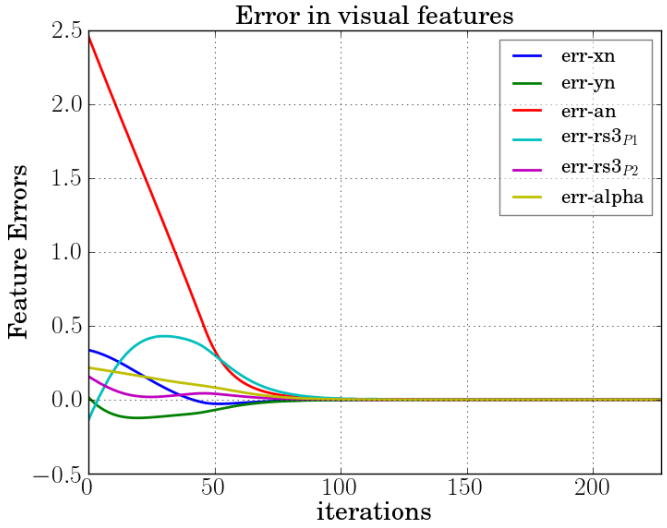
Table 2.1 – Condition Numbers for rs_1, rs_2 and rs_3

In general, it is hard to single out a specific visual feature as best for all for servoing tasks, given that this depends on the type of scene observed by the camera and the several choices available [Tahri 05a] [Tahri 04]. From Section 2.4.4, it is clear that various schemes can be used for the selection of Δ_1 and Δ_2. The design space is large. In addition, the moment invariants-based visual features proposed in existing literature, for example in [Tahri 05a], can be computed using the generalized shifted moments proposed in Section 2.4.2. This leaves us with a multitude of choices with interesting possibilities but an exhaustive exploration of all these choices is not practical. However, with a good degree of certainty, based on results from numerous simulations and several experimental trials in practice, rs_3 might be a better choice from among rs_1, rs_2 and rs_3 considered in this thesis (and proposed in [Tamtsia 13b]). This visual feature always resulted in a very good system conditioning for different types of objects (symmetric, non-symmetric and also for photometric moments that will be treated in Chapter 3). In Table 2.1 are listed images acquired from a fronto-parallel desired configuration and the corresponding condition numbers of the interaction matrices of rs_1, rs_2 and rs_3 evaluated on those images. Clearly, rs_3 exhibits the best conditioning for both the symmetrical and non-symmetrical cases. This is ultimately important from the point of numerical stability of the system [Chaumette 04] and explains the good convergence properties exhibited by rs_3. For this comparison, the visual features were computed from the shift points with $\Delta_1 = 0°$ and $\Delta_1 = 90°$. Indeed, these can also be optimized further for better conditioning by using the joint selection criterion 2.4.3.3. Interestingly, again in this case, rs_3 exhibits the most optimal conditioning. This can be ascertained thanks to our detailed analysis in Section

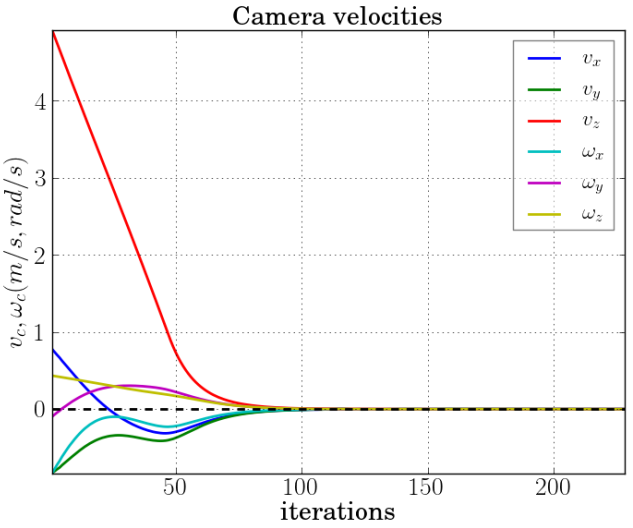


(a) Image at initial pose

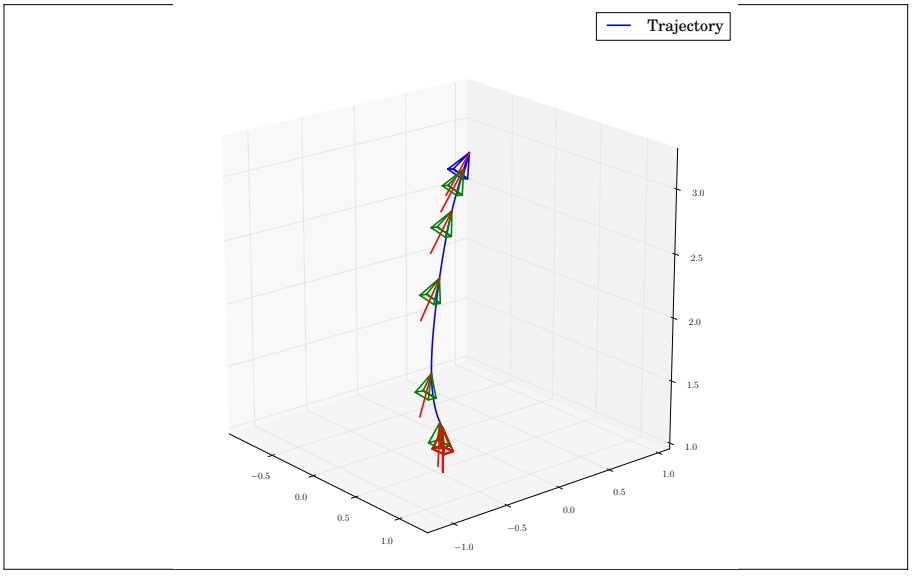
(b) Image learnt from desired pose



(c) Visual feature errors



(d) Control velocities



(e) Spatial trajectory

Figure 2.27 – Servo Results pertaining to Experiment 2.5.5 using rs_3

2.4.4, by comparing the system conditioning in Figures 2.8(c), 2.10 and 2.11(c) for the symmetrical case and in Figures 2.15(c), 2.16(c) and 2.17(c) for the non-symmetrical case. One of the attractive properties of image moments is their large convergence domain. So, several tests involving large displacements were done and in these rs_3 exhibited a very satisfactory behaviour. For instance, we moved the robot $[1m, 1m, 2m]$ in translations and a rotation of 20° around each of the x,y and z axes. The initial image acquired is shown in Fig 2.27(a) and the image learnt from the desired pose in Fig 2.27(b). The control velocities are shown in Fig 2.27(d) while the resulting satisfactory behaviour can be seen from the errors in Fig 2.27(c) and the camera trajectory in Fig 2.27(e).

For some starting poses close to the desired equilibrium, perhaps all the visual features can be convergent. However, in none of our tests, a failure of rs_3 occured where rs_1 or rs_2 exhibited convergence. Based on the above observations, rs_3 has been favoured over the others for use in the subsequent simulations and experiments.

2.6 Summary and Conclusions

In summary, the concept of tunable shift points was introduced. Few selection criteria that can be used to tune them for optimal behaviour was proposed. Moments invariants are then calculated with respect to moments defined with respect to these shift points. Finally, visual features built from these invariants were employed to control the rotational motions around the x and y axes in full 6-DOF VS control schemes.

It was shown that the default choices made in shifted moments may not be the best choice for all cases. Simulations of visual servoing tasks were presented where selection of the shift points were done with respect to the proposed criteria. Selection with these criteria rescued the control from failure and assured a satisfactory convergence. Further, in the case of the orthogonality criterion, it was shown that a decoupled interaction matrix could be obtained. These encompassed both cases of symmetrical and non-symmetrical objects and serve as a validation of the proposed ideas. The proposed approach is interesting since the visual features can be tuned/designed such that a desired characteristic (maximum responsiveness, conditioning) could be obtained from the visual servo. This approach has thus opened some nice possibilities. We sincerely hope that this will inspire more research towards the direction of designing optimal visual features.

As we detailed in the tests, it is hard to single out a specific visual feature and criterion as the best for all cases and in all scenarios. But in general, the joint selection criterion based on the condition number should be favoured owing to its direct connection with system stability. Further, based on observations from our tests and experimental trials, the visual feature rs_3 exhibited a better behaviour (from among the ones tested) and is therefore chosen to be used in the sequel.

Chapter 3

Photometric Moments for Visual Servoing

In this chapter, we describe the central aspect of this thesis : *photometric moments*. An image from a camera contains rich information about the world perceived by the robot. This information is encoded as varying intensity levels also known as luminance. The "raison d'être" of photometric moments is to capture this rich information encoded as intensities and render them useful for visual servoing tasks. The interaction matrix for the photometric moments has been developed in analytical form. It is demonstrated that it is possible to perform visual servoing without using any image processing, matching and visual tracking steps.

3.1 Introduction to Photometric Moments

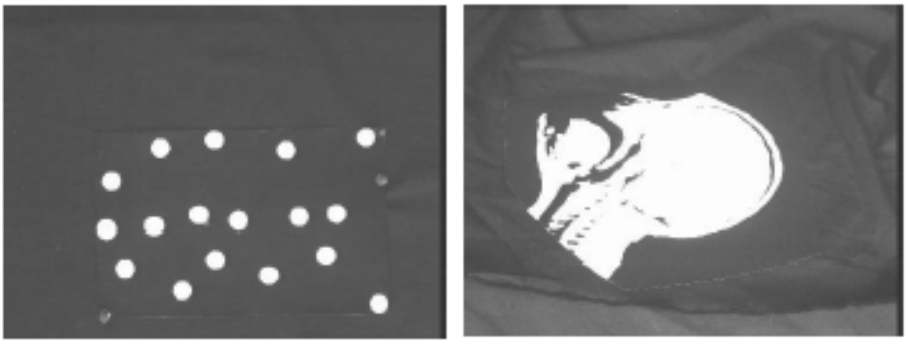
The idea of using intensities in visual servoing is not new. The pure photometric method defines the error directly in terms of the image intensities [Collewet 11] and this redundancy favours a very high degree of accuracy upon convergence. Except for the gradient computation, this method does not employ any other image processing steps. However, the method is highly nonlinear resulting from employing the Levenberg Marquardt method to optimize a nonlinear cost function. This limits the convergence domain of this method. In this work, instead of using intensities directly as a visual feature, photometric image moments are defined using those intensities. This strategy allows to circumvent the non-linearity issue of the pure photometric method. Most importantly, this allows to take advantage of the decoupling properties of the moment invariants.

Previous research showed that progress in visual servoing requires that these methods have to be kept free from image processing and visual tracking processes [Collewet 08a]. These are limitations to the adoption of moments-based visual servoing in practical applications. Recent works [Gridseth 13] [Tatsambon 13] highlighted that the adoption of visual servoing for practical applications is not high. They explicitly explain visual tracking as one among several factors that impede this expansion. That being said, we stick

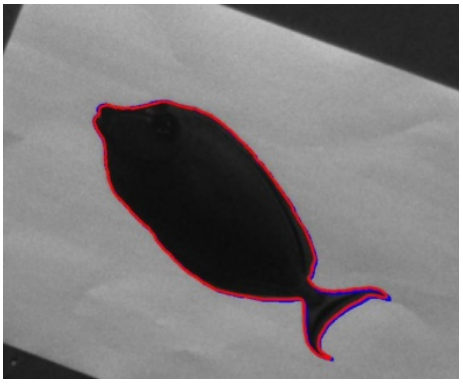
to the same philosophy of photometric methods in not employing any feature extraction, image matching or visual tracking steps. It is possible to define the moments in terms of the intensity distribution from an imaged scene. We have termed these image moments defined from the intensities as *photometric moments*. Photometric moments of an image I can be defined by

$$m_{pq} = \iint_{\pi} x^p y^q \ I((x,y),t) \ \mathrm{d}x\,\mathrm{d}y \tag{3.1}$$

where $I(x,y)$ is the image intensity function and π is the domain of integration. The set $\{x^p y^q\}$ is a polynomial basis and thus image moments are obtained by projection to this basis. The moments thus obtained for different orders are not orthogonal, implying there is some degree of information redundancy among them [Mukundan 98]. If Zernike or Legendre polynomials are used instead as the basis function, we obtain Zernike and Legendre moments respectively which are orthogonal moments.This orthogonality property is useful in pattern recognition [Prokop 92] but this is of no specific importance in visual servoing. We saw in the previous chapter that existing works are based on binary moments. These moments depend on the availability of a set of discrete points or segmented image of a region in the scene (see Figure 3.1). Photometric moments were in fact developed with the goal of overcoming such limitations. They embrace grayscale images which can be quite easily streamed from any standard off-the-shelf monocular cameras. They



(a) Image of a scene with discrete points (b) Image of a scene with a binary target



(c) Image with tracked contours of a region of interest

Figure 3.1 – Reliance of existing methods on the availability of a set of discrete points or segmented interest regions in the scene

take into account intensity information from all the image plane. Photometric moments belong to what can be termed as geometric primitive-less visual servoing, since they do not employ any tracked geometric primitives like points or lines, as done in traditional visual servoing methods.

The developments associated with the interaction matrix for the photometric moments are presented first.

3.2 Modelling of Photometric Moments

The derivative of the photometric moments can be written as

$$\dot{m}_{pq} = \iint_{\pi} x^p y^q \, \dot{I}(x,y) \, \mathrm{d}x \, \mathrm{d}y \tag{3.2}$$

We recall that the photometric moments in Equation (3.1) are computed over the whole image. So, the double integrals here are evaluated over the whole image (or a constant subset of it) and not over a time varying domain $\Omega(t)$ as done previously in Section 2.2. If the derivative of the photometric moments can be expressed in terms of the camera velocity, the interaction matrix of the image moments can be obtained:

$$\dot{m}_{pq} = \mathbf{L}_{m_{pq}} \, \mathbf{v}_c \tag{3.3}$$

To ensure a smooth flow in the developments, it is necessary to recall some steps described in Chapter 1. We then make use of the classical brightness constancy assumption [Horn 81] which considers that the intensity of a moving point $\mathbf{x} = (x, y)$ remains the same between successively acquired images. This constancy can be written by the following well-known equation

$$I(\mathbf{x} + \delta\mathbf{x}, t + \delta t) = I(\mathbf{x}, t) \tag{3.4}$$

where $\delta\mathbf{x}$ is the infinitesimal displacement undergone by the image point after an infinitesimal increment in time δt. For a very small $\delta\mathbf{x}$, a first order Taylor expansion of (3.4) around $\mathbf{x}$ leads to

$$\nabla\mathbf{I}^\top \dot{\mathbf{x}} + \dot{I} = 0 \tag{3.5}$$

where $\nabla\mathbf{I}^\top = \begin{bmatrix} \frac{\partial I}{\partial x} & \frac{\partial I}{\partial y} \end{bmatrix} = \begin{bmatrix} I_x & I_y \end{bmatrix}$ is the spatial gradient at the image point $\mathbf{x}$. We thus obtain

$$\dot{I}(x,y) = -\nabla\mathbf{I}^\top \dot{\mathbf{x}} \tag{3.6}$$

which is the well-known optic flow constraint equation (OFCE) [Horn 81]. It has to be noted here that the optic flow equation itself has been established under several simplifying assumptions: Lambertian surface, point light source at infinity and absence of photometric distortions. Further, this simplifying assumption has also been utilised in [Collewet 08a] to model intensity variations. This is one of the simplest and practical choices that could be made. Another approach was followed in [Collewet 11], where the Phong illumination model [Phong 75] was used to model the intensity variations. An analytical form could be developed containing ambient and diffuse components in this case. In practice however, it would be extremely difficult to deal with all the variables that have

to be measured taking into account several configurations between the light sources, the camera and the target.

We know that $\dot{\mathbf{x}} = \mathbf{L}_{\mathbf{x}}\mathbf{v}_c$ and $\mathbf{L}_{\mathbf{x}}$, the interaction matrix which links the image point velocity to the camera velocity was already developed in Section 1.2.1.1 and reproduced below:

$$\mathbf{L}_{\mathbf{x}} = \begin{bmatrix} \mathbf{L}_x \\ \mathbf{L}_y \end{bmatrix} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \tag{3.7}$$

Let us consider a 3D scene that does not present significant depth discontinuities. In this manner, the depth of the scene points can be related to the image point co-ordinates by the relation

$$\frac{1}{Z} = Ax + By + C \tag{3.8}$$

where A, B and C are scalar parameters that describe the configuration of the scene plane as explained in the modelling in Section 2.2. By plugging (3.8) into (3.7) and $\dot{\mathbf{x}} = \mathbf{L}_{\mathbf{x}}\mathbf{v}_c$ into (3.6), we obtain

$$\dot{I}(x,y) = -\nabla\mathbf{I}^\top \mathbf{L}_{\mathbf{x}}\ \mathbf{v}_c = \mathbf{L}_I\ \mathbf{v}_c \tag{3.9}$$

where $\mathbf{L}_I = -\nabla\mathbf{I}^\top\mathbf{L}_{\mathbf{x}}$. $\mathbf{L}_I$ can be expanded and written as

$$\mathbf{L}_I^\top = \begin{bmatrix} \frac{\partial I}{\partial x}(Ax+By+C) \\ \frac{\partial I}{\partial y}(Ax+By+C) \\ (-x\frac{\partial I}{\partial x} - y\frac{\partial I}{\partial y})(Ax+By+C) \\ -xy\frac{\partial I}{\partial x} - (1+y^2)\frac{\partial I}{\partial y} \\ (1+x^2)\frac{\partial I}{\partial x} + xy\frac{\partial I}{\partial y} \\ -y\frac{\partial I}{\partial x} + x\frac{\partial I}{\partial y} \end{bmatrix} \tag{3.10}$$

Substituting (3.9) into the equation for the moment derivatives (3.2), we see that

$$\dot{m}_{pq} = \iint_\pi x^p y^q\, \mathbf{L}_I\, \mathrm{d}x\, \mathrm{d}y\, \mathbf{v}_c \tag{3.11}$$

We can then write down the interaction matrix of the moments as

$$\mathbf{L}_{m_{pq}} = \iint_\pi x^p y^q\, \mathbf{L}_I\, \mathrm{d}x\, \mathrm{d}y \tag{3.12}$$

Direct substitution of $\mathbf{L}_I$ defined in (3.10) into the above equation gives us

$$\mathbf{L}_{m_{pq}}^{\top} = \begin{bmatrix} \iint_{\pi} x^p y^q \, \dfrac{\partial I}{\partial x}(Ax + By + C)\,\mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, \dfrac{\partial I}{\partial y}(Ax + By + C)\,\mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (-x\dfrac{\partial I}{\partial x} - y\dfrac{\partial I}{\partial y})(Ax + By + C)\,\mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (-xy\dfrac{\partial I}{\partial x} - (1+y^2)\dfrac{\partial I}{\partial y})\,\mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, ((1+x^2)\dfrac{\partial I}{\partial x} + xy\dfrac{\partial I}{\partial y})\,\mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (x\dfrac{\partial I}{\partial y} - y\dfrac{\partial I}{\partial x})\,\mathrm{d}x\,\mathrm{d}y \end{bmatrix} \tag{3.13}$$

The following compact notation is introduced for convenience and fluidity of the ensuing developments.

$$m_{pq}^{\nabla x} = \iint_{\pi} x^p y^q \, \frac{\partial I}{\partial x}\,\mathrm{d}x\,\mathrm{d}y \tag{3.14a}$$

$$m_{pq}^{\nabla y} = \iint_{\pi} x^p y^q \, \frac{\partial I}{\partial y}\,\mathrm{d}x\,\mathrm{d}y \tag{3.14b}$$

A component-wise representation of $L_{m_{pq}}$ can be written as:

$$\mathbf{L}_{m_{pq}} = \begin{bmatrix} L_{\mathtt{m_{pq}}}^{v_x} & L_{\mathtt{m_{pq}}}^{v_y} & L_{\mathtt{m_{pq}}}^{v_z} & L_{\mathtt{m_{pq}}}^{\omega_x} & L_{\mathtt{m_{pq}}}^{\omega_y} & L_{\mathtt{m_{pq}}}^{\omega_z} \end{bmatrix} \tag{3.15}$$

As a representative example, let us consider a single component of the interaction matrix; the one corresponding to the translational velocity in x.

$$\begin{aligned} L_{\mathtt{m_{pq}}}^{v_x} &= \iint_{\pi} x^p y^q \, (Ax + By + C)\, \frac{\partial I}{\partial x}\,\mathrm{d}x\,\mathrm{d}y \\ &= A \iint_{\pi} x^{p+1} y^q \, \frac{\partial I}{\partial x}\,\mathrm{d}x\,\mathrm{d}y + B \iint_{\pi} x^p y^{q+1} \, \frac{\partial I}{\partial x}\,\mathrm{d}x\,\mathrm{d}y \\ &\quad + C \iint_{\pi} x^p y^q \, \frac{\partial I}{\partial x}\,\mathrm{d}x\,\mathrm{d}y) \\ &= A\, m_{p+1,q}^{\nabla x} + B\, m_{p,q+1}^{\nabla x} + C\, m_{p,q}^{\nabla x} \end{aligned} \tag{3.16}$$

With similar developments, the components corresponding to each degree of freedom can

be obtained

$$\left.\begin{aligned}
L^{v_x}_{\mathtt{m_{pq}}} &= A\, m^{\nabla x}_{p+1,q} + B\, m^{\nabla x}_{p,q+1} + C\, m^{\nabla x}_{p,q} \\
L^{v_y}_{\mathtt{m_{pq}}} &= A\, m^{\nabla y}_{p+1,q} + B\, m^{\nabla y}_{p,q+1} + C\, m^{\nabla y}_{p,q} \\
L^{v_z}_{\mathtt{m_{pq}}} &= -A\, m^{\nabla x}_{p+2,q} - B\, m^{\nabla x}_{p+1,q+1} - C\, m^{\nabla x}_{p+1,q} \\
&\quad - A\, m^{\nabla y}_{p+1,q+1} - B\, m^{\nabla y}_{p,q+2} - C\, m^{\nabla y}_{p,q+1} \\
L^{\omega_x}_{\mathtt{m_{pq}}} &= -\, m^{\nabla x}_{p+1,q+1} - \, m^{\nabla y}_{p,q} - \, m^{\nabla y}_{p,q+2} \\
L^{\omega_y}_{\mathtt{m_{pq}}} &= m^{\nabla x}_{p,q} + \, m^{\nabla x}_{p+2,q} + \, m^{\nabla y}_{p+1,q+1} \\
L^{\omega_z}_{\mathtt{m_{pq}}} &= -\, m^{\nabla x}_{p,q+1} + \, m^{\nabla y}_{p+1,q}
\end{aligned}\right\} \tag{3.17}$$

We can observe that the expressions exhibit a complex form and involves image gradients. The terms $m^{\nabla x}_{p,q}$ and $m^{\nabla y}_{p,q}$ need to be computed which add to the computational burden. Further, image gradients are normally computed using derivative filters, which might introduce imprecision in the computed values. So, a further simplification step using Green's theorem was devised, partly to avoid this imprecision. The simplification step also led to an interesting result, which is presented next.

3.2.1 Simplifications using Green's Theorem

We proceed by simplifying the terms $m^{\nabla x}_{pq}$ and $m^{\nabla x}_{pq}$ in the above expressions. We begin with

$$m^{\nabla x}_{pq} = \iint_{\pi} \frac{\partial I(x,y)}{\partial x} x^p y^q \,\mathrm{d}x\,\mathrm{d}y \tag{3.18}$$

Green's theorem is an elegant mathematical tool which lets us compute the integral of a function defined over a subdomain π of $\mathbb{R}^2$ by transforming it into a line (curve/contour) integral over the boundary of π, denoted here as $\partial\pi$:

$$\iint_{\pi} (\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y})\mathrm{d}x\,\mathrm{d}y = \oint_{\partial\pi} P\mathrm{d}x + \oint_{\partial\pi} Q\mathrm{d}y \tag{3.19}$$

If we let $Q = I(x,y)\, x^p\, y^q$ and $P = 0$, then

$$\left.\begin{aligned}
\frac{\partial Q}{\partial x} &= \frac{\partial I}{\partial x}\, x^p\, y^q \, + \, p\, x^{p-1}\, y^q\, I(x,y) \\
\frac{\partial P}{\partial y} &= 0
\end{aligned}\right\} \tag{3.20}$$

Replacing these terms in Green's theorem, we obtain

$$\iint_{\pi} \Big[\frac{\partial I}{\partial x}\, x^p\, y^q \, + \, p\, x^{p-1}\, y^q\, I(x,y)\Big]\mathrm{d}x\,\mathrm{d}y = \oint_{\partial\pi} I(x,y)\, x^p\, y^q \mathrm{d}y \tag{3.21}$$

from which we deduce

$$\iint_{\pi} \frac{\partial I}{\partial x}\, x^p\, y^q \,\mathrm{d}x\,\mathrm{d}y = -\iint_{\pi} \, p\, x^{p-1}\, y^q\, I(x,y)\mathrm{d}x\,\mathrm{d}y + \oint_{\partial\pi} x^p\, y^q\, I(x,y)\mathrm{d}y \tag{3.22}$$

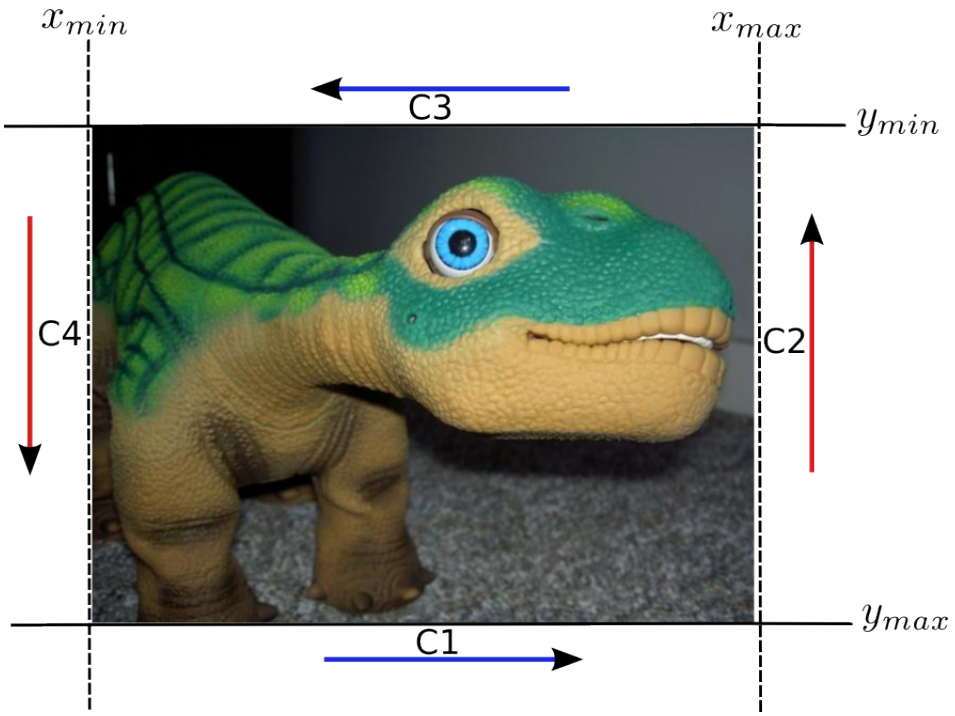


Figure 3.2 – Illustration to demonstrate evaluation of contour integrals in the interaction matrix developments

Recalling our compact notation from Equation (3.14), we see that the first term is equal to $-pm_{p-1,q}$ and is straight-forward to obtain from the basic moments. The second term is a contour integral along $\partial\pi$. We introduce the limits $y = y_{min}$ at the top and $y = y_{max}$ at the bottom of the image. Since $y(= y_{max})$ is constant along $C1$ and $y(= y_{min})$ is constant along $C3$, it is sufficient to integrate along $C2$ and $C4$ (marked by dotted lines in Fig.3.2) to obtain the second term. Proceeding to this next step of integrating along C_2 and C_4, we have

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} + \oint_{C_2} x^p\, y^q\, I(x,y)\mathrm{d}y + \oint_{C_4} x^p\, y^q\, I(x,y)\mathrm{d}y \tag{3.23}$$

Along C_2, y varies from y_{max} to y_{min} while x remains constant at x_{max}. Along C_4, y varies from y_{min} to y_{max} while x remains constant at x_{min}. Introducing carefully these limits, we get

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} + x_{max}^p \int_{y_{max}}^{y_{min}} y^q\, I(x_{max},y)\mathrm{d}y + x_{min}^p \int_{y_{min}}^{y_{max}} y^q\, I(x_{min},y)\mathrm{d}y \tag{3.24}$$

Zero Border Assumption (ZBA) : At this juncture, we make the assumption that the intensity all along the image borders is uniform. This means that images with a ***black background*** are to be used in the visual servoing. Works on KBVS [Kallem 07] [Kallem 09] also introduce this assumption implicitly by using kernels with finite support. This zero border assumption (ZBA) seems reasonable in practice for many cases, as evidenced by

our results. In this case,

$$I(x_{max}, y) = I(x_{min}, y) = I, \forall y \tag{3.25}$$

Therefore, (3.24) can be further simplified to the below form:

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} + (x_{max}^p - x_{min}^p)\, I \int_{y_{max}}^{y_{min}} y^q \mathrm{d}y \tag{3.26}$$

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} + \frac{I}{q+1}(x_{max}^p - x_{min}^p)\left[y_{min}^{q+1} - y_{max}^{q+1}\right] \tag{3.27}$$

From the introduced assumption of the black background[1], I has to be 0. Thus, the black background naturally leads to the second term being evaluated to 0. After this elimination, we obtain finally a simple form for $m_{pq}^{\nabla x}$.

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} \tag{3.28}$$

Without the ZBA, to be precise, the second integral term would have to be evaluated using the appropriate limits and intensities present along the left and right borders of the image (see Fig.3.2). This assumption gets eliminated with an improved modelling scheme that will be presented in Chapter 4, Section 4.2.

In a similar way, we can use the same theorem to simplify the computation of the term $m_{pq}^{\nabla y}$. The substitutions $P = I(x, y)\, x^p\, y^q$ and $Q = 0$ are made in Green's theorem. The corresponding partial derivatives are given by

$$\frac{\partial P}{\partial y} = \frac{\partial I(x, y)}{\partial y}\, x^p\, y^q \;+\; x^p\, q\, y^{q-1}\, I(x, y) \tag{3.29}$$

$$\frac{\partial Q}{\partial x} = 0 \tag{3.30}$$

Substitution into Green's equation (3.19) gives

$$\iint_{\pi} [-\frac{\partial I}{\partial y}\, x^p\, y^q \;-\; x^p\, q\, y^{q-1} I(x, y)]\, \mathrm{d}x\, \mathrm{d}y = \oint_{\partial\pi} x^p\, y^q\, I(x, y) \mathrm{d}x \tag{3.31}$$

Identical steps starting from Equation(3.22) can be applied for the simplification. The zero border assumption (ZBA) will again come into play, this time the integrals would be evaluated along C_1 and C_3, assuming that $I(x, y_{min}) = I(x, y_{max}) = 0, \forall x$. The final expression is as follows:

$$m_{pq}^{\nabla y} = -q\, m_{p,q-1} \tag{3.32}$$

[1] A white background can also be assumed in which case the intensity has to be redefined as $I_{max} - I$. In this case, I should equal I_{max} and the rest of the developments would be identical.

Substituting Equations (3.28) and (3.32) into (3.17), we get the final closed form expressions for the interaction matrix.

$$\left.\begin{aligned}
L_{\mathtt{m}_{\mathtt{pq}}}^{v_x} &= -A(p+1)m_{pq} - Bpm_{p-1,q+1} - Cpm_{p-1,q} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{v_y} &= -Aqm_{p+1,q-1} - B(q+1)m_{p,q} - Cqm_{p,q-1} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{v_z} &= A\,(p+q+3)\,m_{p+1,q} + B(p+q+3)\,m_{p,q+1} \\
&\quad + C(p+q+2)\,m_{pq} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_x} &= q\,m_{p,q-1} + (p+q+3)\,m_{p,q+1} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_y} &= -p\,m_{p-1,q} - (p+q+3)\,m_{p+1,q} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_z} &= p\,m_{p-1,q+1} - q\,m_{p+1,q-1}
\end{aligned}\right\} \tag{3.33}$$

- With respect to Equation (3.17), the image gradients do not appear anymore in the interaction matrix. This is interesting because this was in fact an image processing step (and the only one) required by pure luminance based visual servoing method [Collewet 11]. With these advancements, it is sufficient to compute the moments only to have the interaction matrix.

- Exactly the same analytical forms as in Equation (2.29) has been obtained. It is interesting to note that our case used a method completely different from those in Sections 2.2.1 and 2.2.2 for the developments. So, all the useful results described in the state of the art and recalled in Chapter 2 as regards to the development of visual features are applicable as they are for photometric moments. For instance, in order to calculate $L_{m_{pq}}$, only moments of order upto $p+q+1$ are required. We also note again that the interaction matrix components corresponding to the rotational degrees of freedom are free from 3D parameters A, B and C. All the available moment invariants and theory about choice of visual features discussed previously are valid.

We recall that we have interesting results despite the zero border assumption we have made in the developments. This assumption will be eliminated with the improved model we propose in the following chapter.

3.3 Relation to Kernel based Visual Servoing

As briefly explained in Chapter 1, a strand of research that is close to photometric moments is kernel-based visual servoing (KBVS) [Kallem 07]. The abstract formulation of KBVS is conceptually very elegant. It has some patent disadvantages, the most critical being there are no kernel features proposed to control the rotational degrees of freedom which are orthogonal to the optic axis. There are also other subtle constraints which prevent deployment of KBVS for real world robotic systems. Moreover, if the signal is the image itself, i.e., $S(\mathbf{w},t) = I((x,y),t)$ and the kernel $K(\mathbf{w}) = x^p y^q$, photometric moments are indeed obtained. Each moment of order $p+q$ therefore is a kernel projected value. Photometric moments are thus a specific manifestation of kernel-based visual servoing (KBVS). But there are several differences between photometric moments developed in this work and the research work developed under KBVS. These points are tabulated in Table 3.1.

Kernel Based Visual servoing (KBVS)	Photometric moments based visual servoing
Kernel Based Visual servoing proposed an abstract formulation of a visual feature $\xi(t) = \iint_{\pi} K(x,y)\ I((x,y),t))\ \mathrm{d}x\,\mathrm{d}y$	Photometric moments are image moments obtained by projecting the image to a polynomial basis function $m_{pq} = \iint_{\pi} x^p y^q\ I((x,y),t)\ \mathrm{d}x\,\mathrm{d}y$
Has no solution for control of the rotational degrees of freedom orthogonal to the optic axis.	Builds upon well-established theories about moment invariants to attack this non-trivial problem.
No emphasis is laid on decoupling properties for the control law.	Emphasis is laid on invariance properties of moments to design decoupled VS control laws.
Inspired by [Cideciyan 95], uses the spatial Fourier Transform (FT) to control both translations along and rotations around the optic axis. The use of FT opens the door to possible ringing effects. For the spatial FT to be exact, images with white foreground and backgrounds of black are considered.	Uses 2 different visual features, simple functions of the basic moments to control translations along and rotations around z. Photometric moments are free from any frequency domain features.
To control planar translations, KBVS uses gaussian kernels along the x and y directions either independently or stacked together. These kernels are defined with a finite support. So the integration using $K(x,y)$ is finitely truncated but the limits of this truncation are not explicitly specified [Swensen 10]. In this work on convergence characterization, details of whether a single 2D kernel is used or two separate kernels are employed is not clear. This makes reproduction of the experiments a bit difficult.	To control the planar translations, photometric moments employ the photometric centre of gravity obtained as simple ratios of basic moments in a straight-forward manner.
Kernel feature proposed for control of roll motions are affected by translational motions along the optic axis. Therefore, depth motions need to converge before the control law for roll takes effect.	The visual feature proposed for control of roll motions is independent of the motions along the optic axis. So no precedence of error convergence in specific degrees of freedom is required.
Neither the number of kernels nor optimal placement of these kernels for use in particular scenes have been investigated. Whether to place a kernel at every pixel or at specific interest locations in the scene is an open issue.	The reasons behind the choice of the visual feature to control the individual degrees of freedom have been discussed.

Table 3.1 – Contrasts between photometric moments and kernel-based visual servoing

3.4 Simulation Results

In this section, we present simulation studies of 6DOF positioning tasks achieved using photometric moments. For the results in this section, the control law $\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{+}(\mathbf{s} - \mathbf{s}^*)$ is used with $\mathbf{L}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}(\mathbf{s}(t), Z(t))$ for validating the modelling proposed in the previous section. No image processing, matching or visual tracking steps were used in the reported experiments.

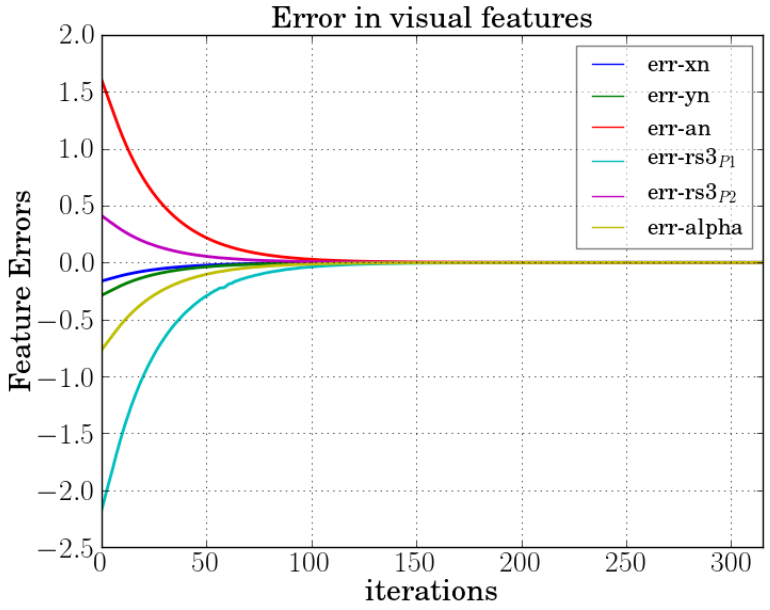
3.4.1 Modelling Validation and Comparison to Pure Luminance

The initial image is shown in Fig 3.3(a) and the desired image is shown in Fig.3.3(b). Let us note that these images are consistent with the zero border assumption (ZBA) made in the developments in Section 3.2.1. The initial and desired poses are chosen such that the image overlap is small. This would typically be the case which will result when the initial pose is far away from the desired one. The displacements required for convergence are a translation of $\mathbf{t} = [1.0m, 1.0m, 1.0m]$ and a rotation of $\mathbf{R} = [25°, 10°, 55°]$. The current interaction matrix $\mathbf{L}_{\mathbf{s}}(\mathbf{s} = \mathbf{s}(t), Z(t))$ is used. This choice enables the validation of the modelling by checking if the control law using the photometric moments result in an exponential convergence of the errors to 0. In simulation, the depths $Z(t)$ are readily available from ground truth and need not be estimated. A gain of $\lambda = 1.0$ was used for this experiment. The visual feature rs_3 with shift angles fixed at $\Delta_1 = 0°$ and $\Delta_2 = 90°$ was chosen for the validation. In principle however, any of the visual features proposed in the previous chapter but computed from the photometric moments can be used to perform this validation.

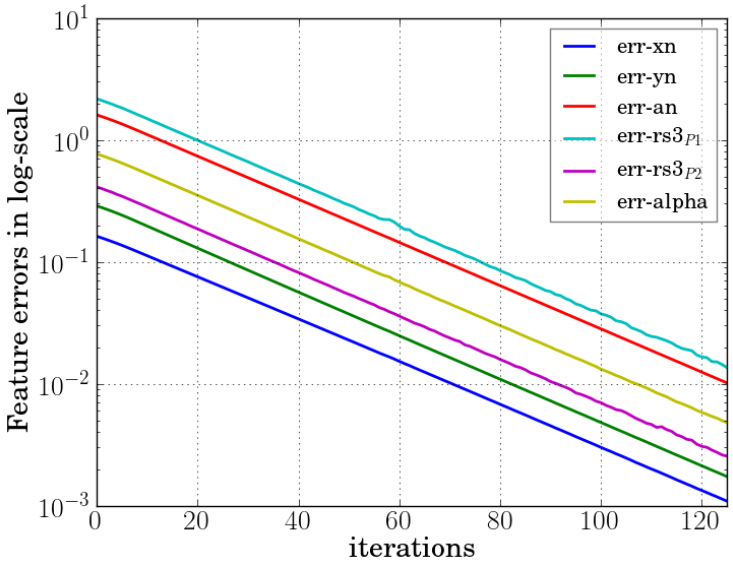
(a) Initial image (b) Desired image

Figure 3.3 – Images for visual servoing with less overlap

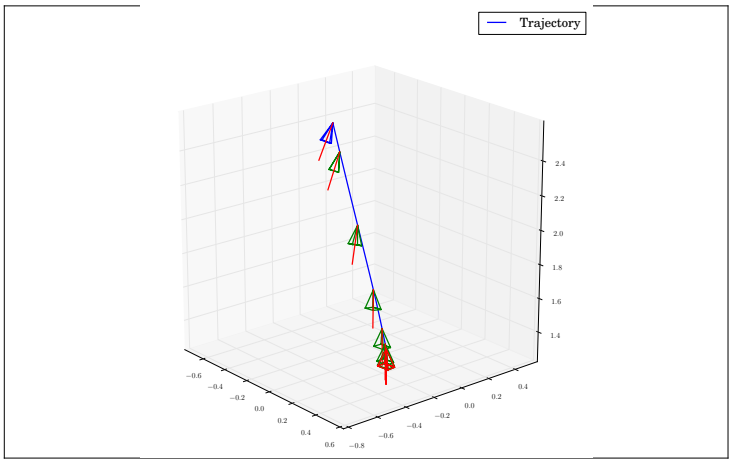
A perfect exponential decrease of the errors is obtained with this control law using the photometric moments in Fig 3.4(a). This can also be ascertained from the parallel straight lines observed in the semilog error plot shown in Fig 3.4(b). Further, a very satisfactory spatial trajectory is obtained as shown in Fig 3.4(c).



(a) Errors in visual features

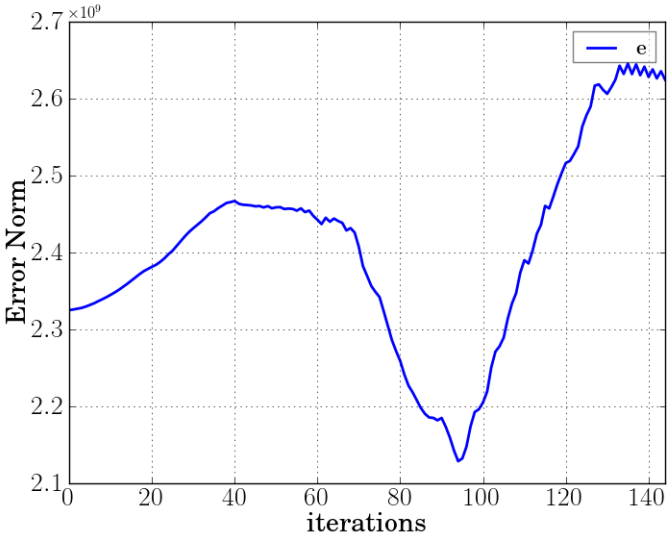


(b) Errors in log-scale to observe exponential decrease

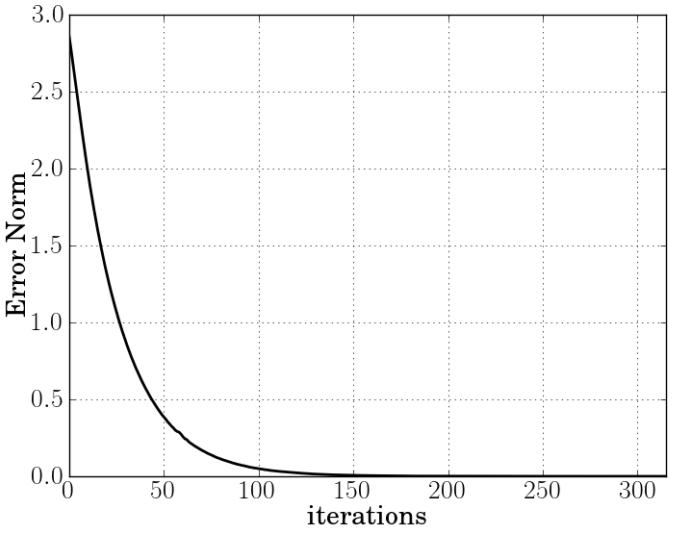


(c) Camera 3-D trajectory

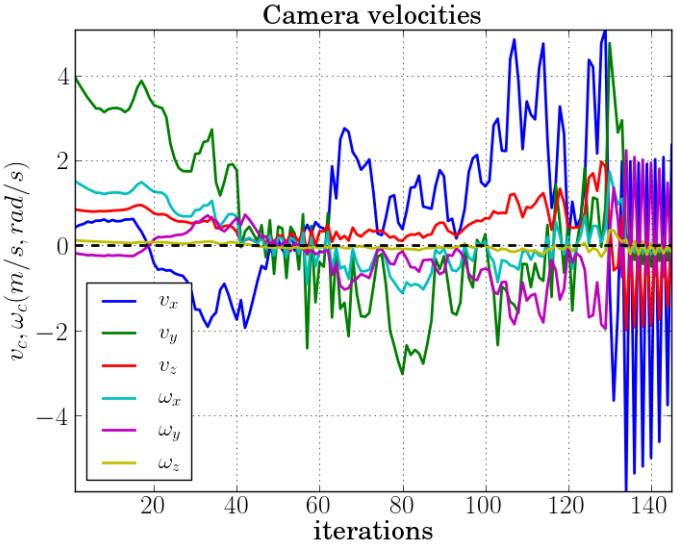
Figure 3.4 – Visual Servoing Results with Photometric Moments using $\widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}(t)$



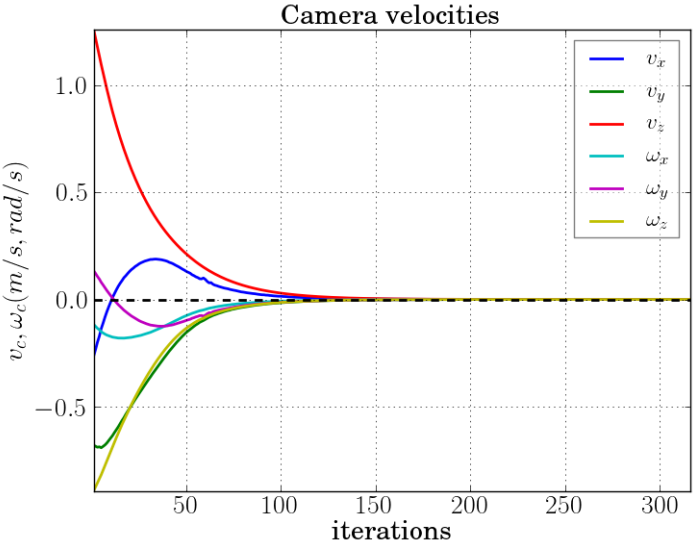
(a) Pure photometric VS : Error Norm $\|\mathbf{I} - \mathbf{I}^*\|$



(b) Photometric moments: Error Norm $\|\mathbf{s} - \mathbf{s}^*\|$



(c) Pure photometric VS : Camera velocities



(d) Camera velocities

(e) Pure photometric VS : Difference image

(f) Photometric moments : Difference image

Figure 3.5 – Comparison of photometric moments and pure photometric VS

Comparison to using pure luminance

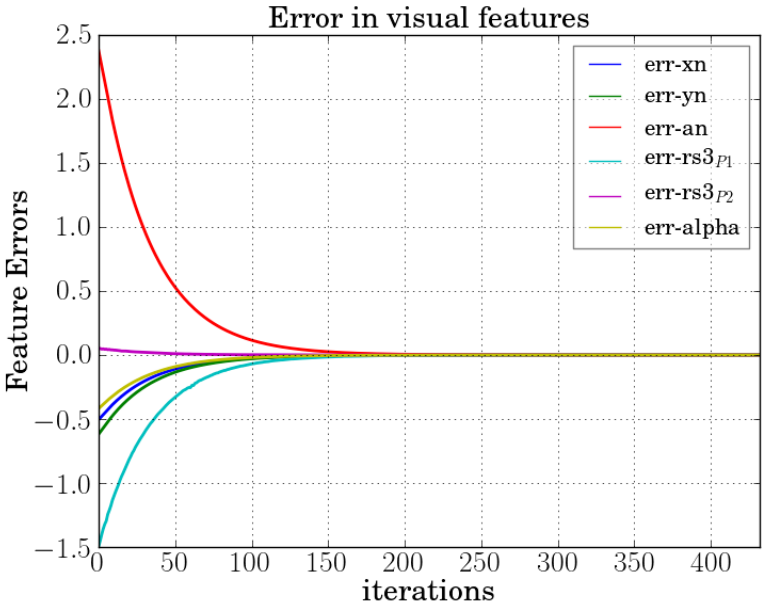
To compare the photometric moments to pure photometric visual servoing, we employ the same control law (Gauss-Newton) employed for the photometric moments. But instead of using the visual features from photometric moments, the set of intensities $\mathbf{I}$ from the image are used directly as visual features. The pure luminance experiment does not converge as evidenced from the huge error in Fig 3.5(a). The velocity profiles generated by the pure photometric features is shown in Figure 3.5(c). The control does not result in a steady decrease of the errors. Instead there is a fluctuation and the image leaves the camera field-of-view. The difference image obtained when more than half of the image left the field-of-view is shown in 4.14(b). This is not surprising since the luminance features have a limited convergence domain as explained in Chapter 1. In comparison, the control velocities computed using the photometric moments (shown in Fig 3.5(d)) ensure that the error in the system decreases exponentially, as seen from Fig 3.5(b).

Performance under large motions

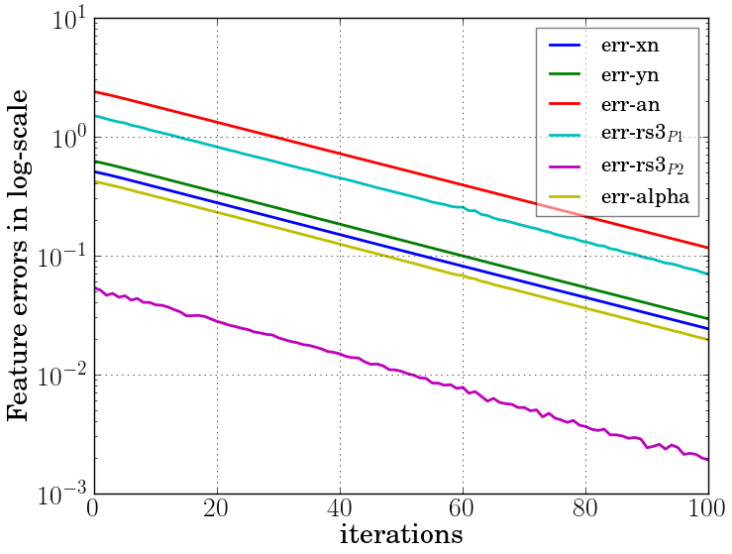
As explained previously, one of the reasons image moments are attractive candidates for visual servoing is that their performance is satisfactory even under large displacements. So, to check this with the photometric moments, we choose a distant initial pose such that the overlap between the desired and current images is lesser than in the previous experiment. The camera observes the image in Figure 3.6(a) from its initial pose. A translational displacement of $\mathbf{t} = [2.0m, 1.0m, 1.0m]$ and a rotational displacement of $\mathbf{R} = [25°, 30°, 55°]$ is necessary to attain the desired pose. The desired image is shown in Fig 3.6(b). The same control law as used in the previous case is employed.

(a) Photometric moments (b) Pure photometric VS

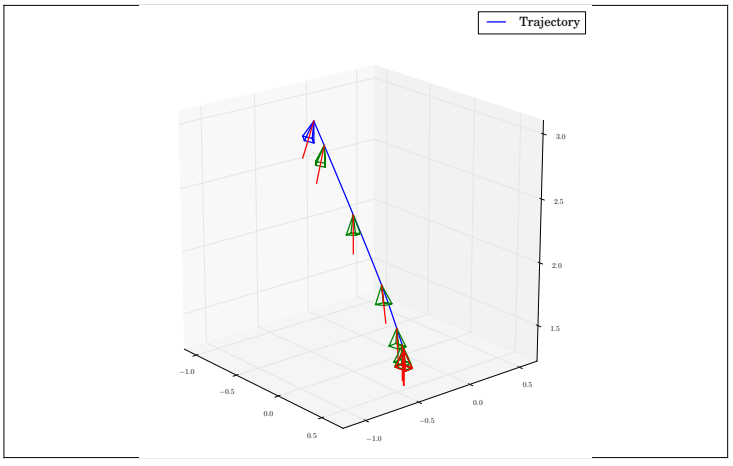
We obtain again a very satisfactory behaviour for the photometric moments. The exponential decrease in the errors can be observed from Figures 3.6(c) and 3.6(d). As shown in Fig 3.6(e), the camera takes a straight path between the initial and the desired poses. In comparison, with the pure luminance features, the generated control velocities (Fig 3.7(c)) resulted in the system being driven away from the desired pose. This is reflected in the error norm of the features shown in 3.7(a) as well as the difference image shown in Fig 3.7(e).



(c) Errors in visual features

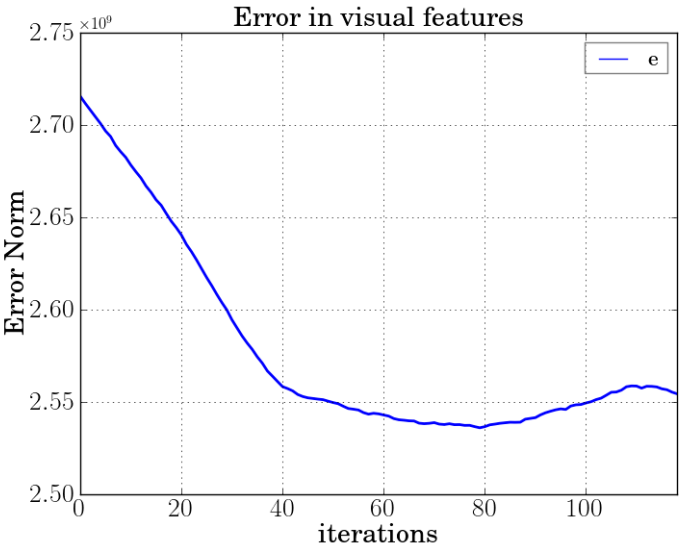


(d) Errors in log-scale to observe exponential decrease

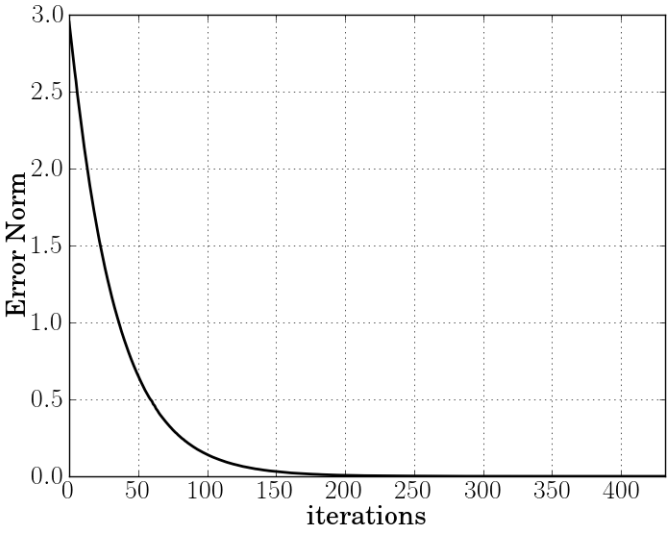


(e) Camera 3-D trajectory

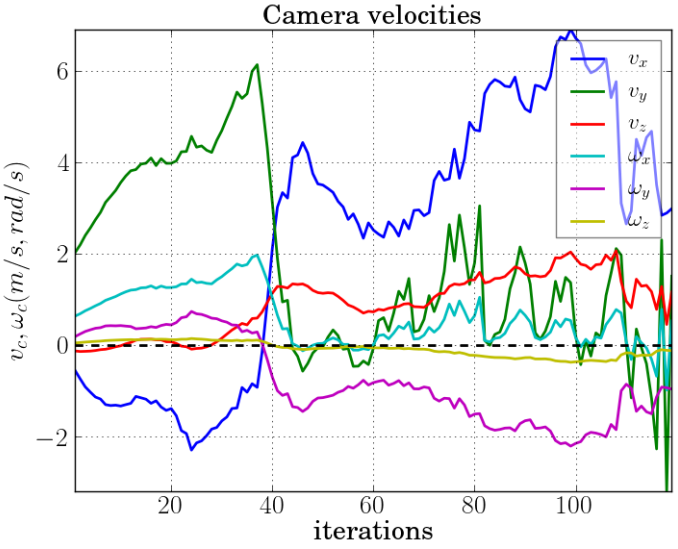
Figure 3.6 – Visual Servoing Results with Photometric Moments using $\widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}(t)$



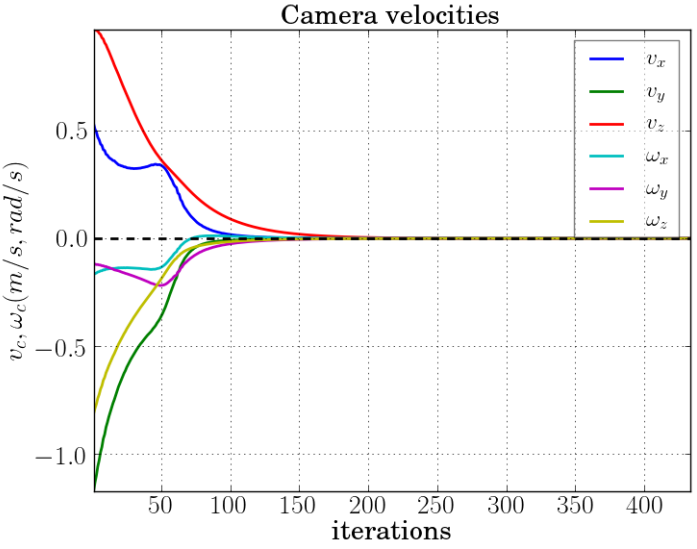
(a) Pure photometric VS : Error Norm $\|\mathbf{I} - \mathbf{I}^*\|$



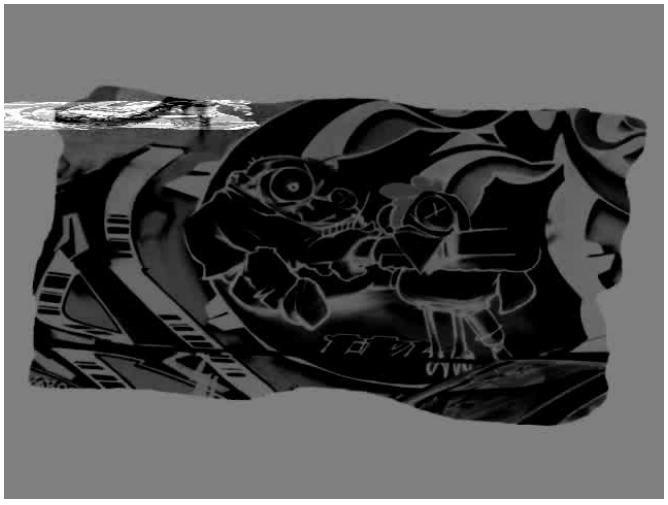
(b) Photometric moments: Error Norm $\|\mathbf{s} - \mathbf{s}^*\|$



(c) Pure photometric VS : Camera velocities



(d) Camera velocities



(e) Pure photometric VS : Difference image

(f) Photometric moments : Difference image

Figure 3.7 – Comparison of photometric moments and pure photometric VS

These tests with the photometric moments validate the correctness of the analytical modelling performed in Section 3.2 to develop the interaction matrix of the photometric moments.

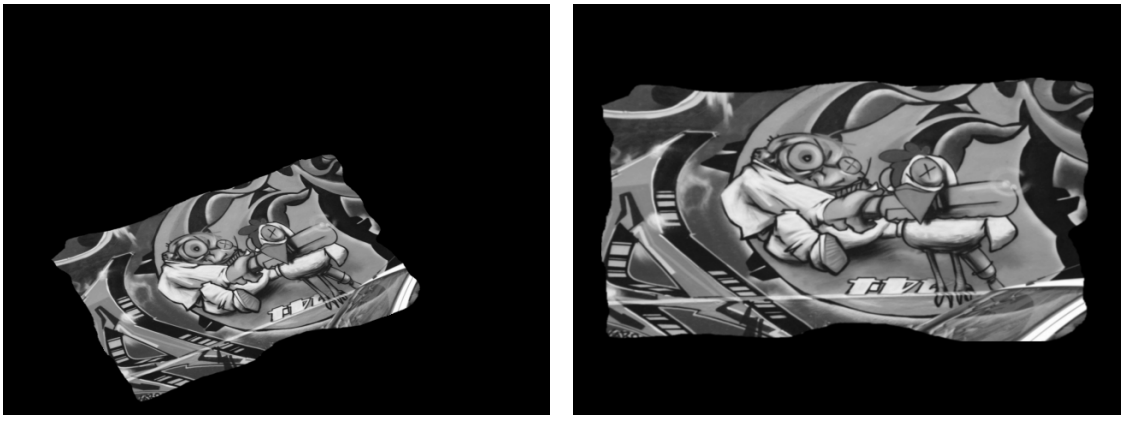
3.4.2 Tests with different configurations of the interaction matrix

In this section, we present simulation results obtained with photometric moments with different configurations of the interaction matrix. The depth was approximated at Z^* for all the configurations (see first column in Table 3.2). A gain of $\lambda = 2.0$ was employed for all the experiments. The desired pose is fixed at a distance of $1.25m$ vertically above the target such that its plane is parallel to the target plane. The initial pose is fixed such that a camera displacement of ${}^{c*}\mathbf{M}_c = [-30cm, 30cm, -60cm, 15°, 10°, 25°]$ is required for the positioning task to be achieved. The initial image is shown in Figure 3.8(a) and the desired image in Fig 3.8(b). The system is required to attain an error level of $1e^{-10}$ for convergence. We also consider the *geodesic area* metric proposed in [Gans 03]. This metric measures the area between the geodesic and the Cartesian trajectory obtained from a visual servoing experiment.

The decrease in errors is satisfactory in all these cases. In the case of the current interaction matrix $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}\left(\mathbf{s}(t), \hat{Z}^*\right)$, an almost exponential decrease of the errors is obtained (See Fig 3.9(a)). The corresponding velocities are shown in Fig 3.9(b). There is no direct control over the spatial trajectories in this case.

Next, let us see the advantages of using the interaction matrix at the desired configuration $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}^*, \hat{Z}^*)$. For this experiment, we obtained

$$\widehat{\mathbf{L}}_{\mathbf{s}*} = \begin{bmatrix} -1 & 0 & 0 & 0.0009 & -1.3338 & 0.0041 \\ 0 & -1 & 0 & 1.2922 & -0.0009 & -0.0059 \\ 0 & 0 & -1 & -0.0062 & 0.0088 & 0 \\ 0 & 0 & 0 & -0.5798 & -2.7709 & 0 \\ 0 & 0 & 0 & 0.4711 & 0.2394 & 0 \\ 0 & 0 & 0 & 0.0657 & -0.0634 & -1 \end{bmatrix} \tag{3.34}$$



(a) Image at initial pose (b) Image learnt at desired pose

Figure 3.8 – Images from the initial and desires poses selected for the visual servoing

with a condition number of 12.96 which is excellent from the point of view of numerical stability. Further, the translation part of the first 3 features is a block diagonal matrix. As for the features for control of rotational motions, that is, for $rs3$ (computed from the shifted moments) and the orientation α, the translational part of the interaction matrix is equal to 0. Thus, for the photometric moments, we find again the same wonderful decoupling properties that existed for the moments-based methods [Chaumette 04] [Tahri 05a] in the state of the art. Of course, with these nice results, the zero border assumption (ZBA) used in the developments should be valid. This restrictive assumption will be dealt with in the forthcoming chapter. Further, the decrease in errors is satisfactory but not strictly exponential. The Cartesian behaviour on the other hand is better when compared with the previous trajectory from using the interaction matrix at the current configuration. The Cartesian behaviour error metrics are tabulated in Table 3.2. The spatial trajectories resulting from these configurations are shown in Fig 3.4.2. It is straight-forward to infer that using the mean of the interaction matrices yields a performance midway between usage of the desired and the current interaction matrices (see Figures 3.9(e), 3.9(f) and 3.4.2).

Interaction Matrix	Camera Distance (cm)	Geodesic Area (m^2)
$\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}\left(\mathbf{s}(t), \hat{Z}^*\right)$	68.07	0.0320
$\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*)$	67.04	0.0246
$\widehat{\mathbf{L}}_{\mathbf{s}} = \left(\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*) + \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), \hat{Z}^*)\right)/2$	67.37	0.0279

Table 3.2 – Cartesian behaviour error metrics with different interaction matrix configurations

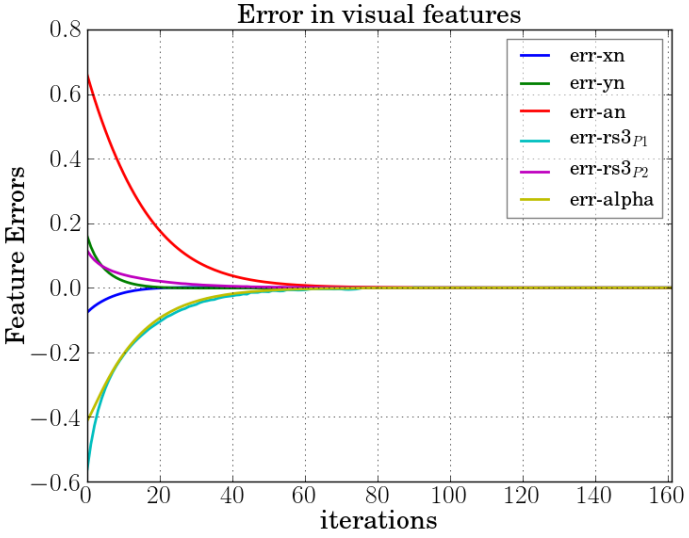
3.5 Experimental Results

This section presents real experiments conducted on the Afma6 gantry robot platform (See Figure 3.11) and the results achieved. The gantry robot has a large workspace that permits to test the visual servoing algorithms with large displacements. A firewire (IEEE1394) camera is mounted on the end-effector conforming to an eye-in-hand configuration. The camera streams images at a 320x240 resolution. We used images with several different textures in our experiments. We describe here only a representative subset of the extensive experiments that were performed. We have not used image processing or visual tracking for the servoing experiments presented. Two sets of positioning experiments were conceived.

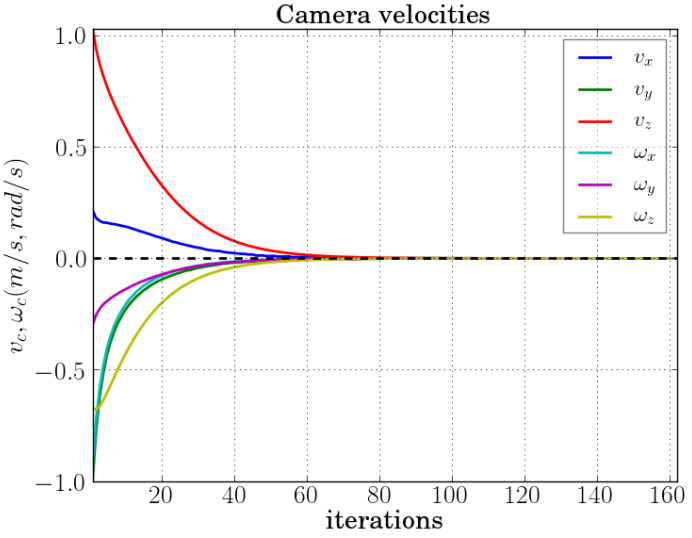
3.5.1 Visual Servoing with SCARA-type actuation

3.5.1.1 Methodology for positioning tasks with SCARA-type actuation

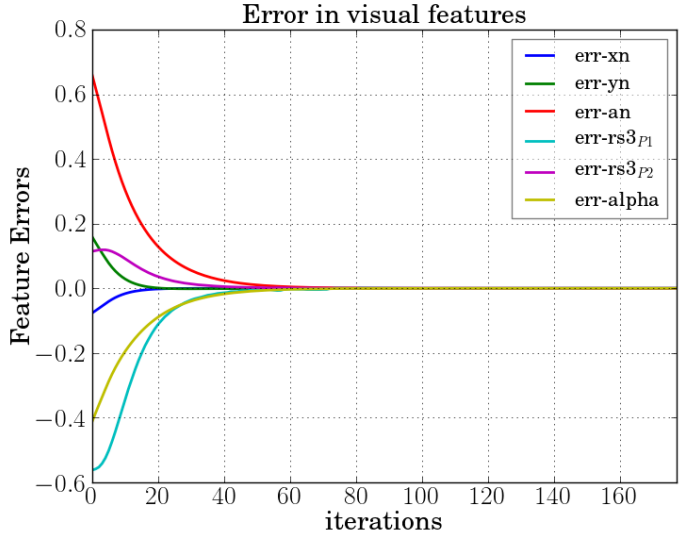
In the first set, we used a reduced set of degrees of freedom of our experimental platform, specifically only the 3D translations and rotation around the optical axis. This set of experiments will allow us to see if there is any undesired behaviour arising due to i) assumptions made in the theoretical developments and ii) the inclusion of pixel intensities in photometric moments. Also, this is the same subset of dof utilised in KBVS [Kallem 07].



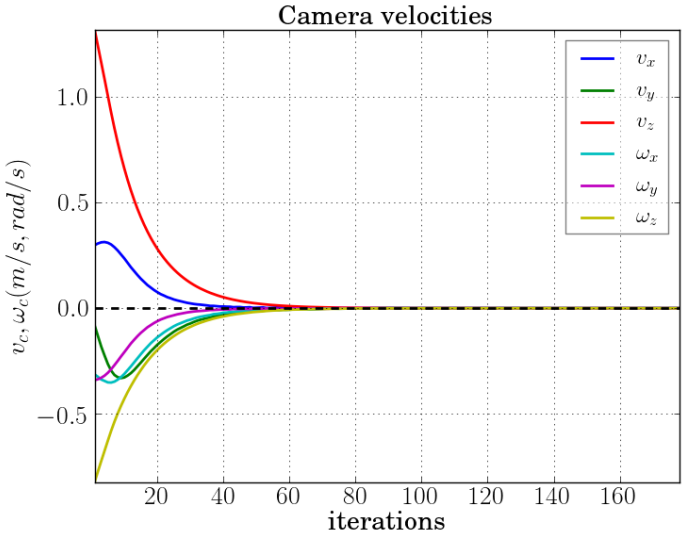
(a) Errors with $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}\left(\mathbf{s}(t), \hat{Z}^*\right)$



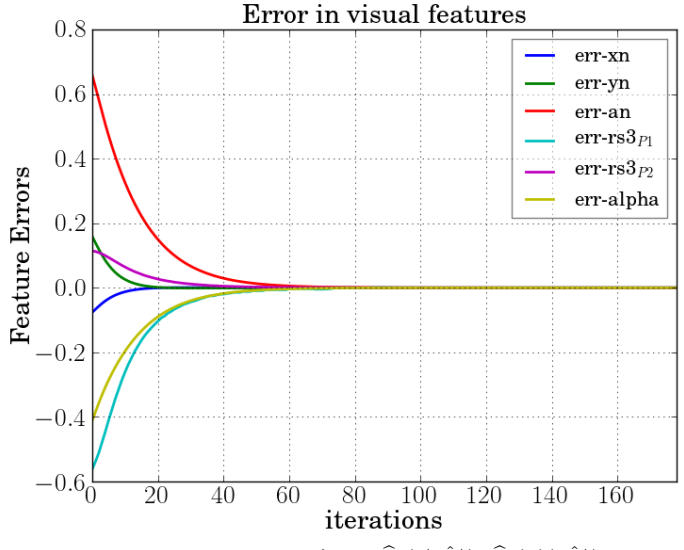
(b) Camera velocities with $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}\left(\mathbf{s}(t), \hat{Z}^*\right)$



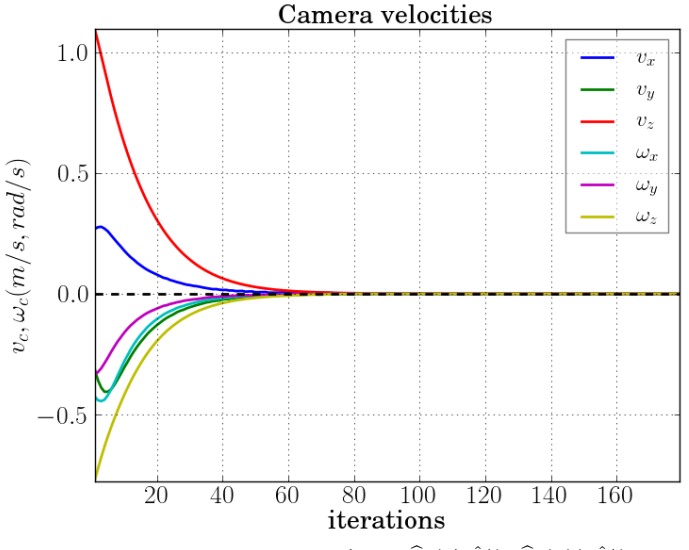
(c) Errors with $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*)$



(d) Camera velocities with $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*)$



(e) Errors with $\widehat{\mathbf{L}}_{\mathbf{s}} = \frac{\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*) + \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), \hat{Z}^*)}{2}$



(f) Camera velocities with $\widehat{\mathbf{L}}_{\mathbf{s}} = \frac{\widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*) + \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), \hat{Z}^*)}{2}$

Figure 3.9 – Visual Servoing results with different configurations for the interaction matrix

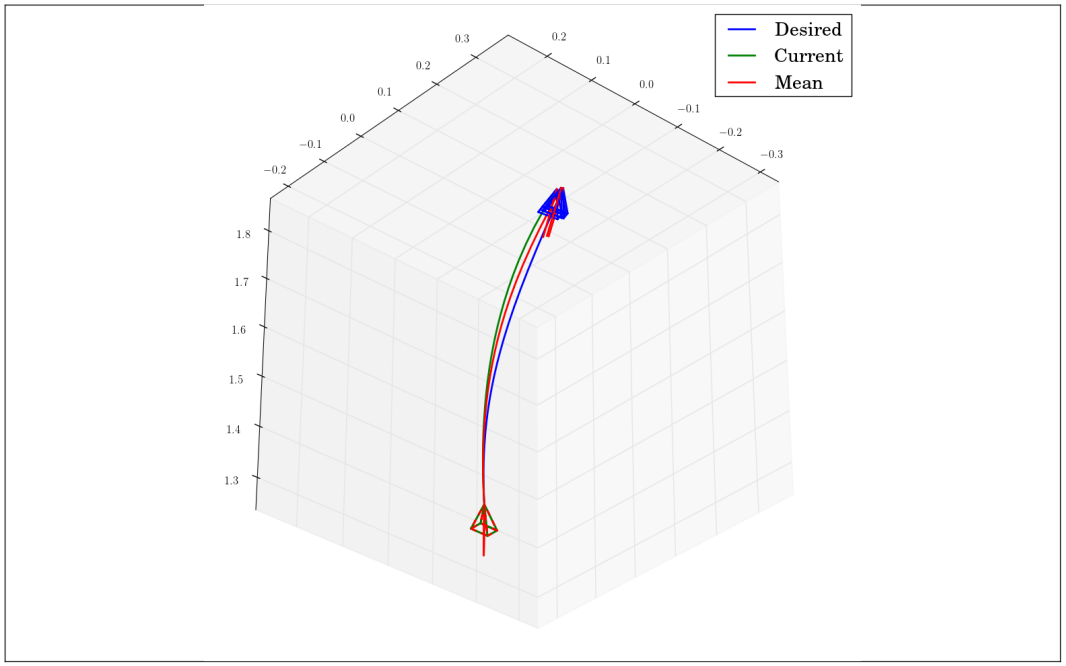


Figure 3.10 – Cartesian behaviour under different control laws

This choice therefore will ease a fair comparison between these two approaches. Let us note that the potential of using image moments in reduced dof configurations should not be underestimated. There are actuated systems (eg. SCARA robot with 3T+1R motions) and industrial tasks which exactly require such a set of reduced motions [Wang 08]. For this set of experiments, the partial velocity screw is denoted $\mathbf{v}_c = (v_x, v_y, v_z, \omega_z)$ and is given by the classic control law:

$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{-1}(\mathbf{s} - \mathbf{s}^*) \tag{3.35}$$

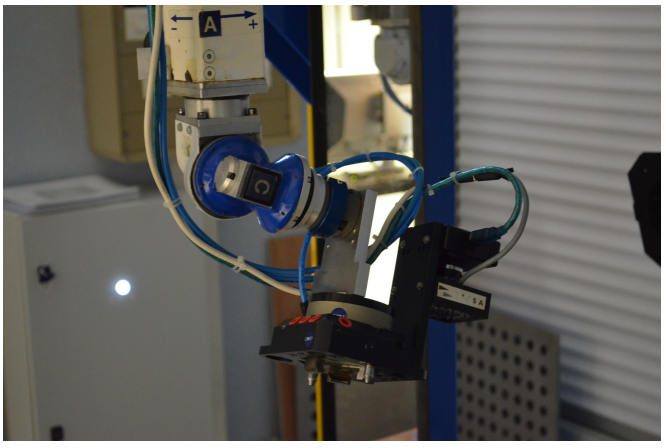


Figure 3.11 – Afma6 6DOF robotic platform equipped with a camera used in experimental validation

where the visual features $\mathbf{s}$ are based on the photometric moments developed in this chapter.

$$\mathbf{s} = (x_n, y_n, a_n, \alpha) \tag{3.36}$$

$x_n = x_g a_n$, $y_n = y_g a_n$, $a_n = z^* \sqrt{a^*/a}$. $a = m_{00}$ is the photometric area. $x_g = m_{10}/m_{00}$ and $y_g = m_{01}/m_{00}$ are the centre of gravity along the coordinate axes. Let us note that the columns of the interaction matrix now correspond only to the degree of freedom that is controlled (the first 3 columns are related to the translational motions and the last column is related to rotation around the optic axis). A nice analytical form can easily be obtained (refer to Equations (2.52) and (2.62)).

$$\mathbf{L_s} = \begin{bmatrix} -1 & 0 & 0 & y_n \\ 0 & -1 & 0 & -x_n \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \tag{3.37}$$

We recall that our features and their interaction matrices depend on photometric moments and not the binary moments like previously. If this feature set is used to control a 4-dof system (like the SCARA robot for instance), the resulting control law will be globally asympotically stable since $\mathbf{L}_s$ is always of full rank 4 and hence $\mathbf{L_s}\widehat{\mathbf{L}}_\mathbf{s}^{-1} = \mathbf{I}_4$ when $\hat{Z}^* = Z^*$ (Let us note that Z^* intervenes indirectly through x_n and y_n and thus in $\mathbf{L_s}$ and $\widehat{\mathbf{L}}_\mathbf{s}$). Also, please see our detailed discussion on stability in Chapter 1 in Section 1.1.2.

3.5.1.2 Experiment with $\mathbf{L}_\mathbf{s}(\mathbf{s}(t), \hat{Z}*)$

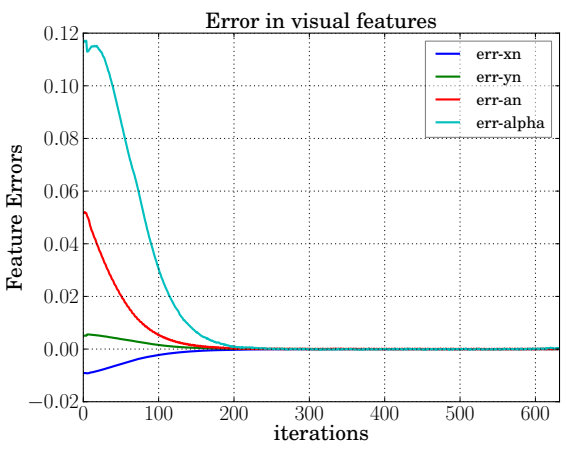
For this experiment, the current interaction matrix is chosen with depth approximated with its value at the desired position $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}(t), \hat{Z}*)$. A gain of $\lambda = 1.5$ was used. The original digital model of the planar target placed in the scene consisted of only 10 gray levels (adapted from [Flusser 09]). The desired image is shown in Figure 3.12(b). The initial pose is chosen such that the image in 3.12(a) is observed by the camera. Let us note that in this case, there is a small violation of the zero border assumption (ZBA) due to the small parts of the object that ae outside of the image in Figure 3.12(a). In fact, the background that appears black is not strictly 0 since we do not perform any segmentation or image processing to exclude those pixels. Since the moments are global features computed over the whole image, the contribution of these pixels (even if minor) are not excluded from this computation. A total displacement of $[-3.15cm, -3.99cm, -11.72cm, -27.1^\circ]$ is necessary to converge to the desired pose. A very satisfactory decrease of the errors is obtained as shown in Figure 3.12(c). The generated control velocities are shown in Figure 3.12(d). The camera follows a very straight-forward trajectory to the goal as seen from Figure 3.12(e). These results validate the correctness of the modelling of the proposed photometric moments, as well as the visual features used.

3.5.1.3 Experiment with $\widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}^*, \hat{Z}*)$

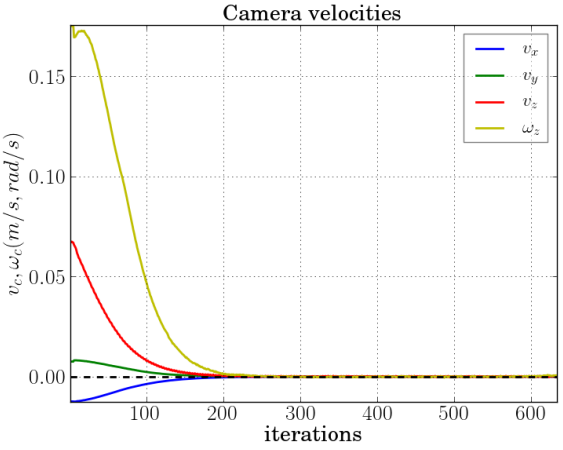
In the previous experiment, we presented results using the updated interaction matrix, which is recommended for the reduced DOF experiments. However, it is also possible to use the interaction matrix at the desired configuration, $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}^*, \hat{Z}*)$ as shown here.

(a) Camera view at initial robot pose

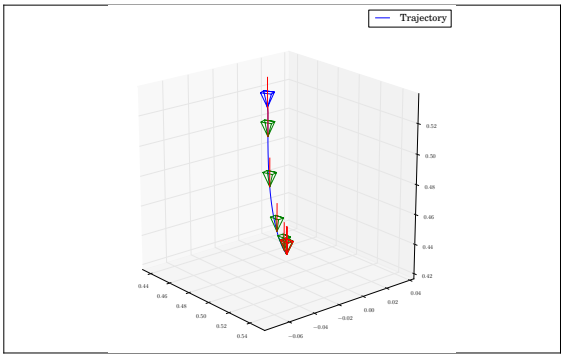
(b) Camera view at desired robot pose



(c) Visual feature errors



(d) Camera view at desired robot pose



(e) Spatial camera trajectory

Figure 3.12 – Servo results for Experiment 3.5.1.2

Also, in these experiments, the zero border assumption is slightly violated at the beginning. Let us note that the lighting conditions were kept deliberately suboptimal and not controlled during the experimentation. A constant gain of $\lambda = 0.2$ was employed. For this experiment, an initial pose far from the desired pose was chosen (See Figs. 3.13(a) and 3.13(b)). The camera displacement to realize was $[-10cm, 8cm, -25cm, 35°]$. The initial robot pose was chosen such that a portion of the target in the scene is occluded in the

camera field of view. In spite of the partial occlusion in the initial image, large displacement and approximated depth, the visual features succeeded in converging to the desired values. In principle, a perfect exponential convergence should be expected only when the current interaction matrix is employed and the depths are estimated perfectly. The final accuracy of the positioning was observed to be $[-1.09mm, -5.22mm, 2.39mm, 0.03°]$. The time evolution of the errors in the moment features and the applied camera velocities are shown respectively in Fig. 3.13(c) and Fig. 3.13(d). Further, during the servo, a satisfactory camera trajectory was obtained. This spatial trajectory is shown in Fig .3.13(e).
Servoing results with irregular shapes : The purpose of this experiment was to test if the servoing was dependent on the target shape. So, we cut a graffiti texture randomly along all its borders to an irregular shape. This is a non-rectangular texture as opposed to the previous case. Our contention is that this will be representative of nonsymmetric geometrical shaped targets. The same experimental settings as for 3.5.1.3 were used. The visual features converge to the desired values driving the robot to the desired pose. The obtained results are presented in Fig.3.14. From Figs. 3.14(a) and 3.14(b), we can clearly perceive a difference in lighting between the initial and final images. However, this did not affect the convergence of the visual servoing. The final accuracy of the positioning from the robot odometry was $-1.17mm, -4.52mm, -1.31mm, 0.09deg]$. As we can observe, the errors in translations are of the order of few mm while the rotation error is even lesser than 1 degree. We can also observe that the camera trajectory is not a straight-line at the beginning of the servo (see Fig. 3.5.1.3). First, a portion of the texture is occluded from the initial camera view. In this setting, there are a subset of intensities present in the desired image which are missing from the initial image and during the initial stages of the servo. Such phenomena are not accounted for in our model and also violates the ZBA. This explains the non-optimal camera spatial trajectory during the initial iterations. However, such phenomena of missing intensities occur especially for large displacements (such as the one we have chosen) and reduce as the system is driven to the desired configuration.

3.5.2 Visual Servoing with Photometric Moments in 6-DOF

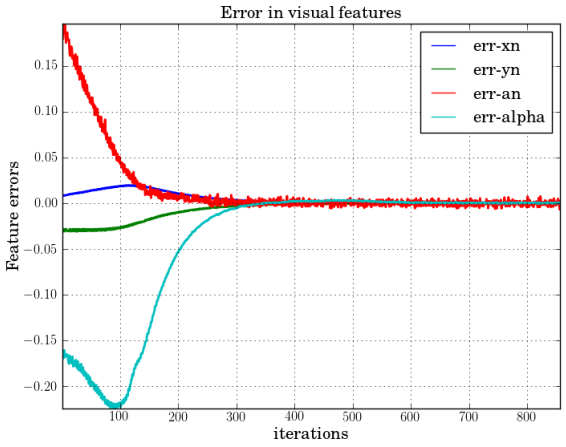
The classical visual servoing control law $\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{+}(\mathbf{s} - \mathbf{s}^*)$ is adopted with the visual features $\mathbf{s} = \left(x_n, y_n, a_n, rs_{3_{p1}}, rs_{3_{p2}}, \alpha\right)$ computed from photometric moments. The first 3 features x_n, y_n, a_n and the last feature α are the same set of features used for the experiments with reduced dof. The visual features $rs_{3_{p1}}$ and $rs_{3_{p2}}$ (as proposed originally in [Tamtsia 13b]) are computed with TRS invariants using the shifted moments (with shift angles at $0°$ and $90°$) respectively. As explained in the previous chapter and also as observed in practice from the results, this choice resulted in a satisfactory behaviour. In the following experiments, we used the coarse value of $\hat{Z}* = 0.8m$ and a constant gain of $\lambda = 0.8$. The desired image is first learnt in an offline step for task specification. The below experiments allow to test the visual servoing with complex motions requiring all the degrees of freedom.

3.5.2.1 Servoing Experiment I

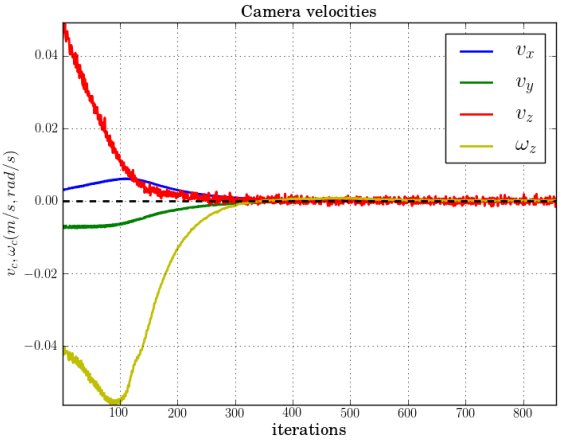
The robot is steered to a distant initial pose such that ${}^{c*}\mathbf{R}_c = [20.00°, -15.97°, -16.20°]$ and ${}^{c*}\mathbf{t}_c = [28.41cm, 29.49cm, -14.28cm]$. The initial pose is such that the camera image

(a) Camera view at initial robot pose

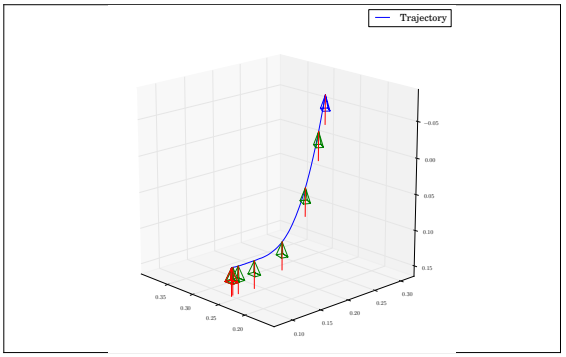
(b) Camera view at desired robot pose



(c) Errors in visual features $\mathbf{s} - \mathbf{s}*$



(d) Control velocities



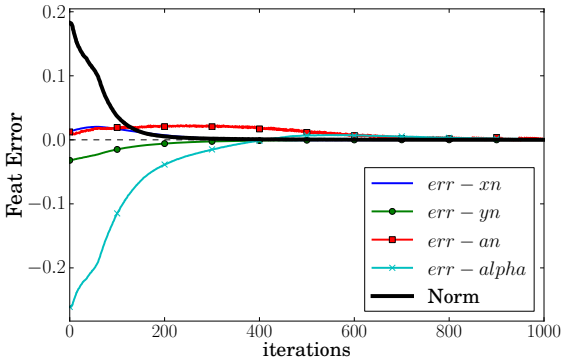
(e) Spatial camera trajectory

Figure 3.13 – Visual servo experimental results pertaining to Experiment 3.5.1.3

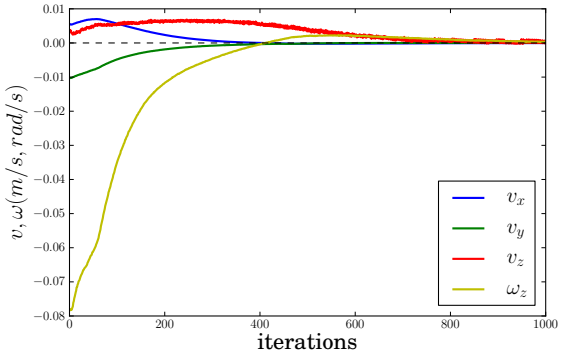
and the target planes are not parallel to each other. The image acquired from this initial pose is shown in Fig3.15(b). Such a task would involve significantly large motion in all the degrees of freedom to regulate the task error to 0. The target is a planar graffiti image commonly used in computer vision research. The interaction matrix was computed with the current feature values but with an approximated desired depth, that is, $\widehat{\mathbf{L}}_{\mathbf{s}} = \widehat{\mathbf{L}}_{\mathbf{s}}(\mathbf{s}(t), \hat{Z}*)$. This choice of the current interaction matrix with approximated depth was discussed in

(a) Camera view at initial robot pose

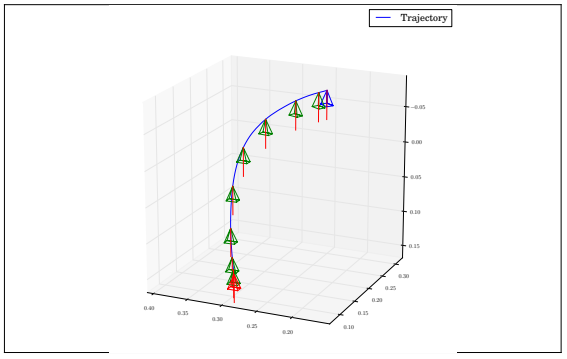
(b) Camera view at desired robot pose



(c) Errors in visual features $\mathbf{s} - \mathbf{s}*$



(d) Control velocities



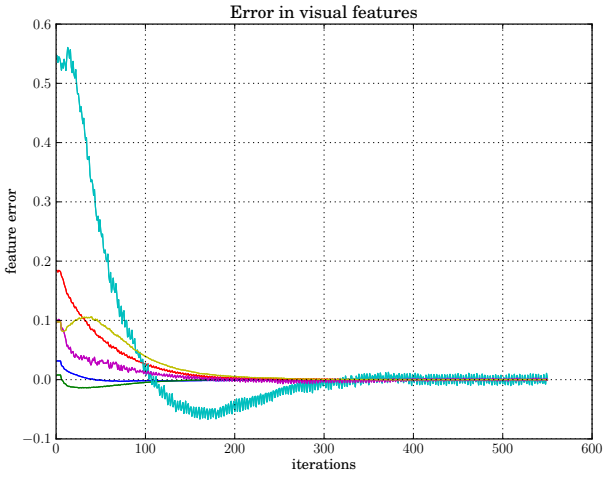
(e) Robot Pose

Figure 3.14 – Visual servo experimental results pertaining to Experiment 3.5.1.3

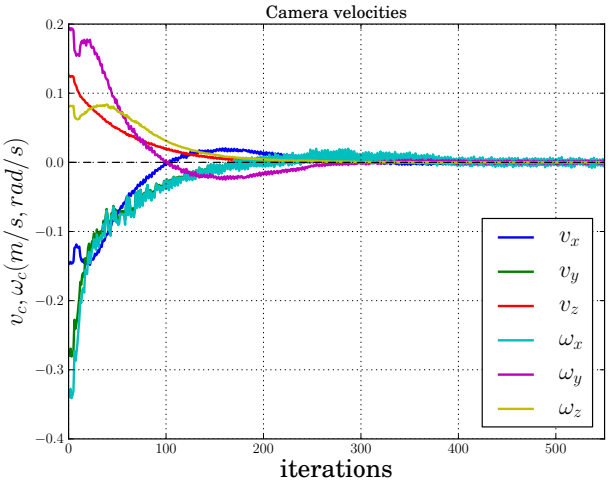
Section 1.1.1.1. The results from this experiment are shown in Fig 3.15. The errors in the features decrease to zero although not perfectly exponential. But this was not surprising but in fact consistent with the theory (and simulation) where the current features were used and depth was approximated. The visual feature rs_3 using shift point p_1 has a numerical range that is higher than the other features and hence the error in this feature appears prominently than the rest of the feature errors. Evidently, there appears an overshoot of the error in this feature beyond zero but eventually converges to zero. This overshoot means that the camera has rotated more than the necessary rotation around x to achieve the task.

(a) Image learnt at desired pose

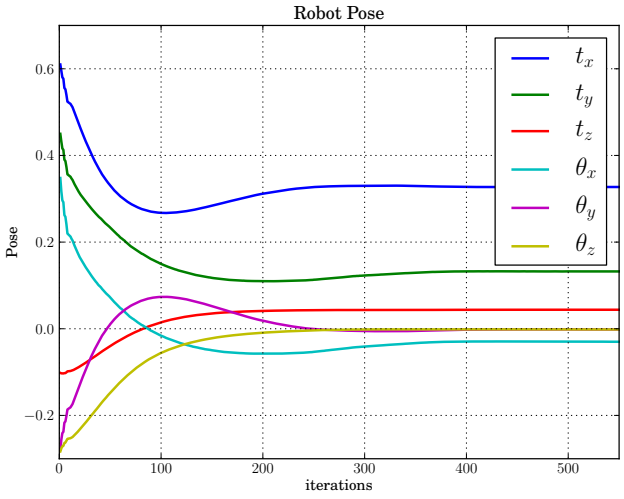
(b) Image at initial pose



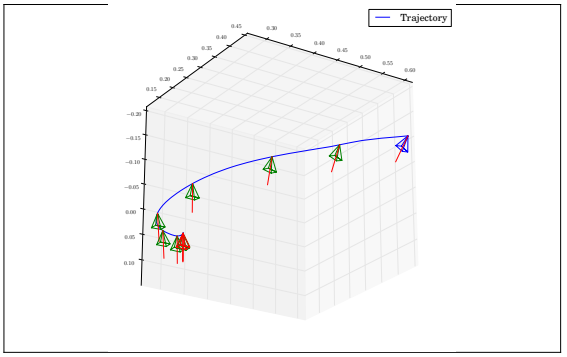
(c) Error in visual features



(d) Camera velocities



(e) Robot pose



(f) Cartesian Trajectory

Figure 3.15 – Experimental Results from Afma6 gantry robot for Experiment 3.5.2.1

But this deviation is not so large and does not affect the convergence of the servoing. The final positioning error $[0.41mm, -1.78mm, 0.17mm, -0.13°, -0.02°, -0.02°]$. The condition numbers of the interaction matrix varied between 10.8 to 11.7, which means the system is well-conditioned and explains the convergence.

Also, the experimental conditions were kept deliberately suboptimal. It should be remembered as well that an image acquired from the camera is a function of the spectral sensitivity of the photoreceptor element, spectral density of the illumination (light sources) and how the material patch reflects the wavelength of the incident illumination (which in turn depends on the relative position of the camera and the lighting sources). In effect, some almost imperceptible variations were observed in the brighter part of image due to the interaction between the lighting and the material in the scene, which explains the noise in the feature errors. Since these effects are not dealt with in the modelling, we recommend not to use shiny or bright materials in the scene.

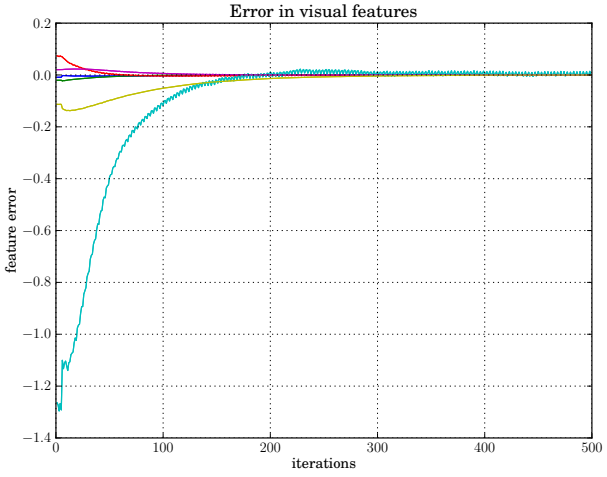
Figure 3.16 – Photograph of side-view showing non-planarity on one side of the target

3.5.2.2 Servoing Experiment II

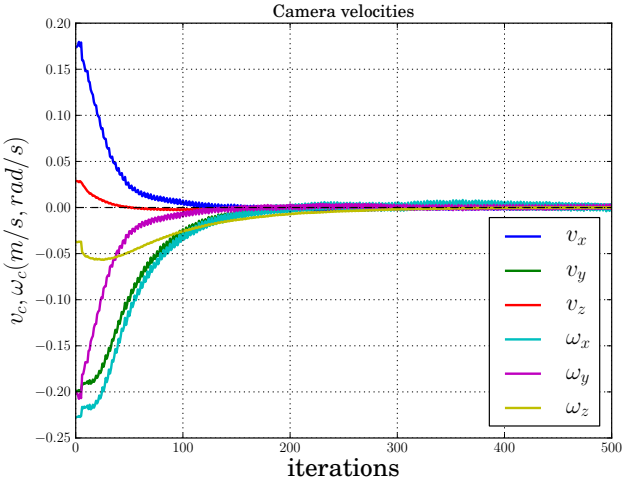
The initial and desired poses were apart by ${}^{c*}\mathbf{R}_c = [25.27°, 14.3°, 12.9°]$ in rotation and by ${}^{c*}\mathbf{t}_c = [-23.60cm, 34.44cm, 10.28cm]$ in translation. A different initial pose was chosen this time but also with a large displacement as before. The same visual features as in the previous experiments on the real robot were used. Also, a different target (See Figs.3.17(a), 3.17(b)) was used to study any changes in the behaviour of the control law. Typically, there are large areas in this target (that appear black) that do not contribute to the information gathered through moments. Nevertheless, we observed that the servoing succeeded with no issues. Further, we now use the control law in Equation (1.23) which uses a mixed configuration for the computation of the interaction matrix. This choice was advocated in [Malis 04] since it better estimated the camera displacement and exhibited a satisfactory behaviour. That is, for this case, $\widehat{\mathbf{L}}_\mathbf{s} = \frac{\widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}^*, \hat{Z}^*) + \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}(t), \hat{Z}^*)}{2}$. The results obtained are presented in Fig. 3.17. The velocity profiles (shown in Fig.3.17(d)) were satisfactory and the visual feature errors (See Fig.3.17(c)) exhibited equally good convergence as in the previous case. The condition numbers for $\mathbf{L}_\mathbf{s}$ varied between 60.3 to 96 which is higher than in 3.5.2.1 but is acceptable.

(a)

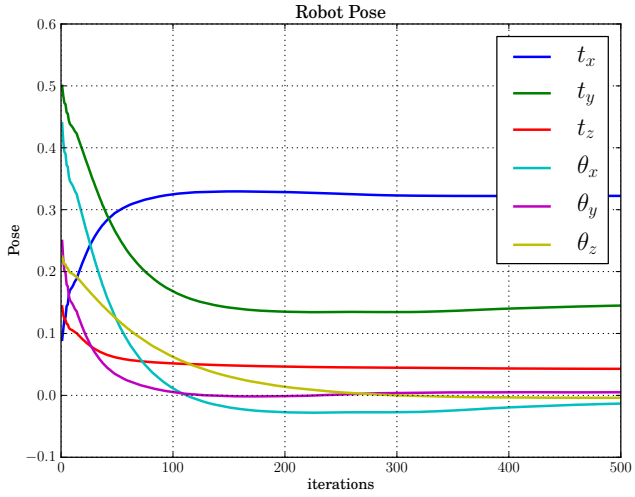
(b)



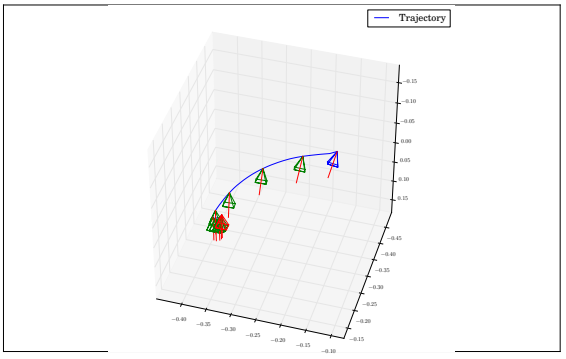
(c)



(d)



(e)



(f)

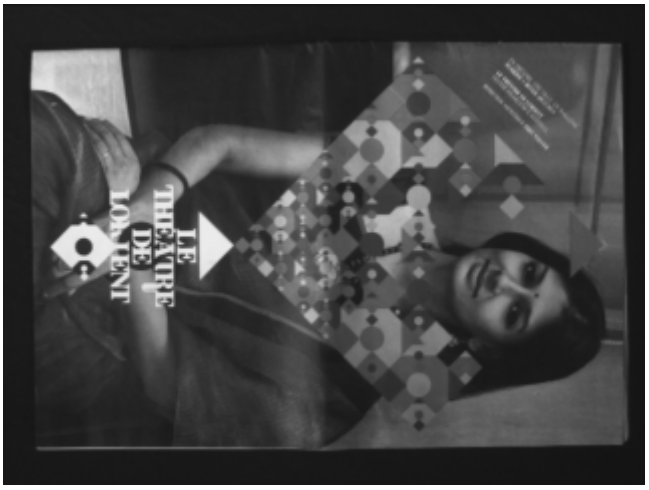
Figure 3.17 – Experimental Results from Afma6 gantry robot for Experiment 3.5.2.2

3.5.2.3 Servoing Experiment with slight non-planarity

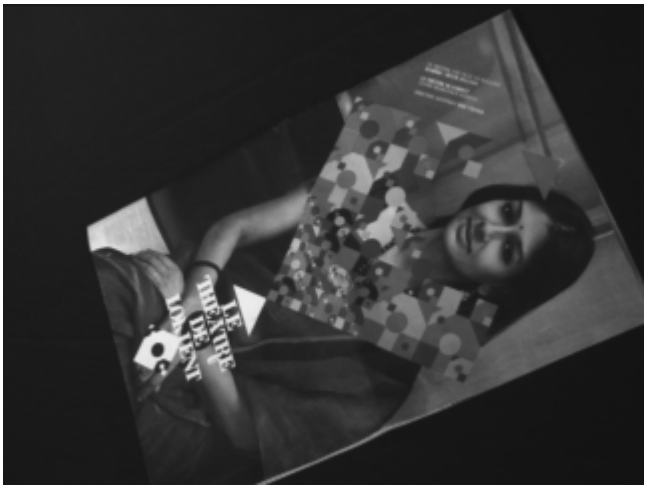
A textured image which shared significant areas with the surrounding was chosen. This experiment was conducted with slight non-planarity in the scene structure as shown in Figure 3.16). The initial and desired poses were such that ${}^{c*}\mathbf{R}_c = [21.36°, -18.54°, 21.30°]$ and ${}^{c*}\mathbf{t}_c = [20.41cm, 34.45cm, 1cm]$ for an overall translation clearance of $56cm$ to attain the goal pose. The interaction matrix was computed with the current feature values but with an approximated desired depth, $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}(t), \hat{Z}*)$ as in Experiment 3.5.2.1. The servo succeeds and the robot reaches the desired pose with a final error in positioning was equal to $e_M = -3.66mm, 7.35mm, -0.87mm, 0.52°, 0.28°, 0.19°]$. The finally accuracy is not very satisfactory in this case which is probably due to the non-planarity introduced in the scene. To improve this situation, the photometric moments method could be combined with the pure luminance feature by sequencing the two methods. From the error profiles, it is clear that the first few iterations are not optimal. In fact, some of the scene areas are not visible in the image and then reappear again as servo progresses. (See Fig.3.19 (c)) from the 33rd servo iteration. The servo was however stable. It can be seen from Fig 3.19(d) that the variations in the condition numbers are small ranging from roughly 23 to 46. The camera trajectory is not very deviant and seems satisfactory as well. The depths are very coarsely approximated in this experiment. When roughly measured, the real depth was more than $1m$ while the approximation was at $Z^* = 0.8m$ as usual. The experiment converged without any problems and thus in agreement with previous results that IBVS methods are robust to depth approximations and small non-planarities in the viewed scene. But the presence of specular reflections or strong lighting fluctuations can cause the method to fail.

3.5.2.4 Servoing Experiment IV

In this section, we present the experimental results obtained from a robot different from the previous one. This one, the Viper850 is also endowed with 6DOF and is shown in Fig 3.20. The desired and initial poses are chosen such that a translational displacement ${}^c\mathbf{t}_{c*} = [26.0cm, 13.25cm, -8.00cm]$ and a rotational displacement of ${}^c\mathbf{R}_{c*} = [7.0°, -21.16°, 17.10°]$ is required. The images acquired from the desired and starting poses are shown in Figs 3.21(a) and 3.21(b) respectively. The classical control law was used with the interaction matrix at the desired configuration, $\widehat{\mathbf{L}}_\mathbf{s} = \widehat{\mathbf{L}}_\mathbf{s}(\mathbf{s}^*, \hat{Z}*)$. With this choice here, we have an opportunity to highlight the nice decoupled structure that is exhibited by the interaction matrix in the case of the proposed photometric moments. The depth was measured again for the experiment with this robot and approximated roughly at $Z* = 0.5m$. A constant gain of $\lambda = 0.2$ was used.



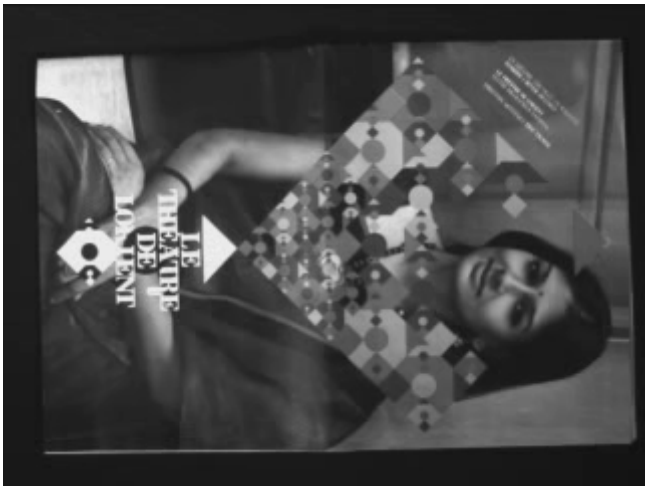
(a) Desired image learnt offline used for task specification



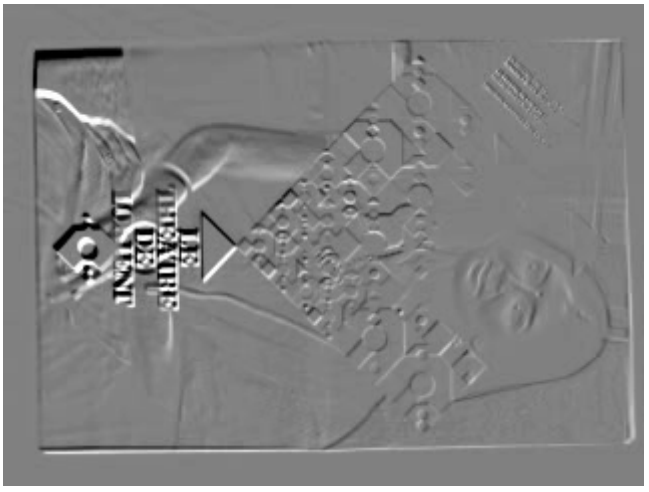
(b) Image acquired from initial pose of the robot

(c) Image acquired at $t \approx 33$ iters

(d) Difference image at $t \approx 33$ iters

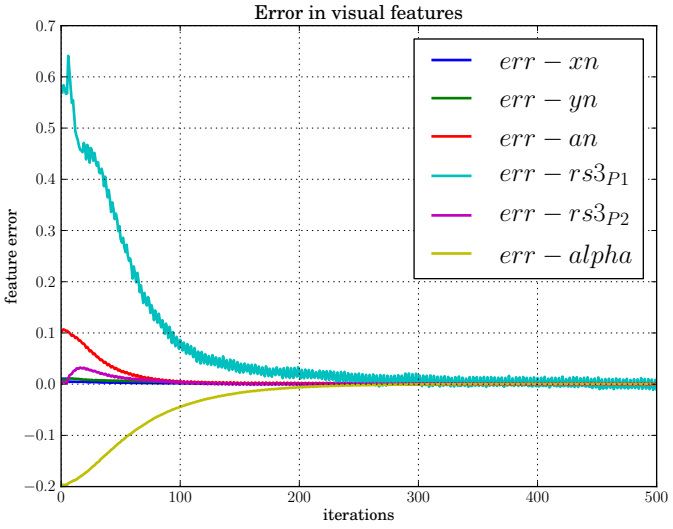


(e) Image acquired at $t \approx 120$ iters

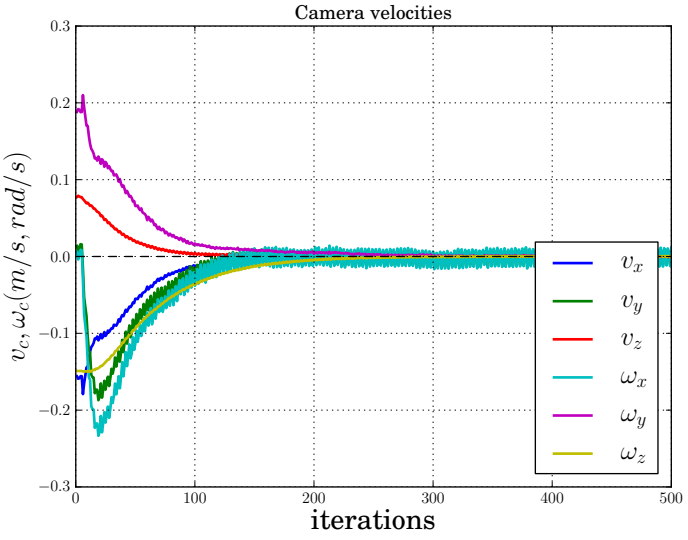


(f) Difference image at $t \approx 120$ iters

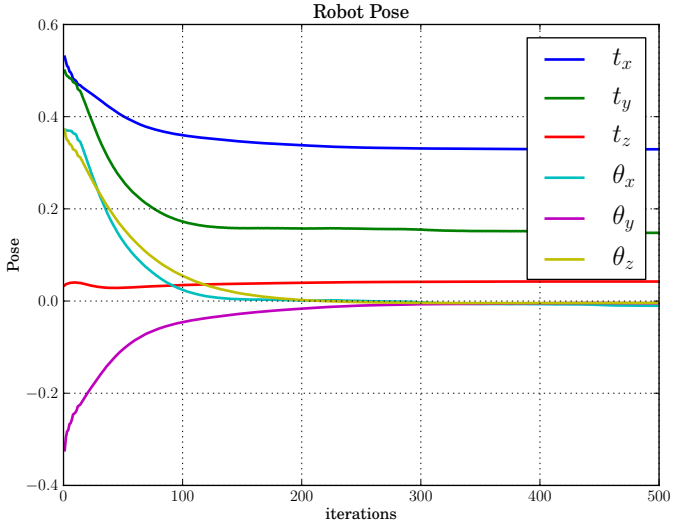
Figure 3.18 – A sampling of images acquired during the Visual Servoing in Experiment 3.5.2.3



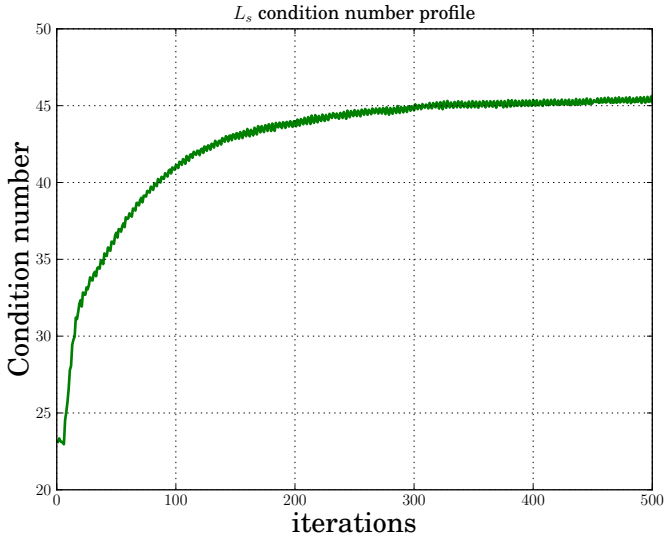
(a) Error in visual features



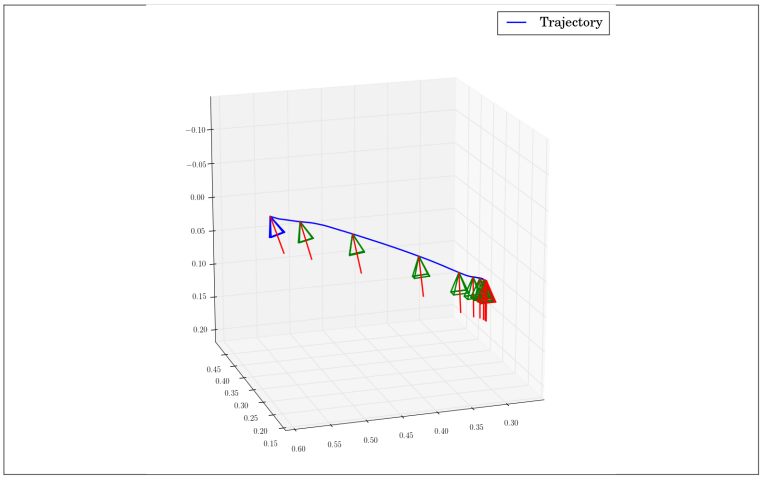
(b) Camera velocities



(c)



(d) Evolution of the condition number of the interaction matrix



(e)

Figure 3.19 – Visual Servoing on a Afma6 gantry robot using Photometric Moments with small non-planarity and large displacement for Experiment 3.5.2.3

Figure 3.20 – Photograph of Viper850 robot used in experimental validation

For this experiment, the following interaction matrix was obtained.

$$\mathbf{L}_{s*} = \begin{bmatrix} -1 & 0 & 0 & 0.0037 & -0.9234 & -0.0202 \\ 0 & -1 & 0 & 0.8677 & -0.0037 & -0.0098 \\ 0 & 0 & -1 & 0.0303 & 0.0148 & 0 \\ 0 & 0 & 0 & -0.9580 & -3.0864 & 0 \\ 0 & 0 & 0 & 0.4925 & 0.1484 & 0 \\ 0 & 0 & 0 & 0.0552 & -0.0071 & -1 \end{bmatrix} \tag{3.38}$$

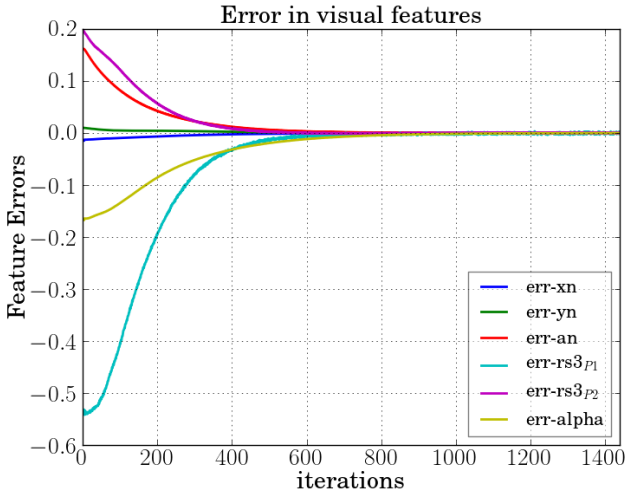
Further, this had an excellent condition number of 10.8401, which is good from the stability point of view. It can be observed from Fig 3.21(c) that the decrease in errors is highly satisfactory. We recall that only the interaction matrix at the desired configuration and approximate depth was employed. The generated velocity profiles are also smooth as shown in Fig.3.21(d). Clearly, the camera spatial trajectory is close to a geodesic as shown in Figure 3.21(e). Further, an accuracy of $[-0.56mm, -0.08mm, 0.14mm]$ in translation and $[-0.01°, 0.04°, -0.03°]$ in rotation was obtained. Thus, we find that the accuracies are also very satisfactory.

3.6 Conclusions

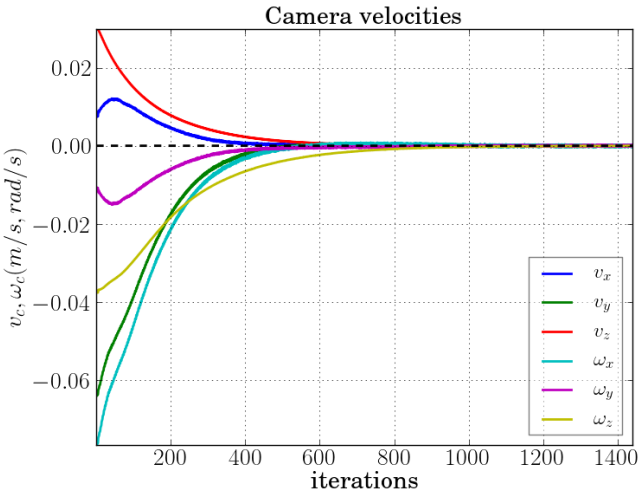
In this chapter, photometric image moments were introduced as new visual features for image-based visual servoing (IBVS). The interaction matrix was developed in closed-form for the proposed photometric moments. Photometric moments avoid any spatial segmentation steps and reduce the image processing to a simple and systematic moments computation on all the image plane. In comparison with existing methods (based on geometric features or binary moments), photometric moments can be used to servo planar complex textured objects. Image moments based visual features leverage the invariant

(a) Image learnt from desired pose

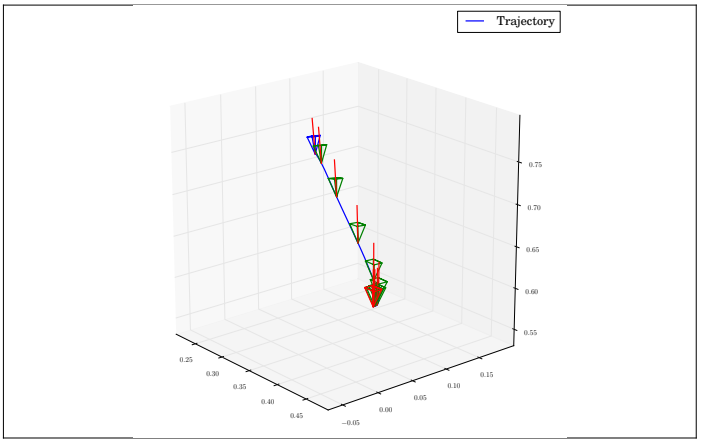
(b) Image acquired from initial pose of the robot



(c) Error in visual features when using rs_3



(d) Camera velocities when using rs_3



(e) Cartesian trajectory

Figure 3.21 – Experimental Results from Viper850 Robot using Photometric Moments for Experiment 3.5.2.4

properties of moments resulting in decoupled control law. Photometric moments, although based on intensity, can be used without any modification of the classical control laws. With the proposed visual features, a simple classical first order control law suffices to perform 6-dof control. Simulation studies have been presented to compare the visual servoing characteristics obtained with several interaction matrix configurations. We have shown experimentally that photometric moments can be used for controlling either 4-dof or all the 6-dof of a real robotic system. They perform well even with large displacements since they possess a large convergence domain.

Although results have shown robustness to a certain degree even with violations in the zero border assumption, more theoretical work is required to remove this restrictive assumption made in the interaction matrix developments. This goal is achieved in the next Chapter.

Chapter 4

Handling Extraneous Image Regions in Photometric Moments-based Visual Servoing

4.1 Introduction

Visual servoing based on photometric data is of great interest since it helps to eliminate image processing and visual tracking steps. It was shown in the previous chapter that it is possible to perform visual servoing with photometric moments without using any visual tracking and image processing steps. A practical issue in such methods is the change in the image resulting from the appearance and disappearance of portions of the scene from the camera field-of-view during the visual servo. To understand this in perspective, let us consider the time variation of the visual features.

$$\dot{\mathbf{s}} = \mathbf{L_s v_c} + \frac{\partial \mathbf{s}}{\partial t} \tag{4.1}$$

If we consider a physically immobile target, then the variations in the visual features are only due to the camera motion effected with the control law. So, the Equation (4.1) reduces to the classic and simple form $\dot{\mathbf{s}} = \mathbf{L_s v_c}$ [Chaumette 06]. When there are changes which occur in the image due to effects which are not accounted for, then it means the second term $\frac{\partial \mathbf{s}}{\partial t}$ in Equation (4.1) is not equal to 0. As shown in this chapter, such unmodelled changes introduce potential disturbances in the control or at worst, leads to convergence problems in the visual servoing. That is why it is important to solve this problem. Ideally, the visual servoing control law should be made immune to the disappearance and reappearance of scene portions from the current image $\mathbf{I}(t)$ to the future ones $\mathbf{I}(t+N)$. Interestingly, this problem was foreseen and explicitly stated more than a decade earlier in [Chaumette 04].

The chapter is organized as follows: First, existing visual servoing literature was examined to see if this problem had been solved before. Problems of somewhat similar

nature existed for geometric feature based methods and a brief treatment of those methods is first provided in Section 4.1.1. Section 4.2 presents the modelling aspects to obtain the interaction matrix of weighted photometric moments. A brief detour to explain the selection of the spatial weighting functions and its effects on the interaction matrix is discussed. Section 4.3 discusses the properties for the most commonly used visual features. Finally, validation results are presented in Section 4.4.

4.1.1 State of the Art

Previous works on IBVS which use tracked geometric primitives dealt with the appearance and disappearance of visual features from the camera field of view and loss of the image signal during the servo. We will see in due course that none of the existing solutions are apt for the case of photometric moments, since almost all of them explicitly require that a set of points are tracked during the visual servoing.

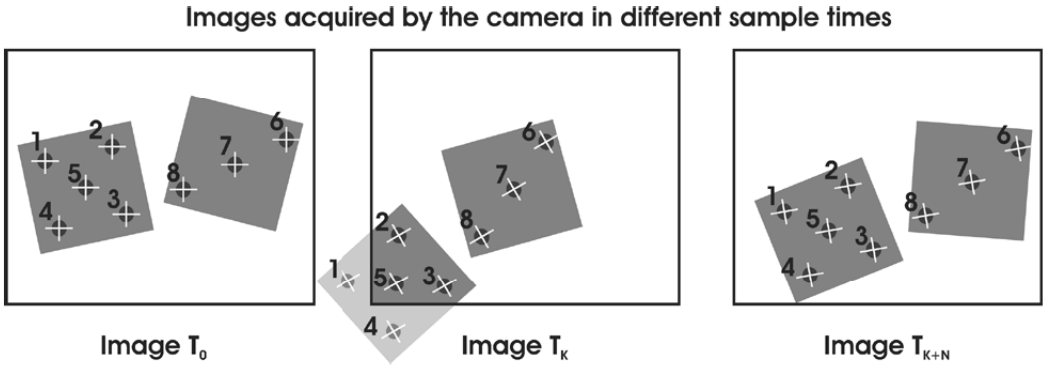


Figure 4.1 – The problem of point features leaving the field of view illustrated in [GarcíaAracil 05]

Made use of extensively in the previous chapters, we know that the control law in IBVS is written as

$$\mathbf{v}_c(t) = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{+} \left(\mathbf{s}(t) - \mathbf{s}*\right) \tag{4.2}$$

When one or more visual features leave the camera field of view, this control law loses continuity. To counter this, many solutions were proposed in literature. These solutions were based on defining an activation function that neutralizes the effect of the missing visual features by smoothly suppressing them from the control. In these techniques, the control law was modified to have this general structure:

$$\mathbf{v}_c(t) = -\lambda \left(\mathbf{D}\widehat{\mathbf{L}}_{\mathbf{s}}\right)^{+} \mathbf{D} \left(\mathbf{s}(t) - \mathbf{s}*\right) \tag{4.3}$$

where $\mathbf{D} = diag(w_1, w_2, \ldots, w_k)$, $0 \leq w_i \leq 1$, considering that the interaction matrix is of dimension $k \times n$.

In works such as [Comport 03], [Marchand 04] and [Comport 06], the activation was inspired by statistical estimation techniques. The weights w_i are determined by using an influence Function (Huber's or Tukey's functions).

In [GarcíaAracil 05], the weights in $\mathbf{D}$ are fixed based on the location of the feature points in the image plane. The image plane is demarcated to safe and border zones (see Fig. 4.2). The weight is a smooth bell-shaped function in the safe zones while being zero at the border zones. The weighting function at pixel (x, y) is given by

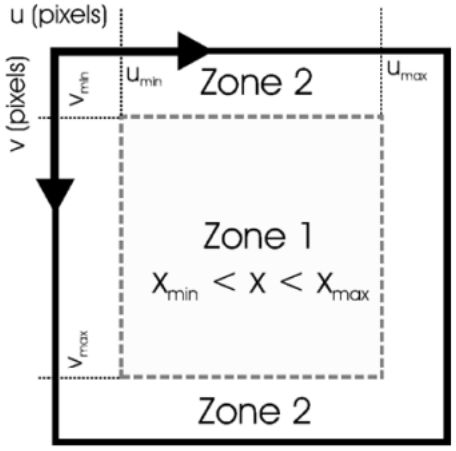


Figure 4.2 – Predicting feature weights from zones in the image plane (Figure reused from [GarcíaAracil 05])

$$w(x, y) = w(x).w(y) \text{ where} \tag{4.4}$$

$$w(x) = \begin{cases} \exp^{-\left(\frac{(x-x_M)^{2n}}{(x-x_{min})^m(x_{max}-x)^m}\right)} & \text{if } x_{min} \leq x \leq x_{max} \\ 0 & \text{if } x = \{x_{min}, x_{max}\} \end{cases} \tag{4.5}$$

with $x_M = (x_{min} + x_{max})/2$ is at the centre of the permitted zone. These works assume that there are sufficient visual features in the field of view and consequently a full rank interaction matrix. In the worst case, if this assumption does not hold, interaction matrix will lose rank. This will affect the pseudo-inverse operator which becomes discontinuous when the rank of this matrix changes. This problem was later resolved with the introduction of a new matrix inversion operator in [Mansard 09] (see for example point features leaving the field of view in Figure 4.2). Using this continuous operator, the control law can be developed as:

$$\dot{\mathbf{q}} = -\lambda \mathbf{J}^{\oplus \mathbf{D}} \mathbf{e} \tag{4.6}$$

$\mathbf{J}^{\oplus \mathbf{D}}$ is the continuous inverse wrt a given activation matrix $\mathbf{D} \in \mathbb{R}^{k \times k}$.

In [Folio 08], a general framework is presented to deal with loss of the image signal during a finite interval during a visual servoing task. In this framework, when the image signal is lost, the control loop uses a set of predicted visual features to counter the effect of loss of signal. Both depth and the visual features at time t are predicted based on history of their values until the previous time instant and knowledge of the dynamics of a set of image points. The results have been demonstrated on a mobile robot with 3dof. This framework is briefly summarized in detail in Frame (2). Recently, an extension to this framework that deals with total image loss was presented in [Durand 13]. The idea was validated with simulated point features with a 6dof free-flying camera and assuming

that perfect depth estimates are always available. The authors argue that the framework is generic and features other than points can be used.

In [Allibert 10] and [Lazar 08], a model-predictive controller is presented where the prediction of the visual features respect the visibility constraints specified in the optimization problem. An improved *model correction step* for use in such model-predictive schemes was introduced in [Cazy 14]. A visual navigation task was considered in [Silveira 06], the scene was modelled by a set of planes which disappear and appear during the navigation. But those planes had to be visually tracked and further, a pose reconstruction was necessary.

Visibility through Path Planning Methods Along with potential issues that could occur with IBVS, [Chaumette 98] demonstrated that even though the control is performed in the feature space, the image features could still leave the field of view when a too coarse approximation of the interaction matrix is used in the control scheme. Path planning methods were developed to tackle issues related to the sensor(image) and physical constraints of the robots. A path planning planner typically generates a reference camera trajectory $\mathcal{C} = \{\mathcal{P}_i(t)\}$ starting from the initial pose $\mathcal{P}_1$ to the desired pose $\mathcal{P}_N$. This is followed by generation of image space trajectories tracked by image-based visual servoing. Constraints mentioned above are then typically injected as constraints to be respected by the reference trajectory. In [Mezouar 02], the visibility constraints are enforced using the potential field based method, also based on the location of the feature points. In order to ensure visibility of image features, [Hafez 07] formulates field-of-view constraints in feature space and injects it as constraints to an optimization problem. Another interesting approach is proposed in [Chesi 09] based on optimization of linear matrix inequalities (LMI). There are several methods and for a concise review of path planning methods, please see [Kazemi 10].

So, to the best of our knowledge, the problem of extraneous image regions remains unsolved to-date in dense visual servoing.

4.1.2 About the proposed approach

The problem is compounded in the case of dense features like photometric moments. This is because they are global features which use information from the entire image plane. We note that existing methods in the state-of-the-art are not applicable to the dense case which do not use geometric features. A solution to this problem can be looked at from different viewpoints.

- This problem can be looked at from an information filtering point of view. That is, a segmentation can be performed on the image such that the extraneous image regions are rejected. This is feasible for binary moments [Chaumette 04] [Tahri 05a] but rather complicated for complex targets. This also defeats the purpose of photometric moments, which were developed in the first place to avoid the image processing and tracking steps.

- The second solution is from the modelling point of view: to introduce a change in the modelling of the moments such that the influence of the extraneous image regions is reduced as much as possible.

Frame 2 A framework to deal with visual signal loss in visual servoing [Folio 08] [Durand 13]

This framework exploits the relation between the variations of the visual features with respect to camera velocity. Let us assume that the interaction matrix for the visual features is known in analytical form. So, we can write

$$\dot{\mathbf{s}}_j = \begin{bmatrix} \dot{s}_1 \\ \dot{s}_2 \\ \vdots \\ \dot{s}_n \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{\mathbf{s}_1(A_{pq})} \\ \mathbf{L}_{\mathbf{s}_2(A_{pq})} \\ \vdots \\ \mathbf{L}_{\mathbf{s}_n(A_{pq})} \end{bmatrix} \mathbf{v}_c = \mathbf{L}_{\mathbf{s}_j(A_{pq})} \mathbf{v}_c \tag{4.7}$$

where $\frac{1}{Z_i} = \sum_{p\geq 0, q\geq 0} A_{pq} x^p y^q$ as described in [Chaumette 04]. The assumption in this framework is that the target consists of a set of image points $\mathbf{x}_i = \{x_i, y_i\}$ projected from a set of 3D points $\mathbf{X}_i = \{X_i, Y_i, Z_i\}$ in the scene. To simplify, let us consider that a set of points from the image $\mathbf{x}$ are used for visual servoing. Then, we can write the dynamics of this system from equations of perspective projection and optic flow equations that relate the motion of the points in the image to the motion of the camera.

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{Z}_i \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{\mathbf{x}} \\ \mathbf{L}_{Z_i} \end{bmatrix} \mathbf{v}_c \tag{4.8}$$

where $\mathbf{L}_{\mathbf{x}}$ is the well-known interaction matrix of a single image point and $\mathbf{L}_{Z_i}$ links variations in the z coordinate of a 3D point with respect to camera motions, given by:

$$\begin{bmatrix} \mathbf{L}_{x_i} \\ \mathbf{L}_{y_i} \\ \mathbf{L}_{Z_i} \end{bmatrix} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \\ 0 & 0 & -1 & \frac{Z_i Y_i}{f} & \frac{Z_i X_i}{f} & 0 \end{bmatrix} \tag{4.9}$$

First, a vector consisting of image projections of the features (now, we assume a set of points as visual features) is assembled $\psi = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_P\}$. These points should belong to a specific region or domain D in the image. When the visual features are lost, the following steps are performed.

1. The dynamic system in (4.7) is solved to estimate $\hat{\psi}$. This requires a correct initialization of the depths Z_i in (4.8). Analytical solutions are possible in the case of points and in case other visual features are used, numerical methods have to be used to solve this system of differential equations [Folio 08].

2. With knowledge of $\hat{\psi}$, determine the surface parameter coefficients A_{pq} that best fit the chosen points using standard least squares methods.

3. With known A_{pq}, integrate system (4.7) to get an estimate of the visual features $\mathbf{s}_j$.

   In Chapter 3, we saw that the interaction matrix for a moment m_{pq} depends on moments of orders upto $p+q+1$. Hence, the moments m_{pq} cannot be predicted directly. In this case, it is necessary to select a set of points from an object contour in the image whose positions can be predicted. This obviously necessitates visual tracking procedures to be integrated in the visual servo loop. That is why this framework is not suited for the photometric moments case.

In this thesis, a solution to this problem is proposed from the modelling point-of-view. A preventive solution is proposed that advocates a modification in the moments definition (and naturally the visual features developed from them) and not the control law itself. The main idea put forth in this chapter is the introduction of spatial weights in the image moments formulation so that the areas where the appearance and disappearance occur (typically close to the borders) have as less influence as possible when compared to the central parts of the image during the visual servoing. During the latter stages of our re-

Figure 4.3 – Region of interest in black oval (Figure reused from [Comaniciu 03])

search, we discovered that a similar problem existed in the kernel based object tracking method proposed in [Comaniciu 03]. In this work, to handle interference from the pixels in the periphery of the region of interest to be tracked (See Figure 4.3), kernels with monotonously decreasing profiles were used around it. This is a strong hypothesis but in practice sufficient to achieve immunity from the interfering pixels. In addition, the zero border assumption (ZBA) made in Chapter 3 is naturally eliminated. We propose to use spatial weighting schemes and explicitly model their effect on the interaction matrix. Weighting function of a specific structure is introduced into the formulation of photometric image moments. Some specific properties of this weighting function are exploited in order to develop the analytical form of the interaction matrix.

4.2 Modelling of weighted photometric moments

We define the weighted photometric moments (WPM) as follows

$$m_{pq} = \iint_{\pi} x^p y^q \, w\left(\mathbf{x}\right) \, I\left(\mathbf{x}, t\right) \, \mathrm{d}x \, \mathrm{d}y \tag{4.10}$$

where $p+q$ denotes the order of the moments, $\mathbf{x} = (x, y)$, spatial points where the intensity $I(\mathbf{x})$ is measured and π denotes the compact image support. $w(\mathbf{x})$ is a spatial weighting function.

4.2.1 General case

To maintain generality, we first develop an analytical form of the interaction matrix for an abstract spatial weighting function $w\left(\mathbf{x}\right)$ and then for a specific family of functions (in

Sec.4.2.2). The only condition is that the first order spatial derivatives of this weighting function exist and are continuous everywhere in π. Furthermore, this function has to be designed so that it attributes a zero value to the pixels at the image borders. This allows to remove the zero border assumption (ZBA) introduced in the developments in Chapter 3. The initial developments are very similar to that used in deriving the interaction matrix of the photometric moments in Section 3.2. So in the following derivation, we follow the same line of reasoning as was done previously, only emphasizing the changes introduced by the weighting function, wherever appropriate.

We can obtain the interaction matrix of WPM by taking the derivative of Equation (4.10).

$$\dot{m}_{pq} = \iint_{\pi} x^p y^q \, w\left(\mathbf{x}\right) \dot{I}\left(\mathbf{x}, t\right) \, \mathrm{d}x \, \mathrm{d}y \tag{4.11}$$

where $\dot{I}\left(\mathbf{x}, t\right) = \mathbf{L_I} \mathbf{v}_c$, $\mathbf{L_I}$ being the interaction matrix of the luminance feature [Collewet 11]. After following the same developments as in Section 3.2, we obtain

$$\left.\begin{aligned}
L^{\mathtt{v_x}}_{\mathtt{m_{pq}}} &= A\, m^{\nabla x}_{p+1,q} + B\, m^{\nabla x}_{p,q+1} + C\, m^{\nabla x}_{p,q} \\
L^{\mathtt{v_y}}_{\mathtt{m_{pq}}} &= A\, m^{\nabla y}_{p+1,q} + B\, m^{\nabla y}_{p,q+1} + C\, m^{\nabla y}_{p,q} \\
L^{\mathtt{v_z}}_{\mathtt{m_{pq}}} &= -A\, m^{\nabla x}_{p+2,q} - B\, m^{\nabla x}_{p+1,q+1} - C\, m^{\nabla x}_{p+1,q} \\
&\quad - A\, m^{\nabla y}_{p+1,q+1} - B\, m^{\nabla y}_{p,q+2} - C\, m^{\nabla y}_{p,q+1} \\
L^{\omega_{\mathtt{x}}}_{\mathtt{m_{pq}}} &= -\, m^{\nabla x}_{p+1,q+1} - \, m^{\nabla y}_{p,q} - \, m^{\nabla y}_{p,q+2} \\
L^{\omega_{\mathtt{y}}}_{\mathtt{m_{pq}}} &= m^{\nabla x}_{p,q} + \, m^{\nabla x}_{p+2,q} + \, m^{\nabla y}_{p+1,q+1} \\
L^{\omega_{\mathtt{z}}}_{\mathtt{m_{pq}}} &= -\, m^{\nabla x}_{p,q+1} + \, m^{\nabla y}_{p+1,q}
\end{aligned}\right\} \tag{4.12}$$

with

$$m^{\nabla x}_{pq} = \iint_{\pi} x^p y^q \, w(\mathbf{x}) \, \frac{\partial I}{\partial x} \, \mathrm{d}x \, \mathrm{d}y \tag{4.13a}$$

$$m^{\nabla y}_{pq} = \iint_{\pi} x^p y^q \, w(\mathbf{x}) \, \frac{\partial I}{\partial y} \, \mathrm{d}x \, \mathrm{d}y \tag{4.13b}$$

where $\dfrac{\partial I}{\partial x}$ and $\dfrac{\partial I}{\partial y}$ are the gradients of the image along x and y. A, B and C are the plane parameters which depend on the relative location of the camera and scene planes. It has to be noted that the form of the interaction matrix is same as 3.17 but the terms $m^{\nabla x}_{pq}$ and $m^{\nabla y}_{pq}$ now have additional terms contributed by the weighting function. They can be simplified using Green's theorem.

$$\iint_{\pi} \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y}\right) \mathrm{d}x \, \mathrm{d}y = \oint_{\partial\pi} P \mathrm{d}x + \oint_{\partial\pi} Q \mathrm{d}y \tag{4.14}$$

If we let $Q = x^p\, y^q\, w(\mathbf{x})\, I(\mathbf{x})$ and $P = 0$ in (4.14), we have

$$\iint_{\pi} \left[x^p\, y^q\, w(\mathbf{x})\, \frac{\partial I}{\partial x} + x^p\, y^q\, \frac{\partial w}{\partial x}\, I(\mathbf{x}) + p\, x^{p-1}\, y^q\, w(\mathbf{x})\, I(\mathbf{x}) \right] \mathrm{d}x\, \mathrm{d}y \tag{4.15}$$

$$= \oint_{\partial\pi} x^p\, y^q\, w(\mathbf{x})\, I(\mathbf{x}) \mathrm{d}y \tag{4.16}$$

So, equation (4.13a) becomes

$$\begin{aligned} m_{pq}^{\nabla x} = & - \iint_{\pi} \left(x^p\, y^q\, \frac{\partial w}{\partial x} I(\mathbf{x}) + p\, x^{p-1}\, y^q\, w(\mathbf{x}) I(\mathbf{x}) \right) \mathrm{d}x\, \mathrm{d}y \\ & + \oint_{\partial\pi} x^p y^q w(x,y) I(x,y) \mathrm{d}y \end{aligned} \tag{4.17}$$

Using the definition of the weighted moments in (4.10), we subsequently have

$$\begin{aligned} m_{pq}^{\nabla x} = & - p\, m_{p-1,q} - \iint_{\pi} x^p\, y^q\, \frac{\partial w}{\partial x} I(\mathbf{x})\, \mathrm{d}x\, \mathrm{d}y \\ & + \oint_{\partial\pi} x^p\, y^q w(x,y) I(x,y) \mathrm{d}y \end{aligned} \tag{4.18}$$

The last term has to be evaluated over the image border $\partial\pi$. In our previous developments in Section 3.2 (and in [Bakthavatchalam 13]), this term was removed thanks to the assumption that the image borders are of zero intensity. This hypothesis is invalid when there are dynamic image regions which enter and leave during the visual servo. Here, thanks to the choice of the weighting function which is such that $w(x,y) = 0$ $\forall (x,y) \in \partial\pi$, the last term vanishes again but without the previous zero intensity assumption.

If we take $P = x^p\, y^q w(x,y) I(x,y)$ and $Q = 0$ in Green's theorem and make the same series of developments as before, $m_{pq}^{\nabla y}$ can be obtained. So, the terms in (4.13) related to the interaction matrix of the WPM become

$$m_{pq}^{\nabla x} = -p\, m_{p-1,q} - \iint_{\pi} x^p\, y^q\, \frac{\partial w}{\partial x}\, I(x,y) \mathrm{d}x \mathrm{d}y \tag{4.19a}$$

$$m_{pq}^{\nabla y} = -q\, m_{p,q-1} - \iint_{\pi} x^p\, y^q\, \frac{\partial w}{\partial y}\, I(x,y) \mathrm{d}x \mathrm{d}y \tag{4.19b}$$

Let us note that if $w(x,y) = 1\ \forall (x,y)$, the gradients of w would vanish. The terms in (4.19) would then reduce to a single term (which was the case in our previous developments in Chapter 3 and in [Bakthavatchalam 13] as well). That is, they reduce exactly to the case of photometric moments with no weight. If the weighting function is not constant, then by replacing (4.19) in Equation 4.12 (Sec.4.2.1), we can observe that the interaction matrix not only contains terms dependent on the weighted moments m_{pq} but

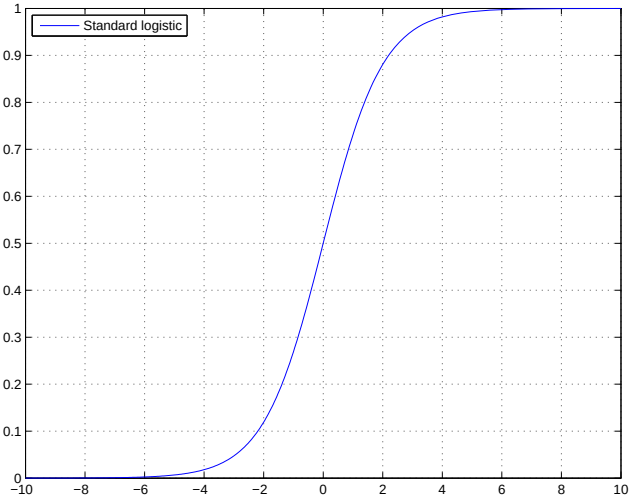


Figure 4.4 – Standard logistic function $l(x) = \dfrac{1}{1+\exp^{-x}}$ *on the domain* $-10 \leq x \leq 10$. *There is a smooth transition from* 0 *to* 1 *with the half-maximum attained at* $x = 0$

.

also terms which are composed of $\iint_{\pi} x^p y^q \dfrac{\partial w}{\partial x} I(\mathbf{x})$. This necessitates the computation of these terms to obtain the interaction matrix related to the weighted photometric moments. The next step is the selection of an appropriate weighting function such that we obtain in the interaction matrix expressions which are functions of the defined moments only.

4.2.2 Weighting schemes

The weighting function should attribute maximal importance to the pixels in the area around the image centre, smoothly reducing it outwards towards the image edges. As pointed out earlier, the first order spatial derivatives of $w(\mathbf{x})$ should be continuous and exist $\forall (x, y) \in \pi$. The standard logistic function was initially considered for the weighting scheme (see Fig.4.4). This function smoothly varies between 0 and 1 and has simple derivatives. This is a standard function that is used in machine learning and is a special case of the generalized logistic function.

$$\mathcal{L}(x) = \frac{A}{B + \exp^{-Cx}} \tag{4.20}$$

With this choice however, it is not possible to obtain the interaction matrix in an analytical form as functions of the weighted photometric moments. We propose to use functions with the general structure given in (4.21), where the exponential is raised to the negative power of a polynomial function.

$$\mathcal{F}(x) = \exp^{-p(x)} \quad \text{with} \tag{4.21}$$

$$p(x) = a_0 + a_1 x + \frac{1}{2} a_2 x^2 + \frac{1}{3} a_3 x^3 + ... + \frac{1}{n} a_n x^n \tag{4.22}$$

Functions of this structure possess the interesting property that their derivatives can be expressed in terms of the function itself. It is straight-forward to find the derivative of $\mathcal{F}(x)$. It is given by:

$$\mathcal{F}'(x) = \exp^{-p(x)}\ p'(x) \quad \text{with} \tag{4.23}$$

$$p'(x) = a_1 + a_2 x + a_3 x^2 + ... + a_n x^{n-1} \tag{4.24}$$

Two simple functions with the form proposed in (4.21) are:

$$g(x, y) = K \exp^{-a(x^2+y^2)} \tag{4.25}$$

which is similar to the classical Gaussian (see Figure 4.5) and a custom exponential function (see Figure 4.6) with higher powers for the spatial coordinates given by

$$w(x, y) = K \exp^{-a(x^2+y^2)^2} \tag{4.26}$$

These choices are interesting because they allow the interaction matrix to be obtained directly in closed-form as a function of the weighted moments itself. So, this computation does not introduce any overhead since it requires nothing other than weighted moments of upto a particular order, like previously. In the weighting functions introduced, K is the maximum value of the weighting function and a can be used to vary the areas which receive maximal and minimal weights respectively.

4.2.2.1 Interaction Matrix with Gaussian Weighting

In this section, the interaction matrix when using the Gaussian as the weighting function ($w(\mathbf{x}) = g(x, y)$) is developed. The moments thus computed can be written as

$$m_{pq} = \iint_{\pi} x^p y^q\, g\,(x, y)\ I\,(\mathbf{x}, t)\ \mathrm{d}x\,\mathrm{d}y \tag{4.27}$$

where $g(x, y)$ is given by Equation (4.25).

The spatial derivatives of g are straight-forward to obtain.

$$\frac{\partial g}{\partial x} = -2ax\, g(x, y) \tag{4.28a}$$

$$\frac{\partial g}{\partial y} = -2ay\, g(x, y) \tag{4.28b}$$

Substituting Equations (4.28) into (4.19), we obtain

$$m_{pq}^{\nabla x} = -p\ m_{p-1,q} + 2a m_{p+1,q} \tag{4.29a}$$

$$m_{pq}^{\nabla y} = -q\ m_{p,q-1} + 2a m_{p,q+1} \tag{4.29b}$$

By substitution of the Equations in (4.29) with the generic interaction matrix in Equation (4.12), the interaction matrix of the Gaussian weighted photometric moments ${}^g\mathbf{L}_{m_{pq}}$ can be deduced.

$${}^g\mathbf{L}_{m_{pq}} = \begin{bmatrix} L_{\mathtt{m_{pq}}}^{v_x} & L_{\mathtt{m_{pq}}}^{v_y} & L_{\mathtt{m_{pq}}}^{v_z} & L_{\mathtt{m_{pq}}}^{\omega_x} & L_{\mathtt{m_{pq}}}^{\omega_y} & L_{\mathtt{m_{pq}}}^{\omega_z} \end{bmatrix} \quad \text{with} \tag{4.30}$$

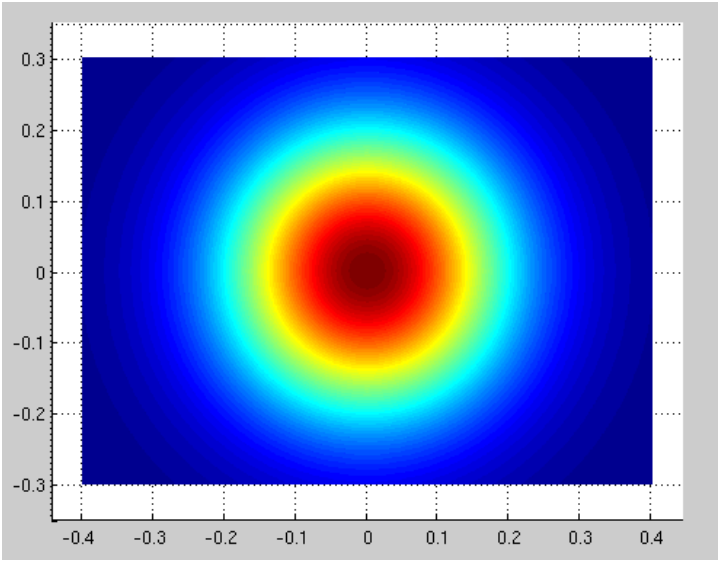


Figure 4.5 – Gaussian function $g(x,y) = \exp^{-25(x^2+y^2)}$ *in the domain* $-0.4 \leq x \leq 0.4$ *and* $-0.3 \leq y \leq 0.3$*. Importance wanes from maximum (red) in the centre radially outwards to minimum(blue) at the edges*

$$
\left.
\begin{aligned}
L^{v_x}_{\mathtt{m_{pq}}} &= A\left(-(p+1)m_{pq} + 2am_{p+2,q}\right) + B\left(-pm_{p-1,q+1} + 2am_{p+1,q+1}\right) \\
&\quad + C\left(-pm_{p-1,q} + 2am_{p+1,q}\right) \\
L^{v_y}_{\mathtt{m_{pq}}} &= A\left(-qm_{p+1,q-1} + 2am_{p+1,q+1}\right) + B\left(-(q+1)m_{p,q} + 2am_{p,q+2}\right) \\
&\quad + C\left(-pm_{p,q-1} + 2am_{p,q+1}\right) \\
L^{v_z}_{\mathtt{m_{pq}}} &= A\Big((p+q+3)m_{p+1,q} - 2a(m_{p+1,q+2} + m_{p+3,q})\Big) \\
&\quad + B\Big((p+q+3)m_{p,q+1} - 2a(m_{p,q+3} + m_{p+2,q+1})\Big) \\
&\quad + C\Big((p+q+2)m_{p,q} - 2a(m_{p+2,q} + m_{p,q+2})\Big) \\
L^{\omega_x}_{\mathtt{m_{pq}}} &= q\,m_{p,q-1} + (p+q+3)\,m_{p,q+1} - 2a(m_{p+2,q+1} + m_{p,q+1} + m_{p,q+3}) \\
L^{\omega_y}_{\mathtt{m_{pq}}} &= -p\,m_{p-1,q} + (p+q+3)\,m_{p+1,q} + 2a(m_{p+1,q} + m_{p+3,q} + m_{p+1,q+2}) \\
L^{\omega_z}_{\mathtt{m_{pq}}} &= pm_{p-1,q+1} - qm_{p+1,q-1}
\end{aligned}
\right\}
$$

On inspection of Equation (4.30), we can deduce the following:

1. We can infer that the interaction matrix of the Gaussian weighted moments can be written as the sum of two matrices as

$$
{}^{g}\mathbf{L}_{m_{pq}} = \mathbf{L}_1 + 2a\,{}^{g}\mathbf{L}_2 \tag{4.31}
$$

where $\mathbf{L}_1$ has the same analytical form as the interaction matrix of the moments in Equation (2.29) and the photometric moments in Equation (3.33). ${}^{g}\mathbf{L}_2$ is directly related to the Gaussian weighting function in the moments definition. Also note that the moments are now computed with the function defined in (4.25).

2. For a weighted moment of order $p+q$, computation of $\mathbf{L}_1$ requires moments of order upto $(p+q+1)$ whereas for $\mathbf{L}_2$, moments of order upto $(p+q+3)$ are necessary.

This is in fact a resultant of the term $(x^2 + y^2)$, to which the exponential is raised (see (4.25)).

3. In the case of a uniform weighting function, all the image pixels receive the same weight $w = 1$. The interaction matrix of the weighted moments reduces to the case in (2.29). This can be easily inferred since $g(x, y) = 1$ would imply $a = 0$ and therefore ${}^{g}\mathbf{L}_{m_{pq}} = \mathbf{L}_1$.

4.2.2.2 Interaction Matrix with Custom Exponential Weighting

In this section, the interaction matrix when using the custom exponential weighting function $w(\mathbf{x}) = w(x, y))$ given by (4.26) is developed. The moments thus computed can be defined as

$$m_{pq} = \iint_{\pi} x^p y^q \, w\,(x, y)\; I\,(\mathbf{x}, t)\; \mathrm{d}x\, \mathrm{d}y \tag{4.32}$$

We then continue with the developments with (4.26) as the weighting function. A visualization of this function as a surface over a 640×480 domain is given in Figure 4.7. The spatial derivatives of the custom exponential are as follows:

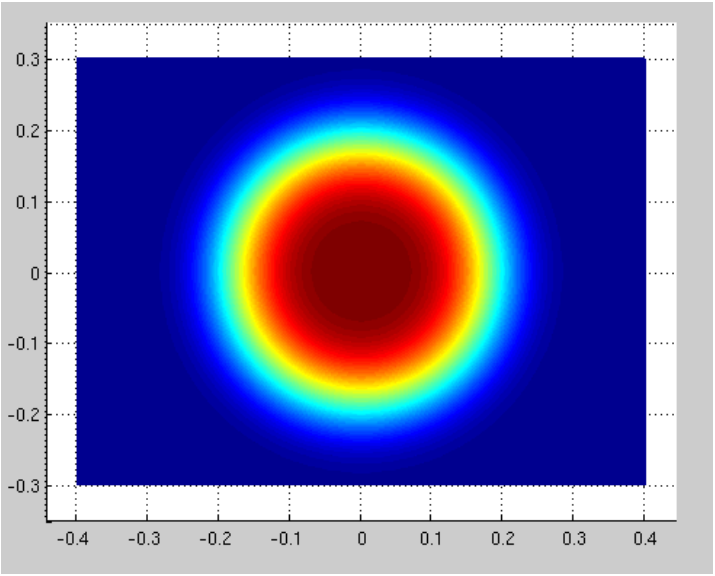


Figure 4.6 – Custom exponential function $w(x, y) = \exp^{-650(x^2+y^2)^2}$ *in the domain* $-0.4 \leq x \leq 0.4$ *and* $-0.3 \leq y \leq 0.3$. *Importance wanes from maximum (red) in the centre outwards to minimum(blue) at the edges*

$$\frac{\partial w}{\partial x} = -4ax(x^2 + y^2)\, w(\mathbf{x}) \tag{4.33a}$$

$$\frac{\partial w}{\partial y} = -4ay(x^2 + y^2)\, w(\mathbf{x}) \tag{4.33b}$$

Substituting (4.33) into (4.19), we obtain

$$m_{pq}^{\nabla x} = -p\; m_{p-1,q} + 4a\,(m_{p+3,q} + m_{p+1,q+2}) \tag{4.34a}$$

$$m_{pq}^{\nabla y} = -q\ m_{p,q-1} + 4a\left(m_{p,q+3} + m_{p+2,q+1}\right) \tag{4.34b}$$

By combining Equations in (4.34) with the generic interaction matrix in (4.12), the interaction matrix of photometric moments $\mathbf{L}_{m_{pq}}$ weighted with the radial function (4.26) can be obtained.

$$\begin{aligned}
{}^{w}\mathbf{L}_{m_{pq}} &= \begin{bmatrix} L_{\mathtt{m}_{\mathtt{pq}}}^{v_x} & L_{\mathtt{m}_{\mathtt{pq}}}^{v_y} & L_{\mathtt{m}_{\mathtt{pq}}}^{v_z} & L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_x} & L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_y} & L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_z} \end{bmatrix} \quad \text{with} \\
L_{\mathtt{m}_{\mathtt{pq}}}^{v_x} &= A\left(-(p+1)\,m_{p,q} + 4\,a\,(m_{p+4,q} + m_{p+2,q+2})\right) \\
&\quad + B\left(-pm_{p-1,q+1} + 4\,a\,(m_{p+3,q+1} + m_{p+1,q+3})\right) \\
&\quad + C\left(-pm_{p-1,q} + 4\,a\,(m_{p+3,q} + m_{p+1,q+2})\right) \\
L_{\mathtt{m}_{\mathtt{pq}}}^{v_y} &= A\left(-qm_{p+1,q-1} + 4\,a\,(m_{p+3,q+1} + m_{p+1,q+3})\right) \\
&\quad + B\left(-(q+1)\,m_{p,q} + 4\,a\,(m_{p,q+4} + m_{p+2,q+2})\right) \\
&\quad + C\left(-qm_{p,q-1} + 4\,a\,(m_{p,q+3} + m_{p+2,q+1})\right) \\
L_{\mathtt{m}_{\mathtt{pq}}}^{v_z} &= A[(p+q+3)\,m_{p+1,q} \\
&\quad - 4\,a\,(m_{p+5,q} + 2m_{p+3,q+2} + m_{p+1,q+4})] \\
&\quad + B[(p+q+3)\,m_{p,q+1} \\
&\quad - 4\,a\,(m_{p+4,q+1} + 2m_{p+2,q+3} + m_{p,q+5})] \\
&\quad + C[(p+q+2)\,m_{p,q} \\
&\quad - 4\,a\,(m_{p+4,q} + 2m_{p+2,q+2} + m_{p,q+4})] \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_x} &= (p+q+3)m_{p,q+1} + qm_{p,q-1} \\
&\quad - 4\,a(m_{p+4,q+1} + 2\,m_{p+2,q+3} + m_{p,q+3} \\
&\quad + m_{p+2,q+1} + m_{p,q+5}) \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_y} &= -pm_{p-1,q} - (p+q+3)m_{p+1,q} \\
&\quad + 4\,a(m_{p+3,q} + m_{p+1,q+2} + m_{p+5,q} \\
&\quad + 2\,m_{p+3,q+2} + m_{p+1,q+4}) \\
L_{\mathtt{m}_{\mathtt{pq}}}^{\omega_z} &= pm_{p-1,q+1} - qm_{p+1,q-1}
\end{aligned} \tag{4.35}$$

From (4.35), the following inferences can be made.

1. The interaction matrix can be expressed as a matrix sum

$$ {}^{w}\mathbf{L}_{m_{pq}} = \mathbf{L}_1 + 4a\mathbf{L}_2 \tag{4.36}$$

   where $\mathbf{L}_1$ is the interaction matrix when $w(x,y) = 1$ and has the same form as in Equations (2.29) and (3.33).And we recall that the moments are now computed with the weighting function defined in (4.32).

2. To compute $\mathbf{L}_1$, moments of order upto $(p+q+1)$ are required whereas $\mathbf{L}_2$ is a function of moments $m_{t,u}$, where $t+u <= p+q+5$. This is in fact a resultant of the term $(x^2+y^2)^2$, to which the exponential is raised (See (4.26)).

3. As before, if all the image pixels are uniformly weighted with $w = 1$, the interaction matrix of the weighted moments reduces to the case in (2.29). If $w = 1 \forall (x,y)$, this would imply $a = 0$ and therefore ${}^{w}\mathbf{L}_{m_{pq}} = \mathbf{L}_1$.

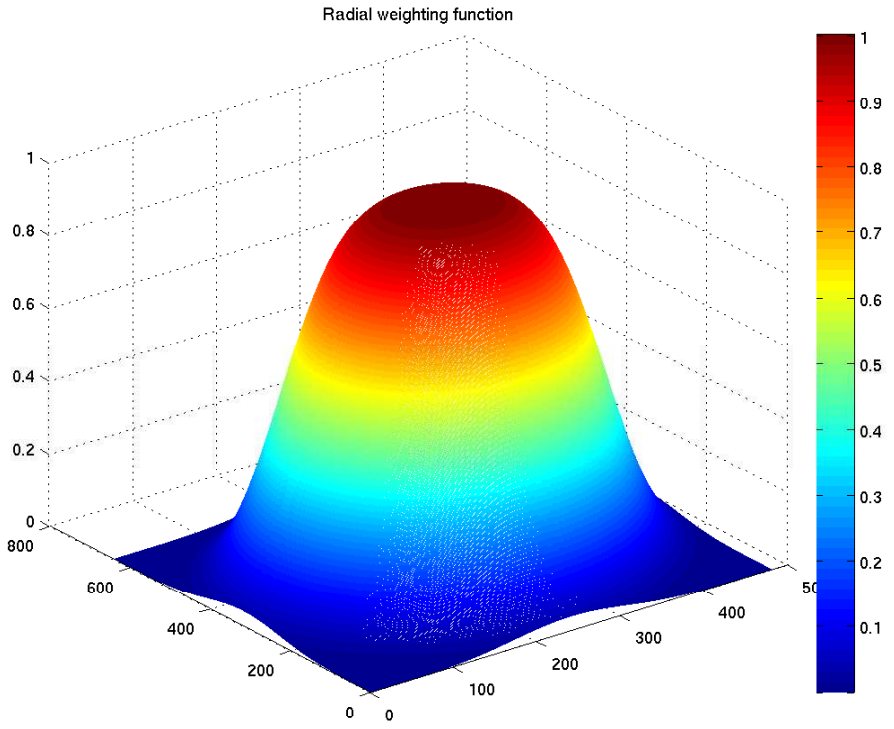


Figure 4.7 – Surface of the weighting function in (4.26) with $K = 1$, $a = 650$

Invariance of WPM There is an important inference which can be drawn which is common to both the weighting function choices discussed above.

1. On observation of $L^{\omega_z}_{\mathtt{m}_{\mathtt{pq}}}$ in Equations (4.30) and (4.35), we see that it does not contain any new terms when compared to Equation (2.29). That is, the weighting function has not induced any extra terms, thus retaining the invariance of the moment invariants to optic axis rotations. This outcome was of course desired from the structure of the weighting function.

2. On the other hand, if we consider $L^{v_x}_{\mathtt{m}_{\mathtt{pq}}}, L^{v_y}_{\mathtt{m}_{\mathtt{pq}}}, L^{v_z}_{\mathtt{m}_{\mathtt{pq}}}, L^{\omega_x}_{\mathtt{m}_{\mathtt{pq}}}$ and $L^{\omega_y}_{\mathtt{m}_{\mathtt{pq}}}$, we observe that additional terms are contributed by the weighting function. As a result, invariants to scale and 2D translations developed in Chapter 2 from the classical moments will not exhibit the same invariant properties when used with WPM. Thus, there is a need to develop new invariants for use with WPM such that they would retain their invariance to translations. Unfortunately, this issue has not been explored in this thesis due to lack of time.

In Equations (4.30) and (4.35). the matrix $\mathbf{L}_2$ is tied directly to the weighting function. The order of moments necessary for its computation is determined by the choice of the weighting function. Finally and as usual, the components of the interaction matrix corresponding to the rotational motions are still free from the 3D parameters A, B and C.

A comparison between using the logistic functions (defined piece-wise), a simple Gaussian (4.25) and a custom exponential function (4.26) to weight a 1-D domain is shown in Fig.4.8. It can be seen that the length of the domain that received maximum

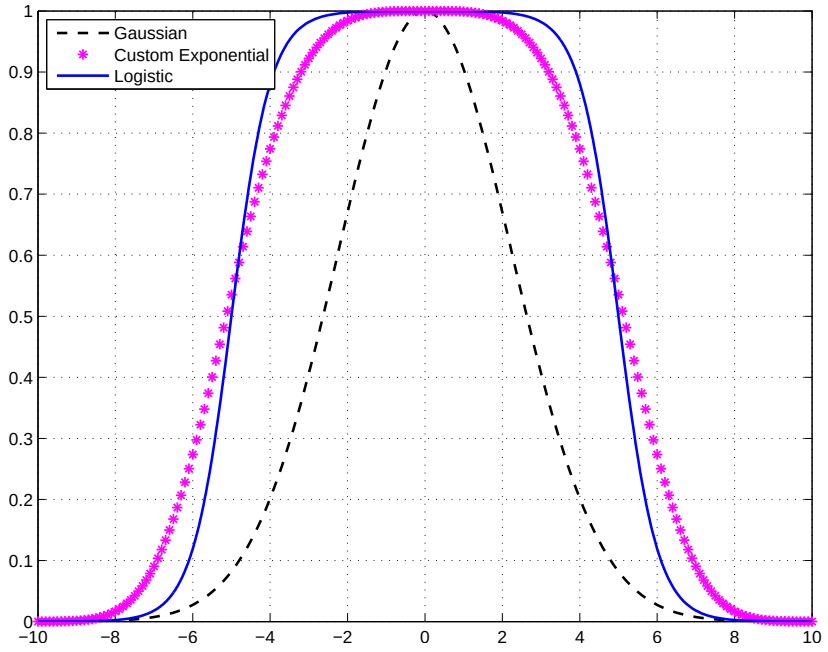


Figure 4.8 – A comparison of weighting with the Gaussian $g(x) = \exp^{-(x^2)/10}$*, Custom Exponential* $ce(x) = \exp^{-(x^4)/1000}$ *and generalized logistic functions(defined piece-wise) on* $10 \leq x \leq 10$

weight is clearly longer for the custom exponential function than for the classical Gaussian, which is a supplementary motivation to select the custom exponential function. With this selection, we will analyse some visual features introduced in Chapter 2, but this time developed from the weighted photometric moments.

4.3 Analysis of the Weighted Moments

In this section, an analysis of the interaction matrix of the lower order weighted moments is made. We take into account moments weighted with the custom exponential function in (4.32).

4.3.1 Weighted Photometric Area Moment

The zeroth order moment can be interpreted as the weighted photometric area. The interaction matrix of this feature can be derived from Equation (4.35) simply by taking $p = q = 0$ and by using $K^{'} = 4a$ for brevity. We obtain

$$\begin{cases} L_{\mathtt{m}_{00}}^{v_x} = -m_{0,0}A + K^{'}[A\alpha_1 + B\alpha_2 + C\lambda_1] \\ L_{\mathtt{m}_{00}}^{v_y} = -m_{0,0}B + K^{'}[A\alpha_2 + B\alpha_3 + C\lambda_2] \\ L_{\mathtt{m}_{00}}^{v_z} = m_{00}(3/Z_g - C) - K^{'}[A\lambda_3 + B\lambda_4 + C(\alpha_1 + \alpha_3)] \\ L_{\mathtt{m}_{00}}^{\omega_x} = 3m_{00}\, y_g - K^{'}(\lambda_2 + \lambda_5) \\ L_{\mathtt{m}_{00}}^{\omega_y} = -3m_{00}\, x_g + K^{'}(\lambda_1 + \lambda_3) \\ L_{\mathtt{m}_{00}}^{\omega_z} = 0 \end{cases}$$

with

$$\begin{cases} \alpha_1 = m_{4,0} + m_{2,2} \\ \alpha_2 = m_{3,1} + m_{1,3} \\ \alpha_3 = m_{0,4} + m_{2,2} \\ \lambda_1 = m_{3,0} + m_{1,2} \\ \lambda_2 = m_{0,3} + m_{2,1} \\ \lambda_3 = m_{1,4} + 2m_{3,2} + m_{5,0} \\ \lambda_4 = m_{0,5} + 2m_{2,3} + m_{4,1} \end{cases}$$

We saw previously in Chapter 2 that the area m_{00} is a very interesting visual feature for the original non-weighted moments. This feature is mainly related to the translation motions in z. It is invariant to (i) rotations around the optic axis and (ii) to planar translations when the camera and target planes are parallel. From the terms in the interaction matrix, we indeed observe that the invariance to optic axis rotations is preserved thanks to our choice of the weighting function. The influence on planar translations can be studied by taking $A = B = 0$ in the corresponding terms of the interaction matrix. Then, we will have $L_{\mathtt{m}_{00}}^{v_x} = K^{'} C(m_{3,0} + m_{1,2})$ and $L_{\mathtt{m}_{00}}^{v_y} = K^{'} C(m_{0,3} + m_{2,1})$. Clearly, third order moments appear, which means that the invariance property is lost. From the experimental results however, this loss did not have any effect on the convergence of the servo. On the contrary, it is to be noted that if these terms are neglected, it will introduce a gross error into the interaction matrix and the visual servoing might not converge at all.

4.3.2 Weighted Centre of gravity

We recall that the centre of gavity is nothing but the ratio between the first and zeroth order moments computed from the acquired image $x_g = m_{10}/m_{00}$ and $y_g = m_{01}/m_{00}$. The related interaction matrix can be computed using

$${}^{w}\mathbf{L}_{x_g} = \left({}^{w}\mathbf{L}_{m_{1,0}} - x_g \, {}^{w}\mathbf{L}_{m_{0,0}}\right)/m_{0,0} \tag{4.37}$$

Then, we obtain

$$\begin{cases} L_{x_g}^{v_x} = -1/Z_g \\ \quad + \dfrac{K'}{m_{0,0}}\Big[A\epsilon_1 + B\epsilon_2 + C(\alpha_1 - x_g\lambda_1)\Big] \\ L_{x_g}^{v_y} = \dfrac{K'}{m_{0,0}}\Big[A\epsilon_2 + B(\delta_2' - x_g\alpha_3) + C(\alpha_2 - x_g\lambda_2)\Big] \\ L_{x_g}^{v_z} = x_g/Z_g + 4\left(An_{20} + Bn_{11}\right) \\ \quad - \dfrac{K'}{m_{0,0}}\Big[A(\phi_1 - x_g\lambda_3) + B(\phi_2 - x_g\lambda_4) \\ \quad + C(\lambda_5 - x_g(\alpha_1 + \alpha_3))\Big] \\ L_{x_g}^{\omega_x} = (-3x_g\, y_g + 4n_{1,1}) \\ \quad + \dfrac{K'}{m_{0,0}}\Big[(\phi_2 + \alpha_2) + x_g(\lambda_2 + \lambda_4)\Big] \\ L_{x_g}^{\omega_y} = -(1 - 3x_g^2 + 4n_{20}) \\ \quad + \dfrac{K'}{m_{0,0}}\Big[(\phi_1 + \alpha_1) - x_g(\lambda_1 + \lambda_3)\Big] \\ L_{x_g}^{\omega_z} = y_g \end{cases}$$

with

$$\begin{cases} \epsilon_i = \delta_i - x_g\alpha_i \\ \delta_1 = m_{5,0} + m_{3,2} \\ \delta_2 = m_{4,1} + m_{2,3} \\ \phi_1 = m_{2,4} + 2m_{4,2} + m_{6,0} \\ \phi_2 = m_{5,1} + 2m_{3,3} + m_{1,5} \end{cases}$$

and $n_{i,j} = \mu_{i,j}/m_{0,0}$, where $\mu_{i,j}$ are the centred moments introduced in Section 2.3.2.1. From the second term $L_{x_g}^{v_y}$ in the interaction matrix, we can notice that translational velocities in y will modify the value of x_g even when the camera and target planes are parallel. Hence x_g is not invariant to motions in y. This loss of invariance to translation motions was quite expected due to spatial weighting. We can notice the weighting function did not alter the behaviour of the centre of gravity to optic axis rotations. For instance, $L_{x_g}^{\omega_z} = y_g$ was also the case when the weighting is uniform, that is, when $w(\mathbf{x}) = 1, \forall \mathbf{x}$. In a similar manner, the interaction matrix of the other coordinate y_g can also be obtained.

$${}^w\mathbf{L}_{y_g} = \frac{{}^w\mathbf{L}_{m_{0,1}} - y_g\, {}^w\mathbf{L}_{m_{0,0}}}{m_{0,0}} \tag{4.38}$$

After some developments, we obtain as before the terms of the interaction matrix

$${}^w\mathbf{L}_{y_g} = \begin{bmatrix} L_{y_g}^{v_x} & L_{y_g}^{v_y} & L_{y_g}^{v_z} & L_{y_g}^{\omega_x} & L_{y_g}^{\omega_y} & L_{y_g}^{\omega_z} \end{bmatrix} \tag{4.39}$$

where

$$\begin{cases} L_{y_g}^{v_x} = \dfrac{K^{'}}{m_{0,0}}\Big[A(\delta_2 - y_g\alpha_1) + B\epsilon_2^{'} + C(\alpha_2 - y_g\lambda_1)\Big] \\ L_{y_g}^{v_y} = -1/Z_g + \\ \quad \dfrac{K^{'}}{m_{0,0}}\Big[A\epsilon_2^{'} + B(\delta_1^{'} - y_g\alpha_3) + C(\alpha_3 - y_g\lambda_2)\Big] \\ L_{y_g}^{v_z} = y_g/Z_g + 4\left(An_{11} + Bn_{02}\right) \\ \quad - \dfrac{K^{'}}{m_{0,0}}\Big[A(\phi_2 - y_g\lambda_3) + B(\phi_1^{'} - y_g\lambda_4) \\ \quad + C(\lambda_4 - y_g(\alpha_1 + \alpha_3))\Big] \\ L_{y_g}^{\omega_x} = (1 - 3y_g^2 + 4n_{02}) \\ \quad - \dfrac{K^{'}}{m_{0,0}}\Big[(\phi_1^{'} + \alpha_3) - y_g(\lambda_2 + \lambda_4)\Big] \\ L_{y_g}^{\omega_y} = -\left(3x_g\, y_g + 4n_{1,1}\right) \\ \quad + \dfrac{K^{'}}{m_{0,0}}\Big[(\phi_2 + \alpha_2) - y_g(\lambda_1 + \lambda_3)\Big] \\ L_{y_g}^{\omega_z} = -x_g \end{cases}$$

with

$$\begin{cases} \epsilon_i^{'} = \delta_i^{'} - y_g\alpha_i \\ \delta_1^{'} = m_{0,5} + m_{2,3} \\ \delta_2^{'} = m_{1,4} + m_{3,2} \\ \phi_1^{'} = m_{4,2} + 2m_{2,4} + m_{0,6} \end{cases}$$

The interaction matrix does not present any specifically interesting properties except having characteristics similar to ${}^{w}\mathbf{L}_{x_g}$ as far as the y coordinate of the centre of gravity respectively.

In an analogous way, an analysis of the lower order moments m_{00}, x_g and y_g for the WPM using a Gaussian weighting function (defined in Equation (4.27)) can be performed. This is trivial to do and the expressions would be simpler than those obtained in the above analysis done for the custom weighting function.

4.3.3 Weighted Centred moments

Photometric centred moments are a natural extension of the centred moments and are given by

$$\mu_{pq} = \iint_{\pi} (x - x_g)^p\,(y - y_g)^q\; w\left(\mathbf{x}\right)\; I\left(\mathbf{x}, t\right)\; \mathrm{d}x\,\mathrm{d}y \tag{4.40}$$

where x_g and y_g are the weighted centre of gravity described previously in Section 4.3.2. As explained in Section 2.3.2.1, the interaction matrix of the centred moments can be

easily calculated from the relation:

$$\mu_{pq} = \sum_{k=0}^{p}\sum_{l=0}^{q}(-1)^{p+q-k-l}\binom{p}{k}\binom{q}{l}(-x_g)^{(p-k)}(-y_g)^{(q-l)}\, m_{kl} \tag{4.41}$$

where m_{kl} is a weighted moment of order $k+l$. Further, the interaction matrix of the centred moments can be expressed as

$$\left.\begin{aligned}\mathbf{L}_{\mu_{pq}} = \sum_{k=0}^{p}\sum_{l=0}^{q}(-1)^{p+q-k-l}\binom{p}{k}\binom{q}{l}\Big[m_{kl}\Big((p-k)(x_g)^{p-k-1}y_g\,{}^{w}\mathbf{L}_{x_g}\\ +(q-l)x_g(y_g)^{q-l-1}\,{}^{w}\mathbf{L}_{y_g}\Big)+(x_g)^{p-k}(y_g)^{q-l}\,{}^{w}\mathbf{L}_{m_{kl}}\Big]\end{aligned}\right\} \tag{4.42}$$

where ${}^{w}\mathbf{L}_{m_{kl}}$ are the interaction matrices of the weighted moments, ${}^{w}\mathbf{L}_{x_g}$ and ${}^{w}\mathbf{L}_{y_g}$ are already known from Equations (4.37) and (4.39) respectively. The loss of invariance resulting from the weighting function affects the centred moments as well. These moments are no longer invariant to planar translations and thus invariants developed from these centred moments would no longer have the same beneficial properties. That is why we pointed out previously that there is a need for new invariants that can be used with WPM.

4.3.4 Weighted Photometric orientation feature

The orientation α is a very interesting feature due to its direct relation to rotational motions around the optic axis. It can be computed from the second order moments.

$$\alpha = \frac{1}{2}\arctan\left(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}}\right) \tag{4.43}$$

where $\mu_{p,q}$ are the centred moments defined in Equation (4.40). It would be interesting to study the orientation feature calculated using the weighted and non-weighted moments. For the texture in Figure 4.9, the orientation with the weighted moments was computed

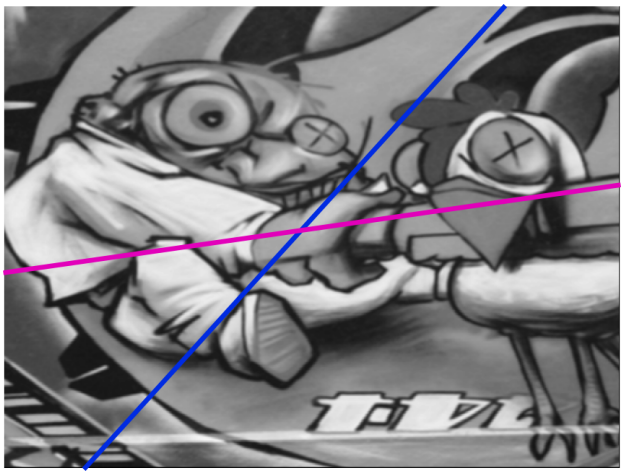


Figure 4.9 – Orientation Feature α

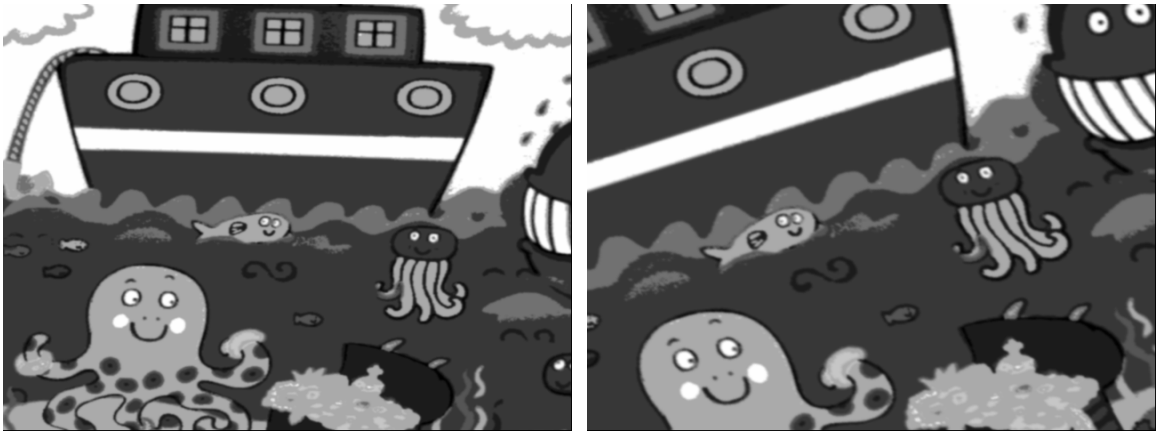
as $8.075°$ whereas with the non-weighted moments, the orientation value obtained was $51.87°$. This is not surprising since in the case of the weighted moments, the spatial weights are such that only the pixels in the central portions of the image are attributed more importance and the pixels in the periphery of the image are ignored. This feature is reliable in any case and as evidenced by the results that have been presented. The interaction matrix of α can be obtained from the appropriate moments by taking the derivative of (4.43).

$$ {}^{w}\mathbf{L}_{\alpha} = \frac{\mathbf{L}_{\mu_{1,1}}\left(\mu_{2,0}-\mu_{0,2}\right)-\mu_{1,1}\left(\mathbf{L}_{\mu_{2,0}}-\mathbf{L}_{\mu_{0,2}}\right)}{4\mu_{1,1}^{2}+\left(\mu_{2,0}-\mu_{0,2}\right)^{2}} \tag{4.44} $$

Due to space constraints, the exact terms in Equation (4.44) are not given here. As with the case of the centre of gravity and area feature, the behaviour with respect to optic axis rotations is not modified. That is, the last term of the interaction matrix is equal to -1, indicating that it is linearly related to rotational motions around the optic axis. We recall that this is a very useful property for visual servoing.

4.4 Validation Results for 4dof

4.4.1 Validation of Analytical Modelling



(a) Camera view at desired robot pose (b) Camera view at initial robot pose

Figure 4.10 – Testing the weighted photometric moments

We first consider only 4dof namely 3d translational motions and rotation around the optic axis. In this reduced-DOF, weighted photometric moments are surprisingly effective, as is demonstrated after the modelling validation. The desired camera pose is chosen such that it is $1.0m$ vertically above the target in a fronto-parallel configuration. The image learnt from the desired pose is shown in Fig.4.10(a). The initial pose is selected in such a way that the camera observes the image shown in Fig 4.10(b). The displacements required for convergence are a translation of ${}^{c*}\mathbf{t}_{c} = [4.0cm, 3.0cm, 20cm]$ and a rotation of $R_z = 25°$ around the optic axis. From the desired and initial images, we can clearly observe that the borders around the image are not of zero intensity. The experiment is set up deliberately so that the zero border assumption (ZBA) presented in Chapter 3 is

violated. The objective of this experiment is to validate the correctness of the improved modelling scheme introduced with weighted photometric moments.

The classic first-order Gauss-Newton control law with the reduced feature set, as proposed in 3.5.1.1 was used. The interaction matrix $\mathbf{L_s}(\mathbf{s}(t), Z(t))$ was updated at each iteration. This choice enables the validation of the modelling by checking if the control law using the weighted photometric moments result in an exponential convergence of the errors to 0. In simulation, the depths $Z(t)$ are readily available from ground truth and need not be estimated. A gain of $\lambda = 2$ was used for this experiment.

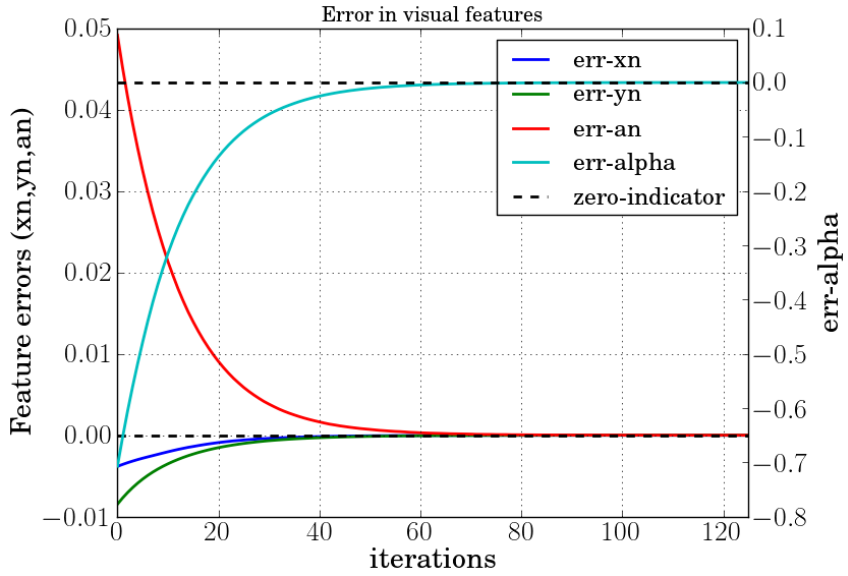
As expected, the control law results in an exponential convergence of the visual feature errors as shown in Fig.4.12(a). This can be observed more clearly from the log-scale plot in Fig.4.11(b). The camera trajectory is also very satisfactory as shown in Fig.4.11(c), with a deviation of only $0.96cm$ from the ideal geodesic. This is further confirmed by the low value of the geodesic area metric at $0.0035m^2$. We note that usage of instantaneous depths is not mandatory for error convergence. The interaction matrix can be used with the current visual feature values $\mathbf{s}(t)$ and depth can just be approximated with the value at the desired pose $Z(t) = Z^*$. This choice still results in an extremely satisfactory behaviour with a near exponential (but not perfect) convergence of the visual feature errors as seen from Figure 4.12. This has been demonstrated in [Bakthavatchalam 15] as well.

This validates the enhanced modelling scheme introduced with the weighted photometric moments. The influx of information especially at the borders of the image is handled by the improved model using weighting functions. This exponential convergence validates the correctness of the improved modelling scheme introduced with the weighted photometric moments. Next, we will compare the weighted photometric moments with currently existing methods.

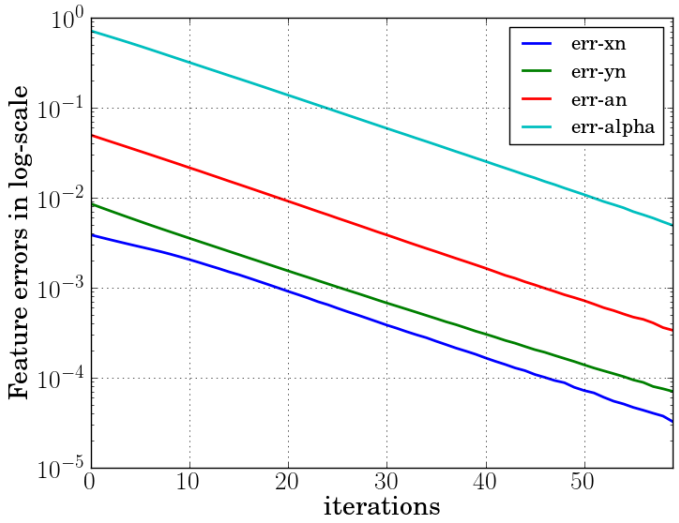
4.4.2 Comparison to state of the art

4.4.2.1 Comparison with non-weighted moments

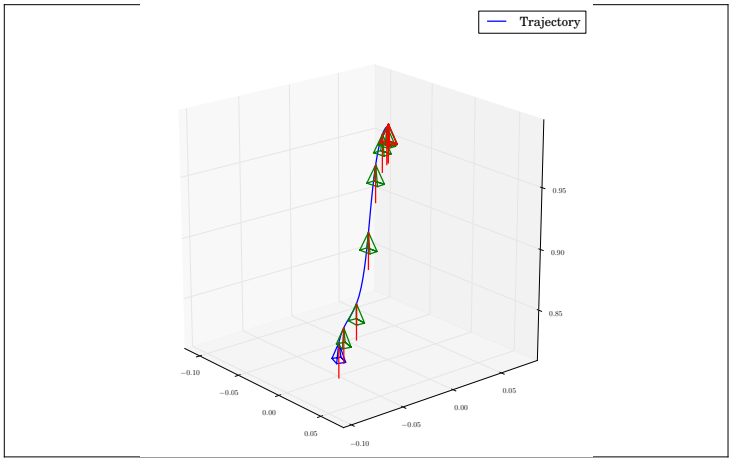
First, visual servoing using the weighted photometric moments is compared with the non-weighted moments, namely photometric moments introduced in Chapter 3 (and in [Bakthavatchalam 13]). For this, the same desired and initial poses chosen in Section 4.4.1 were used. Also, the control law and the interaction matrix configurations is the same. The only difference for the non-weighted moments is that the weighting is uniform for all the pixels. As seen from Fig.4.13(a), the decrease in the errors for the non-weighted moments is not satisfactory at all and the servoing did not succeed. After 50 iterations for example, the algorithm using the proposed WPM is driven near to convergence as shown in Fig.4.13(a). For the non-weighted moments, there is still a large error as shown in 4.13(e). The reason is because these non-weighted photometric moments ignore the effect of the influx of intensities in their modelling by the restrictive zero border assumption (ZBA). Since image moments are global features which are computed from the integral of the product of the spatial coordinates and the intensity, the feature errors are not representative of the actual system state. This in turn leads to velocities which drive the system away from the desired equilibrium state. The experiment was stopped when the error increased rapidly, otherwise leading to a jitter. The difference image $\mathbf{I} - \mathbf{I}^*$ at the end of servoing for the non-weighted case is shown in Fig.4.14(a) and for the weighted



(a) Errors in visual features (Multi-scale) y-axis on the left for x_n, y_n, a_n and α

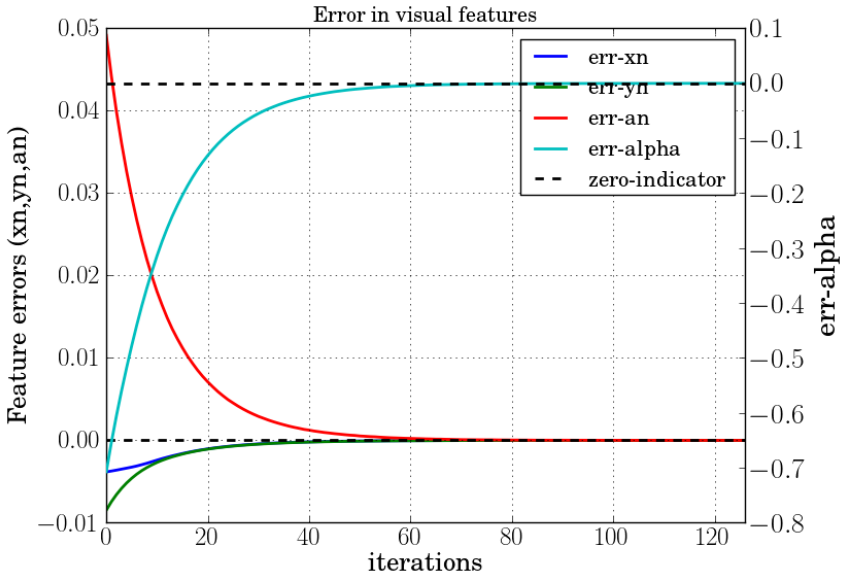


(b) Errors in log-scale to observe exponential decrease



(c) Camera 3-D trajectory

Figure 4.11 – Visual Servoing Results with the proposed Weighted Photometric Moments using $\widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}(t)$



(a) Errors in visual features (Multi-scale) y-axis on the left for x_n, y_n, a_n and α

Figure 4.12 – Satisfactory convergence with depth approximation

photometric moments is shown in Fig.4.14(c).

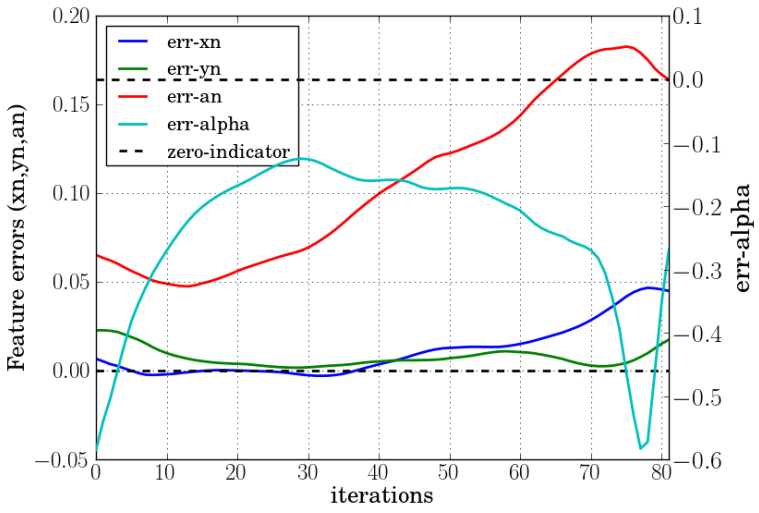
4.4.2.2 Comparison with Pure photometric Visual Servoing

In this experiment, the objective is to compare the behaviour of the weighted photometric moments and the pure luminance [Collewet 11] as visual features (when the zero border assumption no longer holds). The following classical visual servoing control law was used.

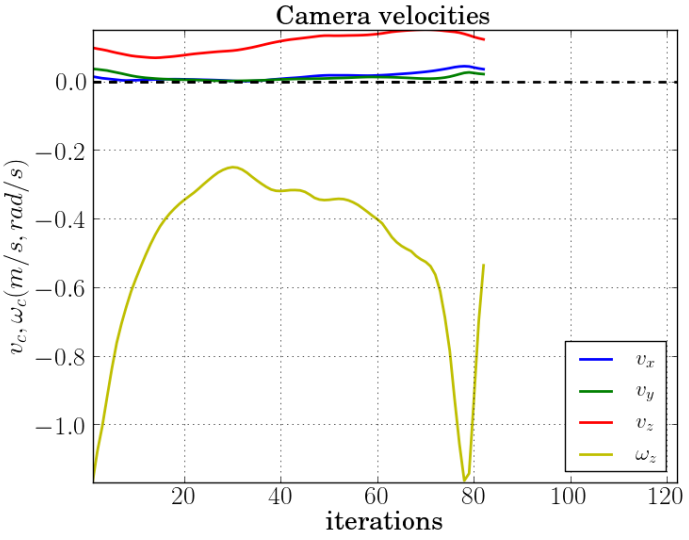
$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{I}}^{+} (\mathbf{I} - \mathbf{I}^*) \tag{4.45}$$

where $\mathbf{I}$ is the image acquired at the current pose, $\mathbf{I}^*$ is the image learnt from the desired pose, $\mathbf{L_I} \in N_{pix} \times N_{dof}$ is the interaction matrix of the luminance feature. This control law is equivalent to a Gauss-Newton[1] optimization and is used here for a fair comparison between the visual features. Further, for this comparison, the interaction matrix at the current configuration $\mathbf{L_I}(\mathbf{I}(t), Z(t))$ was used. We consider only $N_{dof} = 4$ degrees of freedom, just like for the weighted photometric moments. From the evolution of the error norm in Fig.4.13(c) and velocities in Fig.4.13(d), we observe that $\|\mathbf{e}\| \neq 0$ but the control velocities $\mathbf{v}_c$ become 0. In this case, the servo is considered to have reached a local minimum [Chaumette 06] and the task execution failed. Therefore, no more control actions are generated to act on the error $\mathbf{I} - \mathbf{I}^*$. The difference image at the end of the servo in Fig.4.14(b) shows that the errors still remain while for the weighted photometric moments in Fig.4.14(c), the errors have been regulated to 0. In pure photometric visual servoing, the number of rows in the interaction matrix is equal to the number of pixels in the image. When the interaction matrix is updated at each servo iteration, the time per iteration would be higher than the case when the interaction matrix is evaluated only

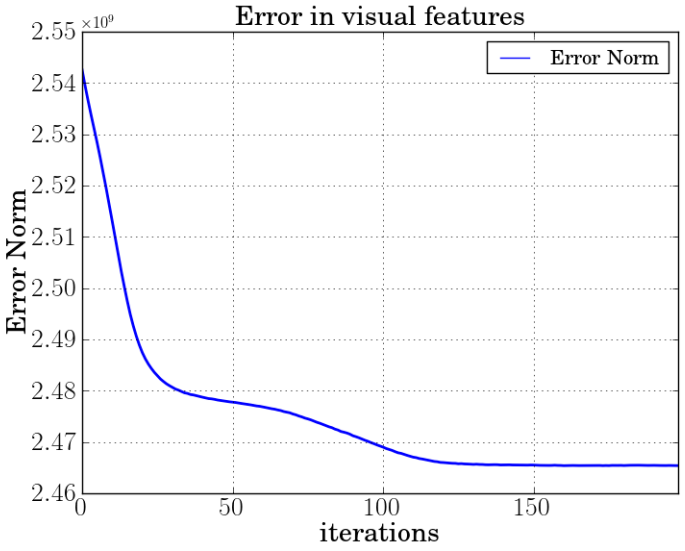
[1] In [Collewet 11], the proposed control law used an approximated Hessian based on the Levenberg-Marquardt approach



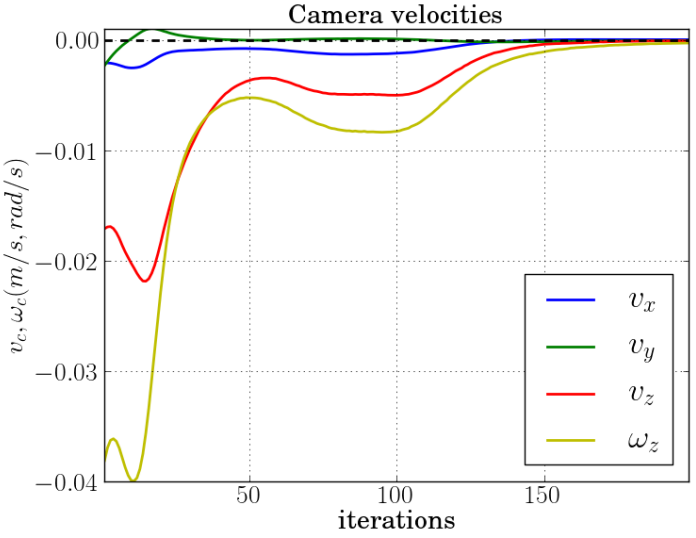
(a) Non-weighted moments : Errors in x_n, y_n, a_n on left and α on right



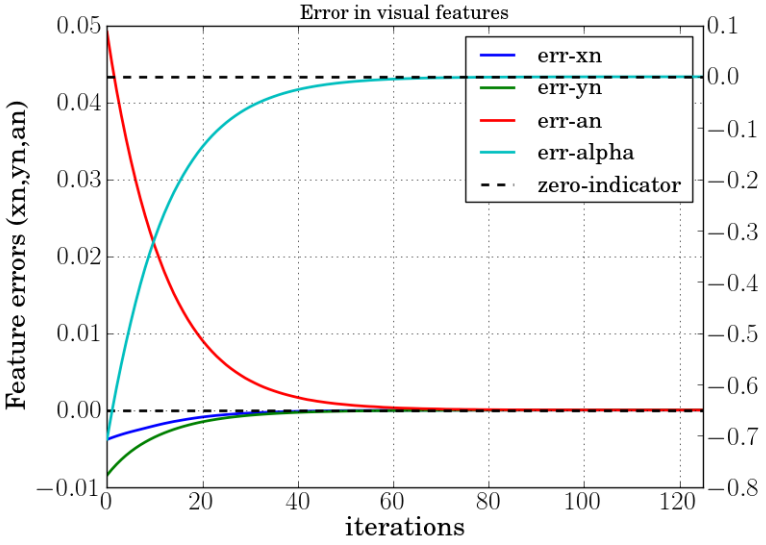
(b) Non-weighted moments : Camera velocities



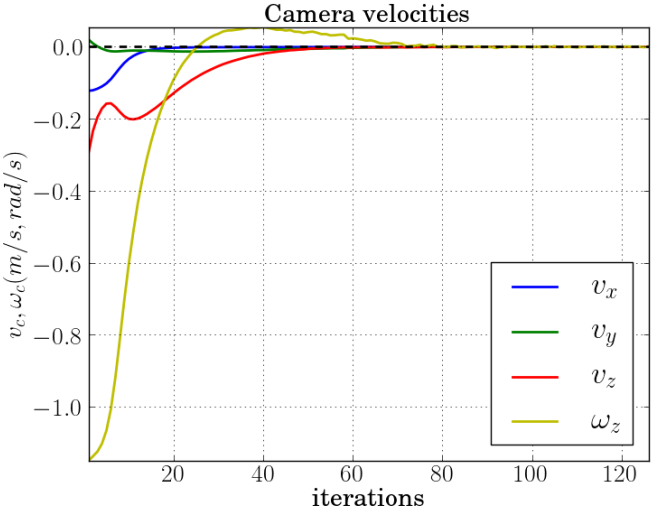
(c) Pure photometric VS : Error Norm $\|\mathbf{I} - \mathbf{I}^*\|$



(d) Pure photometric VS : Camera velocities



(e) WPM : Errors in x_n, y_n, a_n on left and α on right



(f) Camera velocities

Figure 4.13 – Comparison of WPM to non-weighted moments and pure photometric VS

(a) Non-weighted moments

(b) Pure photometric VS

(c) Weighted photometric moments : Difference image at end of servo

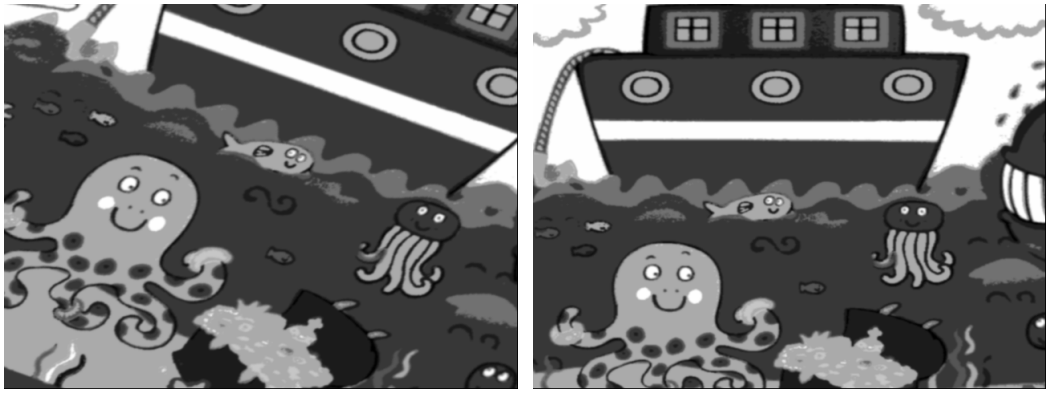
Figure 4.14 – Difference Images at the end of visual servoing using different visual features

at the desired pose. In this particular experiment with images of size 640×480, the pure photometric method took on the average $218ms$ per iteration whereas the weighted photometric moments took $76ms$.

Thus, we clearly see that not only moments but also pure photometric VS method runs into problems when there are uncommon image portions between the desired and currently acquired images during the servo. In contrast, the weighted photometric moments introduced in this chapter is better equipped to handle the difficult cases and facilitates convergence with the specified error profiles.

4.4.3 Visual Servoing with constant JPC control law

Unlike the previous experiment, here the interaction matrix at the desired configuration was used, with the depth approximated by the value at the desired pose $\widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}}(\mathbf{s}^*, \hat{Z}^*)$. As explained in Chapter 1, this configuration is called the constant Jacobian Pseudoinverse Control (JPC) law in [Malis 04]. In this experiment, the same desired pose as previously has been used. But an initial pose different from the previous one has been selected. The image acquired from this pose is shown in Fig.4.15(a). The desired image is shown in



(a) (b)

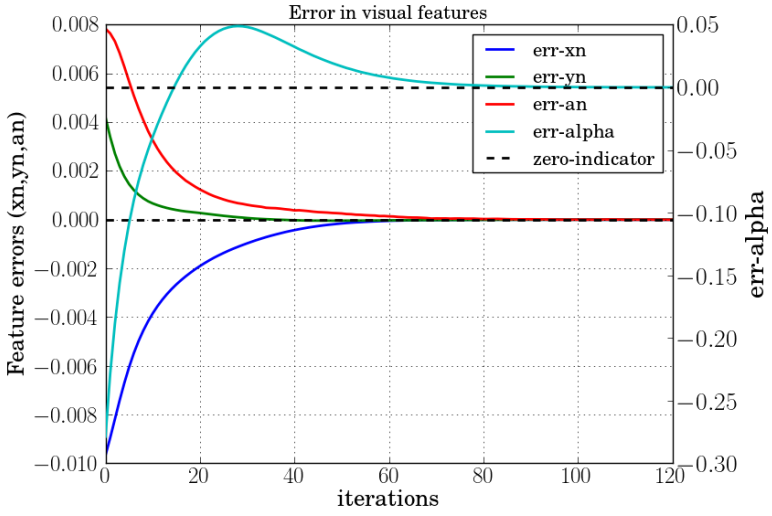
Figure 4.15 – Images acquired from initial and desired poses

Fig.4.15(b). From these two images, we can observe that there is no distinguishable background and the ship structure remains obstructed in the initial image. The displacements required for convergence are a translation of ${}^{c*}\mathbf{t_c} = [2.41cm, 6.65cm, 10cm]$ and a rotation of $\mathbf{R}_z(25°)$ around the optic axis. A gain of $\lambda = 2.0$ was used. Hence, the control action is linear in $(\mathbf{s} - \mathbf{s}^*)$ since $\mathbf{L}_{\mathbf{s}*}$ is constant and is simpler than in [Collewet 11]. The interaction matrix computed at the desired pose has a condition number of 2365.2 and is given by

$$\mathbf{L}_{\mathbf{s}*} = \begin{bmatrix} 0.1417 & -0.1134 & -0.0042 & -0.0069 \\ -0.1145 & -0.0015 & -0.0153 & 0.0032 \\ 0.1030 & 0.1024 & 0.0716 & 0 \\ -0.3779 & -8.4665 & -0.0879 & -1.0000 \end{bmatrix} \tag{4.46}$$

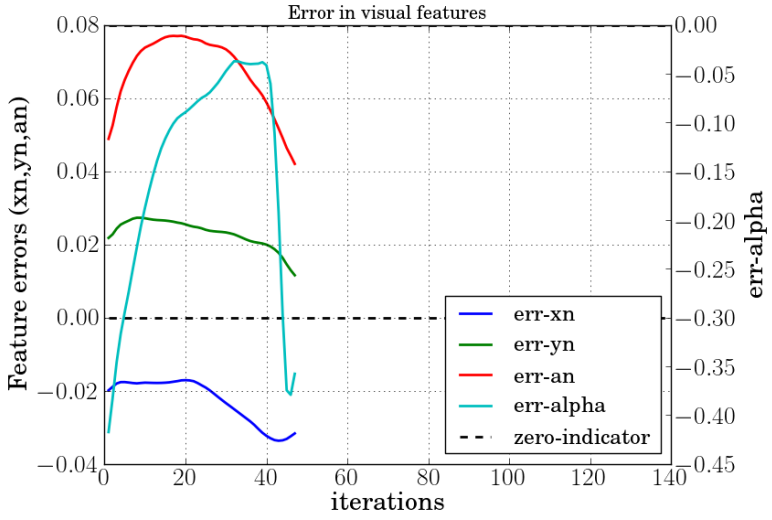
From Eq.(4.46), several inferences can be made. The feature a_n is invariant to optic axis rotations and α has retained its property with respect to optic axis rotations (-1 in last column). But x_n (resp. y_n) is no more invariant to translations in y (resp. x). This invariance loss was discussed in Sec.4.2.2. Also, a_n and α in (4.46) are more reactive to errors in degrees of freedom different from the ones they are intended to control (translations along and around the optic axis respectively). This consequence is due to the weighting function, which implies new features would have to be determined for a more optimal design. At this juncture, we would like to stress the importance of explicit analytical modelling which facilitates such analyses. Despite the loss in invariance, the visual servoing converged satisfactorily, especially with respect to the translation motions. The rotation error increases briefly before the final convergence. This is due to the extraneous regions appearing in the image during the initial stages of the servo. The 3D camera trajectory is not close to a straight line (see Fig.4.16(e)) but an overall satisfactory behaviour was obtained. The camera traversed a distance of $14.11cm$ in this experiment.

With the same desired and initial poses, the visual servoing with non-weighted photometric moments was tested. As seen from Fig.4.16(c), the decrease in the errors for the non-weighted moments is not satisfactory at all and the servoing diverged. After about 40 iterations, the algorithm using the proposed WPM is driven near to convergence whereas for the non-weighted moments, there is a large error as seen from the difference image in Fig.4.16(d). The experiment was stopped when the error norm increased rapidly, oth-



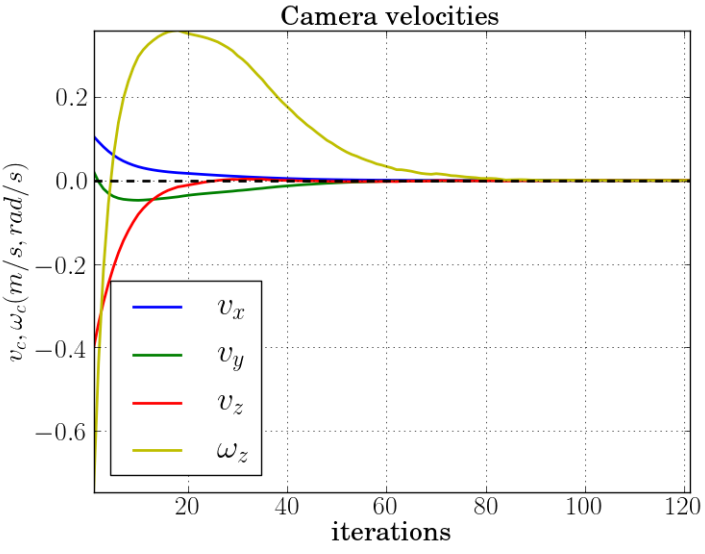
(a) Visual feature errors for WPM

(b) Difference image at end of servo for WPM

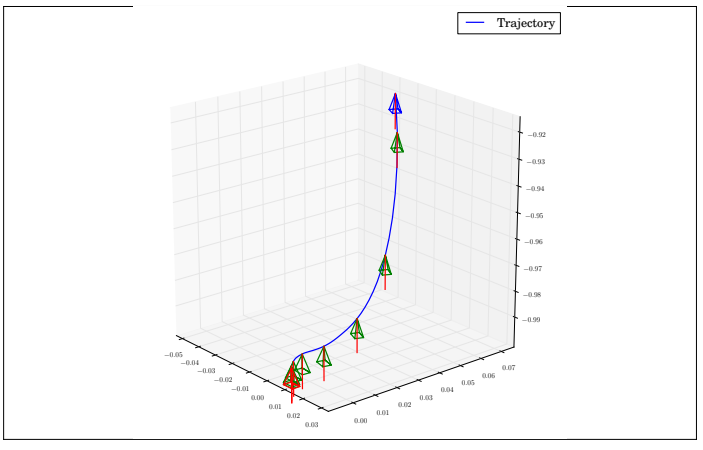


(c) Visual feature errors for non-weighted moments

(d) Difference image at end of servo for non-weighted moments

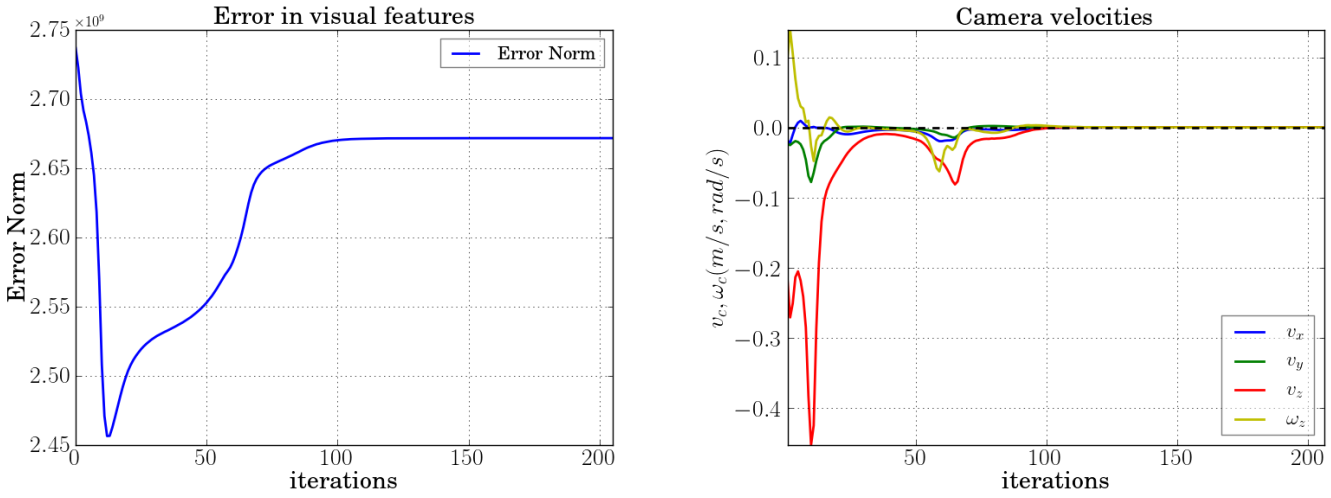


(e) Camera velocities for WPM



(f)

Figure 4.16 – Servo Results using constant JPC control law (see Sec.4.4.3)



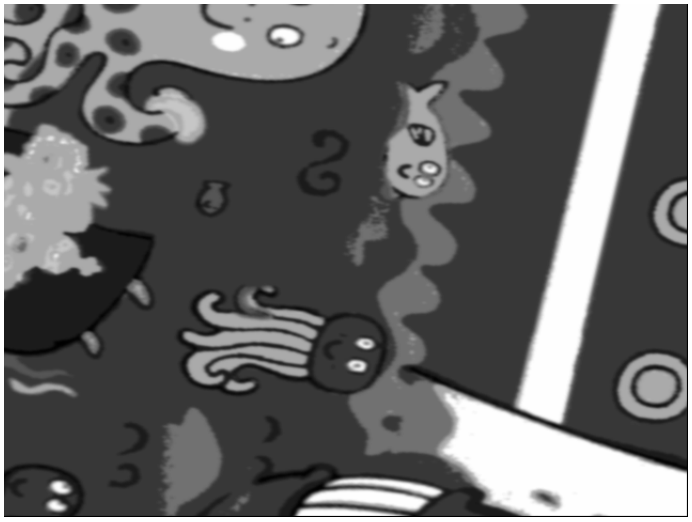
(a) Error norm $\|\mathbf{I} - \mathbf{I}^*\|$ (b) Camera velocities

Figure 4.17 – Servoing with pure luminance feature (see Sec.4.4.3)

erwise leading to a jitter. It is evident that the control law in this case cannot handle the extraneous regions. The same experiment was repeated again with the pure luminance feature. In this case again, we see that the generated velocities (shown in Figure 4.17(b)) could not regulate the errors to 0. The servo attained a local minimum. The experiment was stopped when there was no decrease in errors even after 100 control iterations.

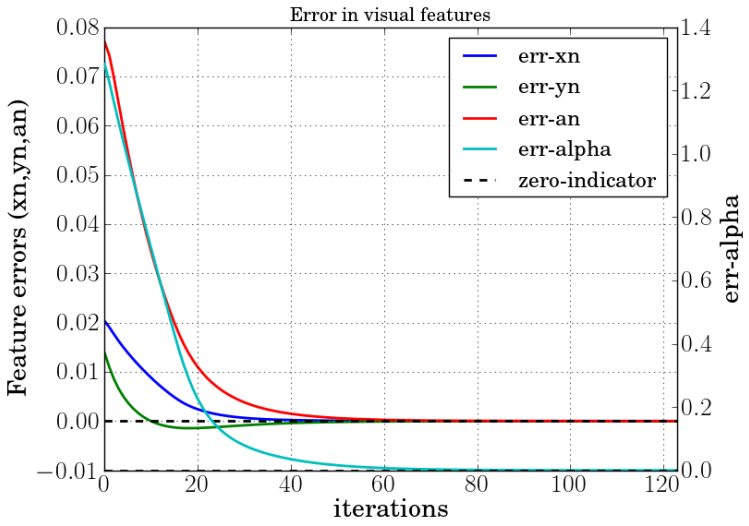
4.4.4 Performance under large motions

Presence of large rotational motions around the optic axis are in general difficult for visual servoing algorithms. For instance, when Cartesian coordinates of points in the image are employed as visual features, the camera recedes along the optic axis and further away from the desired pose. This classical issue is known as the camera retreat problem [Chaumette 98]. The goal of this experiment is to test the robustness of the weighted photometric moments when there are large rotations to be performed. Accordingly, an initial pose is chosen in such a way that a rotation of $100°$ is required around the optic axis for convergence to the desired pose. The image acquired from the initial pose is shown in Fig.4.18(a). What is expected is regulation of the errors to 0 such that the camera observes the image learnt from the desired pose shown in Fig.4.18(b). It has to be noted that a large motion is required for achieving the task.Precisely, the total translational displacement required for convergence are ${}^{c*}\mathbf{t_c} = [5cm, 4cm, 25cm]$. In addition to the rotation, a displacement along the optic axis is also necessary for convergence. The classical visual servoing control law with the same gain of $\lambda = 2$ as in the previous experiments was used. For this difficult task, the mean of the interaction matrices computed from the desired and current images was selected because this choice was shown to better represent the camera displacement and improve the visual servoing behaviour in [Malis 04]. The depths are not updated at each iteration and only approximated using $Z^* = 1$. This is on purpose to show that online depth estimation would not be necessary and an approximation of its value at the desired pose would be sufficient for convergence. The visual servoing converged to the

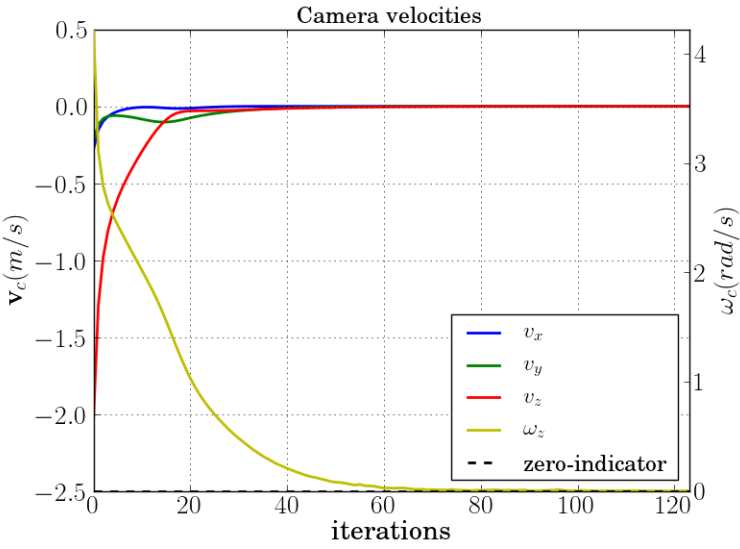


(a) Camera view at initial robot pose

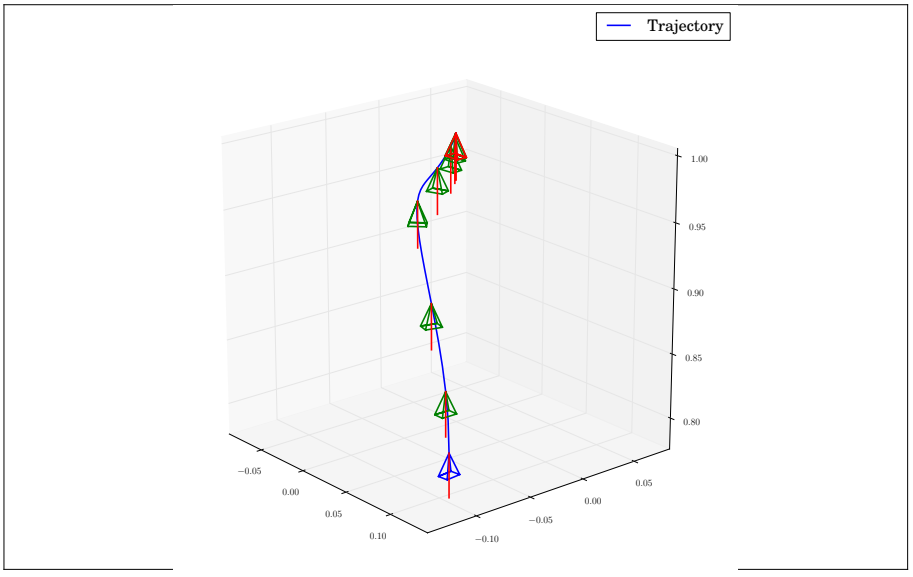
(b) Camera view at desired robot pose



(c)



(d)



(e)

Figure 4.18 – Performance under large rotations

desired pose with an accuracy of 0.29° in rotation and $[-0.07mm, -0.48mm, 0.61mm]$ in translation. The control velocities generated are shown in Fig.4.18(d) and the resulting Cartesian trajectories are shown in Fig.4.18(e). This experiment demonstrates that weighted photometric moments make excellent candidates for visual servoing endowed with a very large convergence domain.

4.4.5 Convergence Study with Weighted Photometric Moments

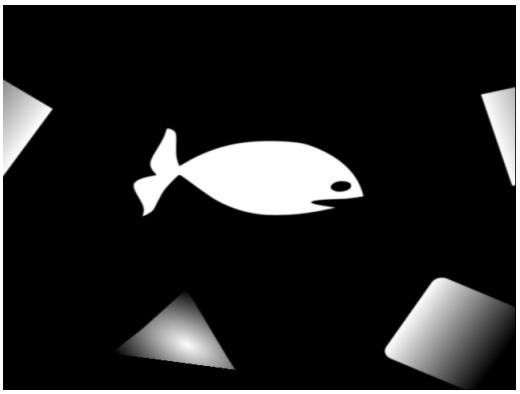
We recall that in IBVS, only local asymptotic stability can be demonstrated. However, an empirical analysis of the convergence can always be performed. Our goal here is to demonstrate the drastic improvement in convergence brought about by the enhanced modelling introduced through the weighted photometric moments.

4.4.5.1 Using synthetic imagery

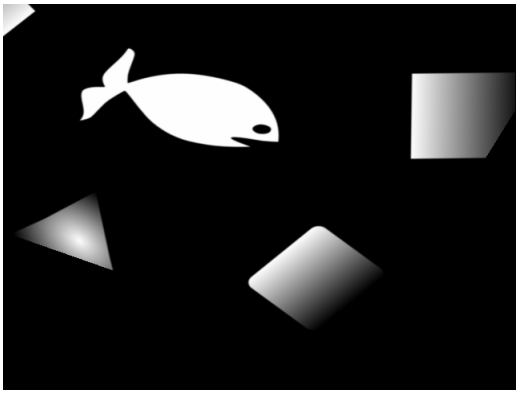
For this analysis, a synthetic image with several polygonal blocks at the image borders was prepared. This enables to simulate the inflow and outflow of information at the image borders. The desired pose was chosen at $1.8m$ vertically above the planar target with an initial planar rotation of 10° around the optic axis. The image learnt from this pose is shown in Fig.4.19(a). For the convergence analysis, positioning tasks starting from 243 different initial poses were considered. These 243 poses were divided to 3 sets of 81 poses each, conducted at 3 different depths of $1.8m$, $1.9m$ and $2.0m$ respectively. For all these initial poses, the camera is subjected to a rotation of 25° around the optic axis with respect to the horizontal. The rotations around the x and y axes are not considered since we take into account here only 4dof.

We used the interaction matrix at the desired configuration and the depth is approximated to its value at the desired pose. The same gain of $\lambda = 2$ as in the previous experiments was used. We consider an experiment to have converged if the task error $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ is reduced to less than $1e^{-10}$ in a maximum of 300 iterations. In addition to this condition, we also ensure that the SSD error defined by $\mathbf{e}_{SSD} = \sum_{\mathbf{x}}[I(\mathbf{x}) - I^*(\mathbf{x})]^2/N_{pix}$ between the final and learnt images is less than 1.0. Thanks to the SSD criterion, it is ensured that a non-desired equilibrium point is not considered as converged. That being said, in the experiments reported as converged, the final accuracy in pose is less than $1mm$ for translations and less than 1° for the planar rotation.

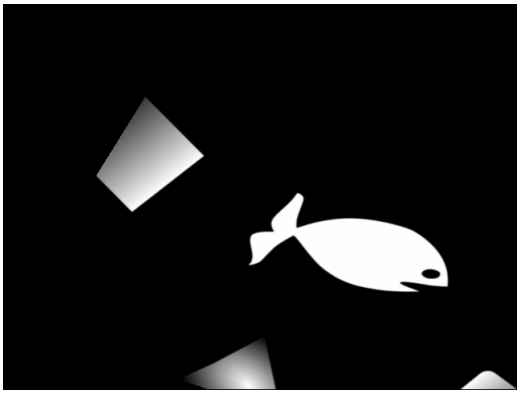
The results for the non-weighted moments are shown in Fig.4.20. Non-weighted moments meet with failure in all cases since they are not equipped to handle the information inflow from the image borders. This is not surprising since we also saw this in the previous experiments. An interesting connection to previous work on image moments could be drawn here. In [Chaumette 04] about binary moments, it was assumed that a well-segmented or thresholded region of the object is available in the image. In other words, there should not be any extraneous regions entering or leaving the image. The kind of images that have been considered in this experiment (as well as others in this chapter) for the servoing exactly violate this stated assumption. So the dismal performance of the non-weighted moments are in complete agreement with theory established in the state of the art. Let us note that with the photometric moments, we are not restricted by this assumption anymore. Then, results for the same set of servoing experiments when using the



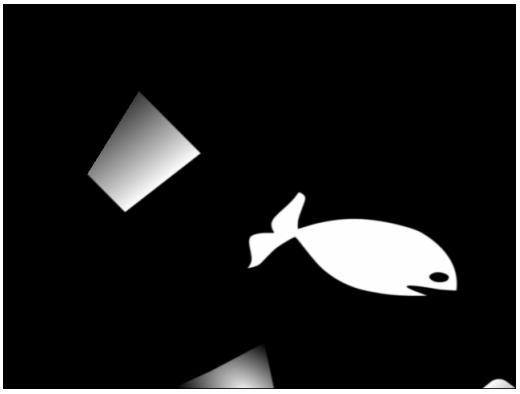
(a) Image learnt from desired robot pose



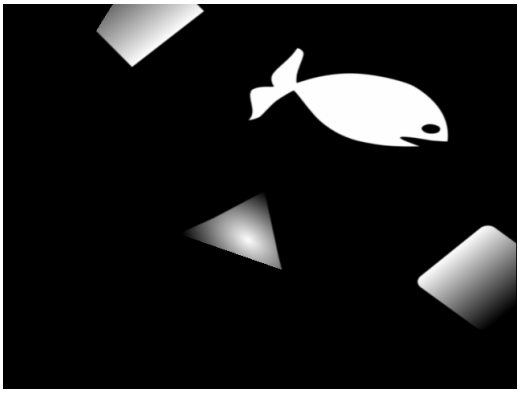
(b)



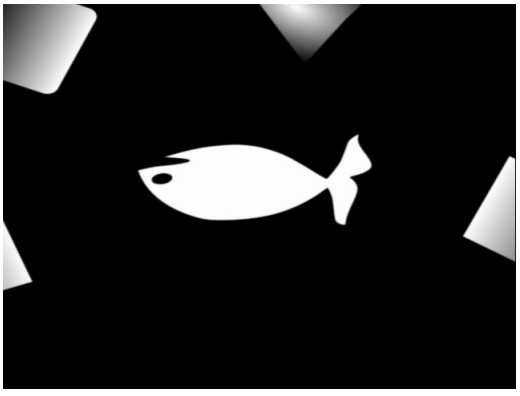
(c)



(d)

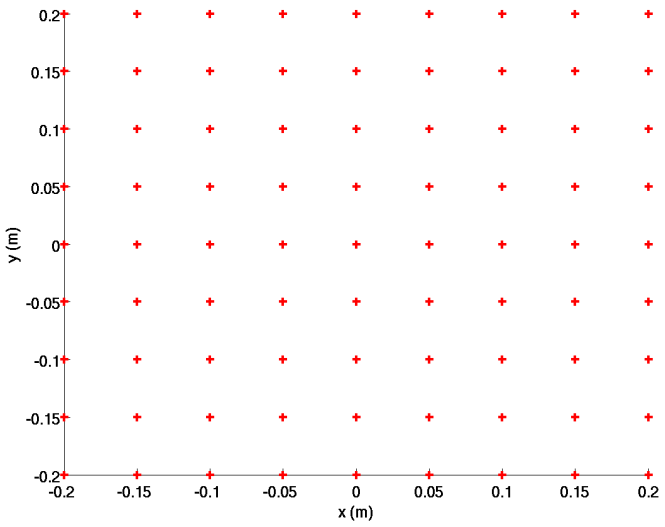


(e)

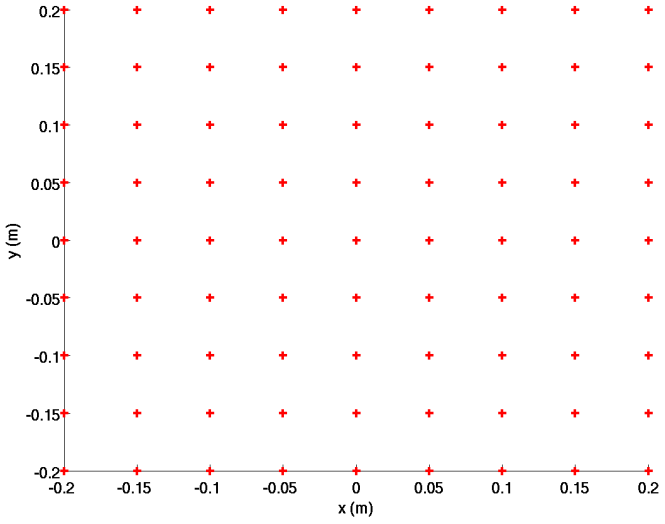


(f) Image observed at equilibrium point in a failure case

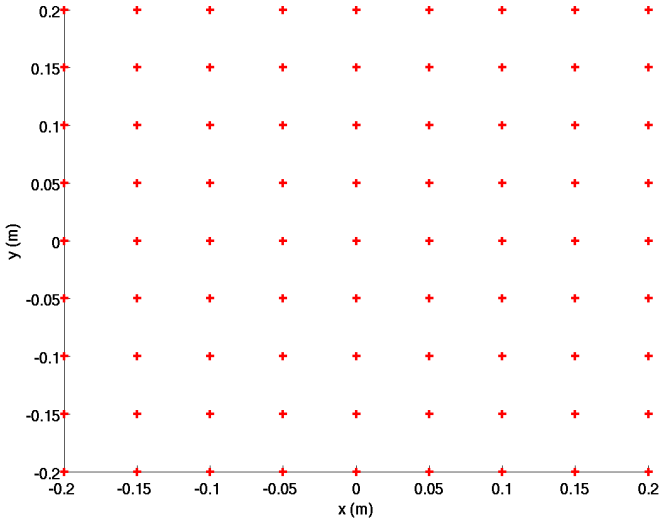
Figure 4.19 – A sampling of 4 different images from the generated initial poses are shown in 4.19(b)-4.19(e)



(a) Convergence Domain at $Z = 1.8m$

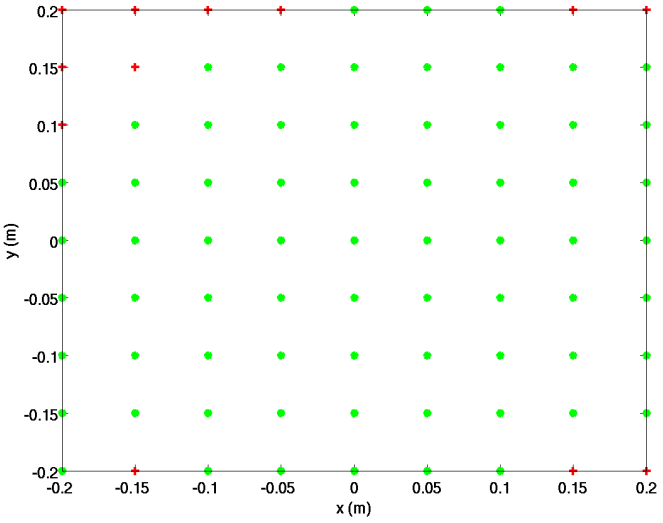


(b) Convergence Domain at $Z = 1.9m$

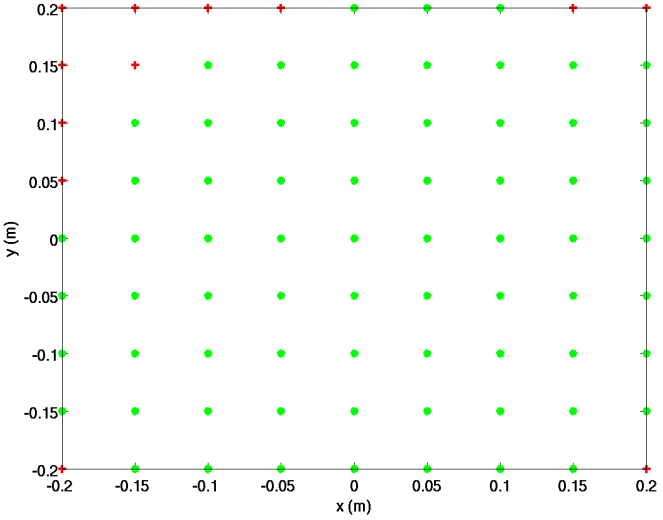


(c) Convergence Domain at $Z = 2.0m$

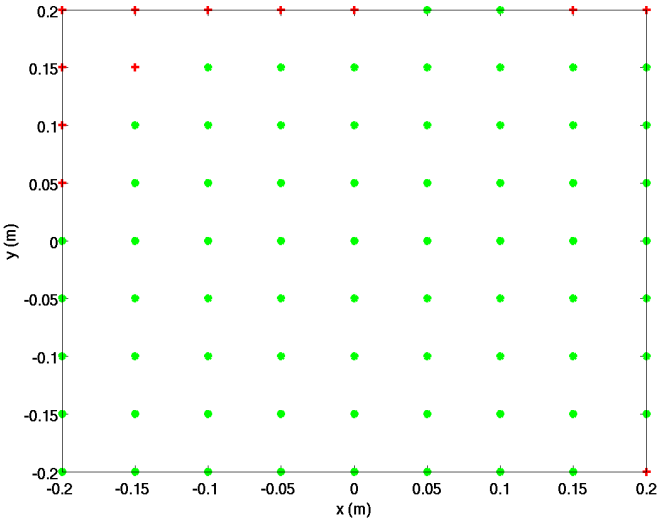
Figure 4.20 – Convergence results using non-weighted moments.Red cross : failed to converge



(a) Convergence Domain at $Z = 1.8m$

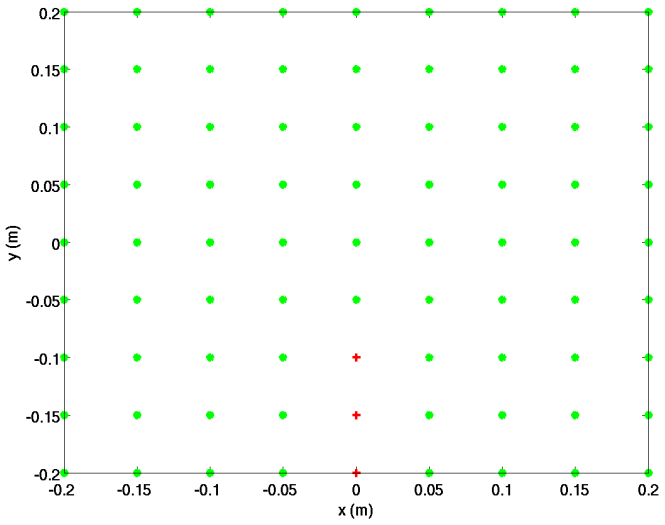


(b) Convergence Domain at $Z = 1.9m$

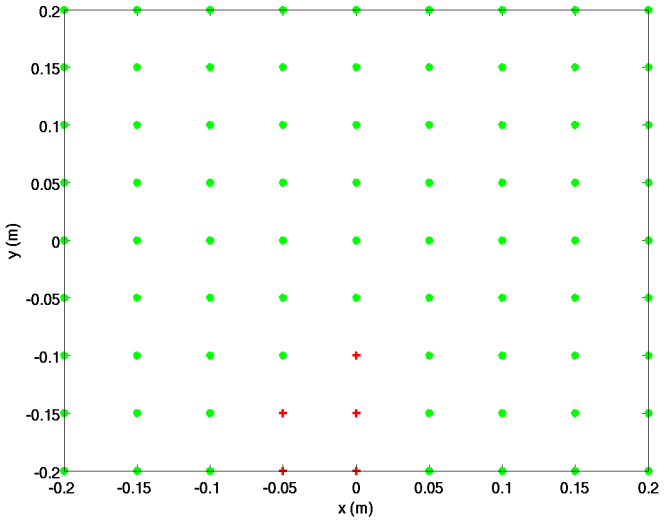


(c) Convergence Domain at $Z = 2.0m$

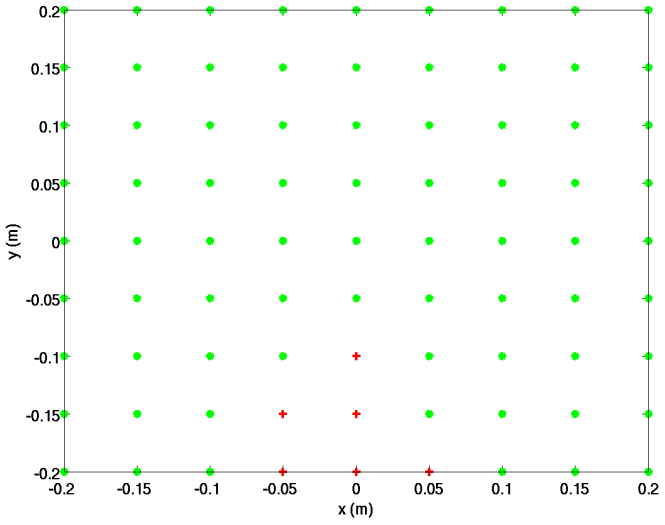
Figure 4.21 – Convergence results using pure luminance feature. Red cross : failed to converge



(a) Convergence Domain at $Z = 1.8m$



(b) Convergence Domain at $Z = 1.9m$



(c) Convergence Domain at $Z = 2.0m$

Figure 4.22 – Convergence results using WPM. Green dot : convergence, Red cross : failure

Table 4.1 – Convergence Rates with Synthetic Imagery

Experiment Set#	Non-weighted	Pure Luminance	Weighted Moments
1	0%	85.1%	96.29%
2	0%	85.1%	93.82%
3	0%	85.1%	92.59%
Average	0%	85.1%	94.23%

pure luminance feature is shown in Figure 4.21. In this case, convergence is at 85.1% for each set as tabulated in 4.1. In all the 3 sets of experiments, the convergence is better in the case of the weighted photometric moments and these are tabulated in Table 4.1. Also, a graphical representation with green dots denoting converged poses and red crosses denoting failure is given in Fig.4.22. This demonstrates that the WPM is clearly better from the point of view of convergence than the non-weighted case. Next, let us take a brief diversion to examine the failed cases. The small amount of failed cases with weighted moments resulted from the system getting attracted to an equilibrium point which is not the desired one. We note that the task function is still regulated to 0 but in this case, the pose would not be the desired one. This could be ascertained as well by an examination of the SSD error between the image at the end of the servo and the learnt image from the desired pose. In the failed case, this error was equal to 7037.3. The final image obtained at such an equilibrium point shown in Fig.4.19(f). Nevertheless, the weighted photometric moments exhibit better convergence properties in comparison to non-weighted moments. It should not be forgotten however that the visual servoing in general is always dependent on the scene observed by the camera. It is also interesting to study the behaviour of the weighted moments with textured targets, which is done next.

4.4.5.2 Using dense texture

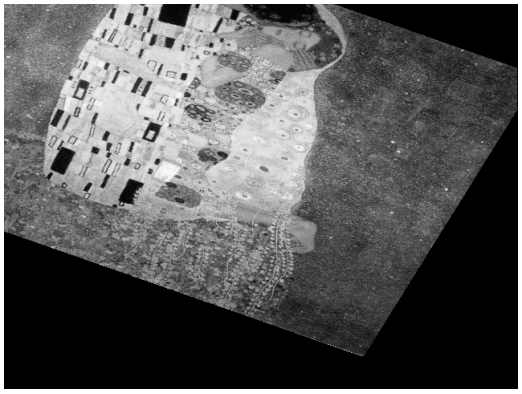
In this section, the dense texture shown in Fig.4.23(a) is placed in the scene. Such textures for example can be present on any planar surface of an object that needs to be manipulated by the robot. The image learnt from the desired pose is shown in Fig.4.23(a). Images from 4 different initial poses are shown in Figs.4.23(b) to 4.23(e). The classical VS control law was used with the interaction matrix computed at the desired pose, maintaining exactly the same conditions as for experiment in Sec.4.4.5.1. For these set of experiments with

Table 4.2 – Convergence Rates with dense texture

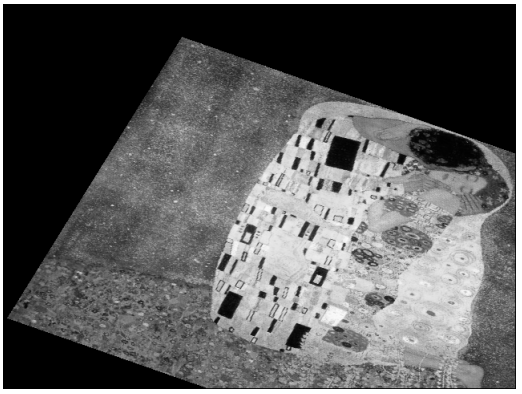
Experiment Set#	Non-weighted	Pure Luminance	Weighted Moments
1	51.85%	0%	100%
2	55.55%	0%	100%
3	60.49%	0%	96.29%
Average	55.96%	0%	98.76%

the dense texture, the pure luminance feature failed to converge in all the trials with an oscillatory behaviour at near convergence. This failure is probably due to errors in the image discretization at the borders of the *Klimt* texture. For coherence with the tabulated

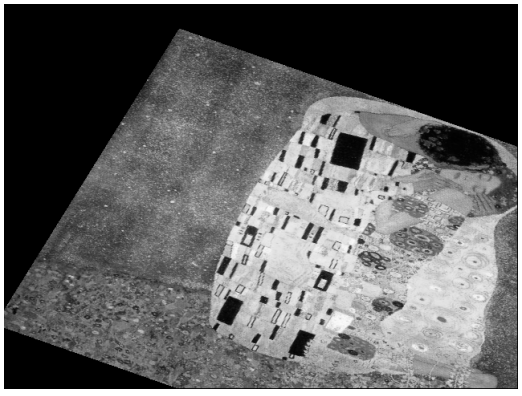
(a) Image learnt from the desired pose



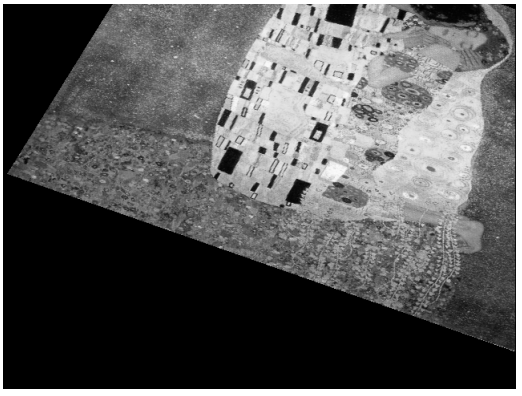
(b)



(c)



(d)



(e)

Figure 4.23 – A sampling of 4 different images from the generated initial poses are shown in 4.23(b)-4.23(e)

results for the synthetic case, the convergence rate has been marked 0% for pure luminance in Table 4.2. Let us note clearly that this should not be attributed directly to the extraneous regions in this particular case. But at the same time, it is interesting also to observe that this issue has not affected the weighted photometric moments. The results for the non-weighted moments are shown in Fig.4.24 and for the weighted photometric moments in 4.26. The convergence rates are tabulated in 4.2. The non-weighted moments have converged on an average only in 55% of the cases. In comparison, for the weighted moments, only 3 cases failed to converge out of 243 total runs with a very satisfactory convergence rate of 98%. In fact, in the first two sets of experiments, the weighted photometric moments converged for all the generated poses yielding a 100% convergence rate. No convergence to any undesired equilibrium points were observed, thanks to the textured object. The final accuracies for all the converged experiments was less than $1mm$ in translation and less then 1° in rotation. Based on the clear improvements in convergence rate, we conclude that weighted photometric moments are effective as a solution to the problem of extraneous image regions.

4.4.6 Experimental Results

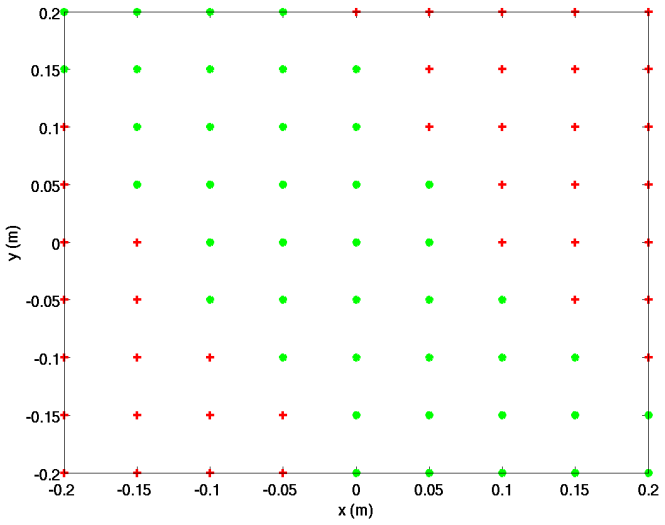
In this section, we report the experimental results obtained from the Viper850 platform configured for SCARA-type actuation. The desired camera pose is chosen vertically above the scene such that the image in Fig.4.27(a) is observed by the robot. The robot is then moved to the initial pose from where the robot observes the image shown in Fig.4.27(b). The displacement required between the initial and desired poses is ${}^{c*}\mathbf{t_c} = [-2.32cm, 0.97cm, -8.3cm]$ in translation and $\mathbf{R}_z(-11^\circ)$ rotation around the optic axis. Like in the simulations, there is no distinction between what is the foreground and what constitutes the background in the initial and desired images. The classical VS control law was used with a gain of $\lambda = 0.3$ and the interaction matrix $\mathbf{L}_{s*}$ at the desired configuration with the depth roughly approximated at 40cm.

4.4.6.1 Experiment I

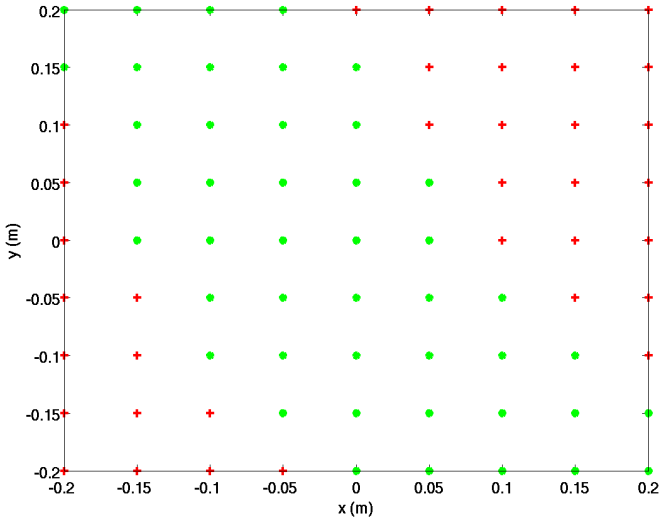
The form of the interaction matrix is similar to that in (4.46) and had a condition number of 818.39 with same invariance properties discussed in 4.4.3. The extraneous image regions have induced large variations in the error in a_n during the initial iterations as seen from Fig.4.27(c). Nevertheless, the system is driven to convergence by the generated control velocities shown in Fig.4.27(d). The final error in the achieved robot pose is $[-0.29mm, -2.21mm, -0.96mm]$ in translation and 0.1° around the optic axis respectively.

4.4.6.2 Experiment II

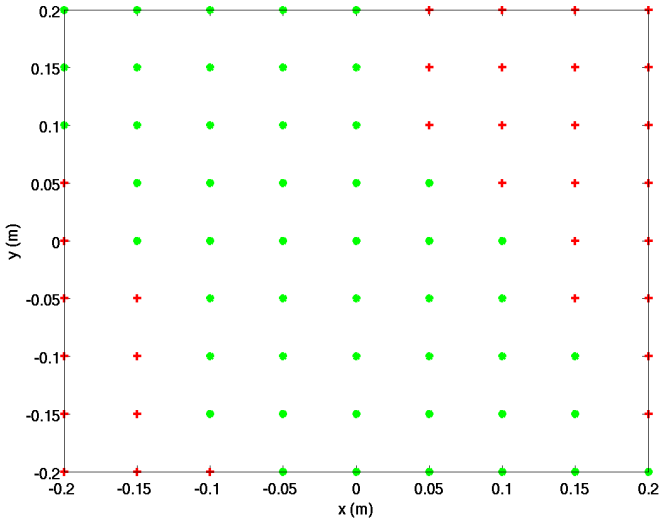
In this experiment, the same initial and desired poses as Experiment 4.4.6.1 is chosen, but with a different photometric target. The images acquired at the desired and initial poses are shown in Figures 4.28(a) and 4.28(b) respectively. The interaction matrix exhibited a condition number of 4435.2, higher than in the previous experiment. Indeed, the interaction matrix even for the same visual feature depends on the spatial distribution pattern of the intensities in the image. Further, from Figure 4.28(c), we find that the error in a_n is



(a) Convergence Domain at $Z = 1.8m$

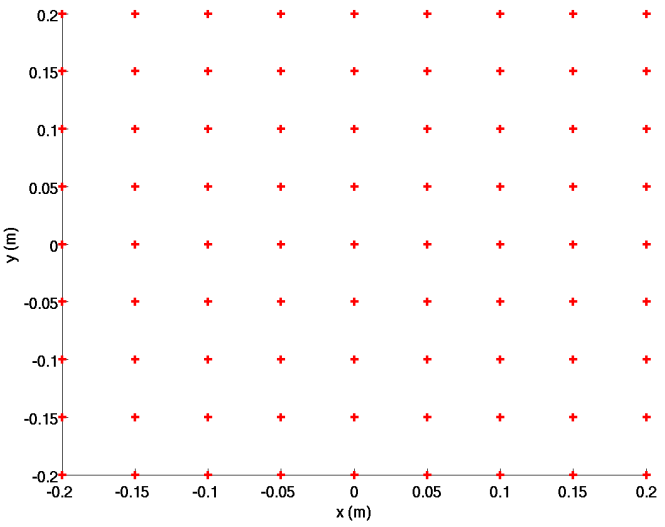


(b) Convergence Domain at $Z = 1.9m$

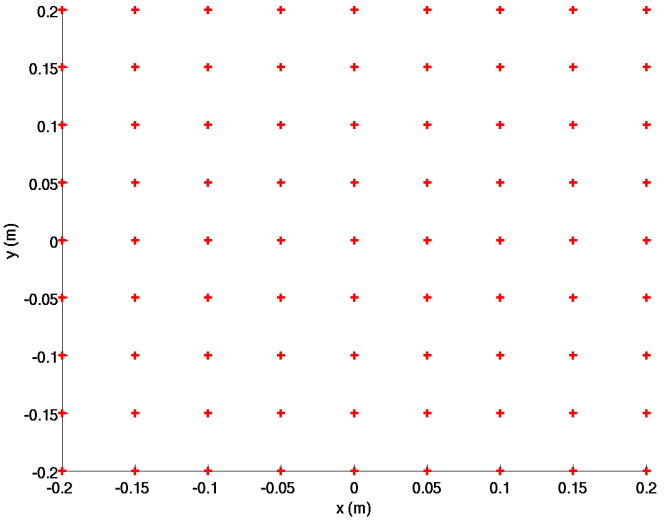


(c) Convergence Domain at $Z = 2.0m$

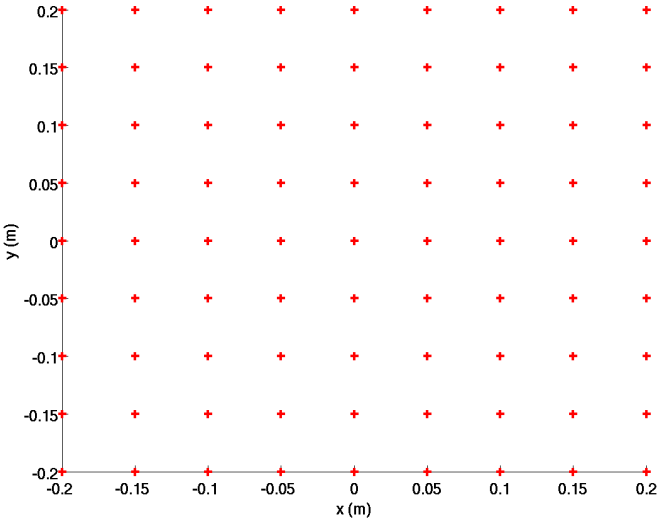
Figure 4.24 – Convergence Results for non-weighted moments. Green dot : convergence, Red cross : failure



(a) Convergence Domain at $Z = 1.8m$

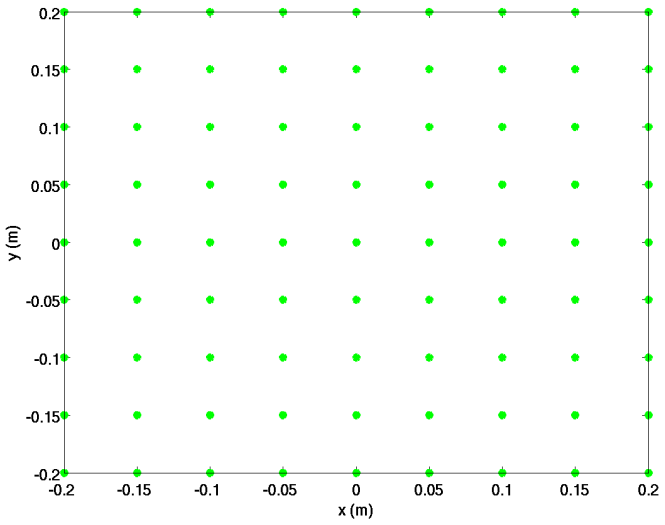


(b) Convergence Domain at $Z = 1.9m$

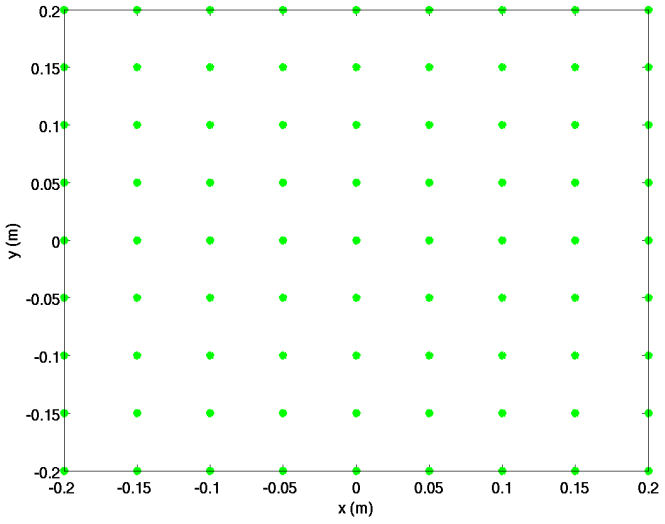


(c) Convergence Domain at $Z = 2.0m$

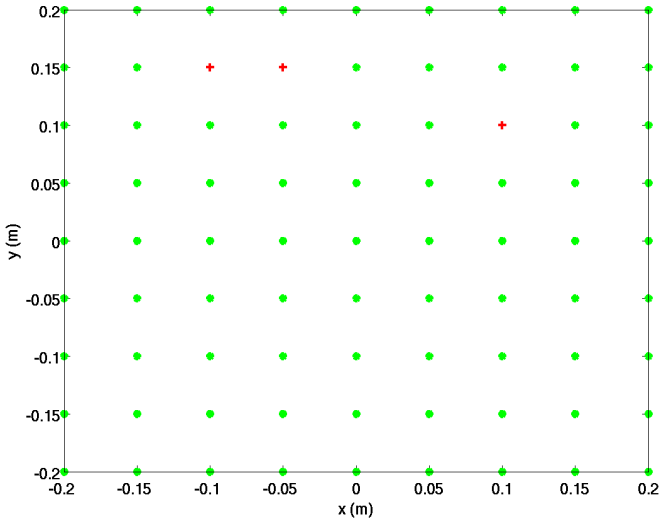
Figure 4.25 – Convergence results using pure luminance. Red cross : failed to converge



(a) Convergence Domain at $Z = 1.8m$



(b) Convergence Domain at $Z = 1.9m$

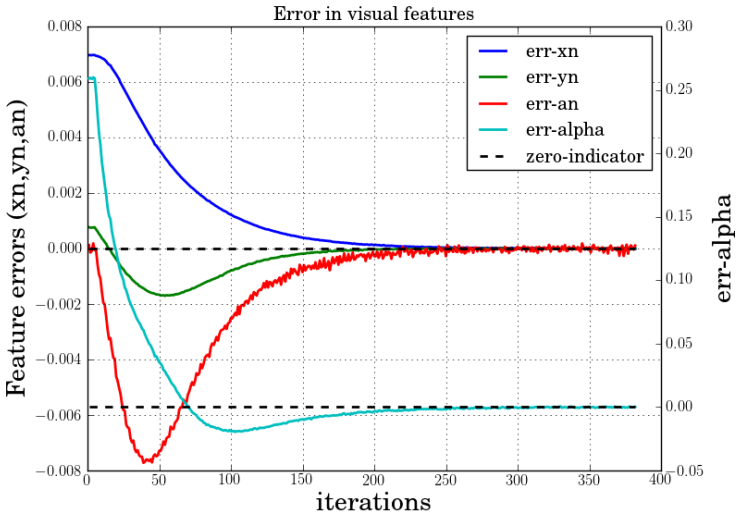


(c) Convergence Domain at $Z = 2.0m$

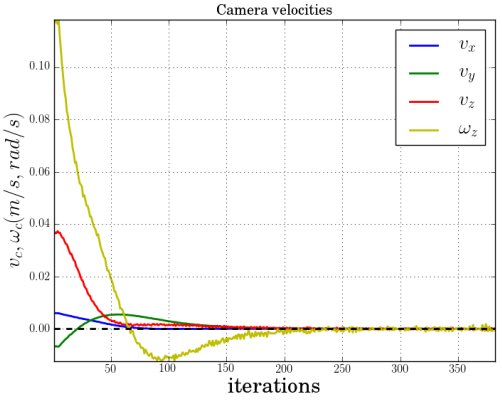
Figure 4.26 – Convergence results using WPM. Green dot : convergence, Red cross : failure

(a) Camera view at desired robot pose

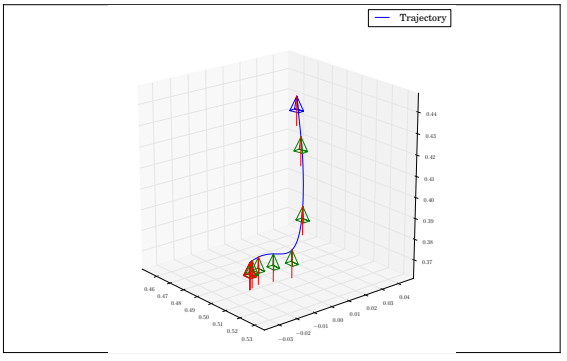
(b) Camera view at initial robot pose



(c) Errors in visual features (y-axis Multi-scale) x_n, y_n, a_n on left and α on right



(d) Camera velocities

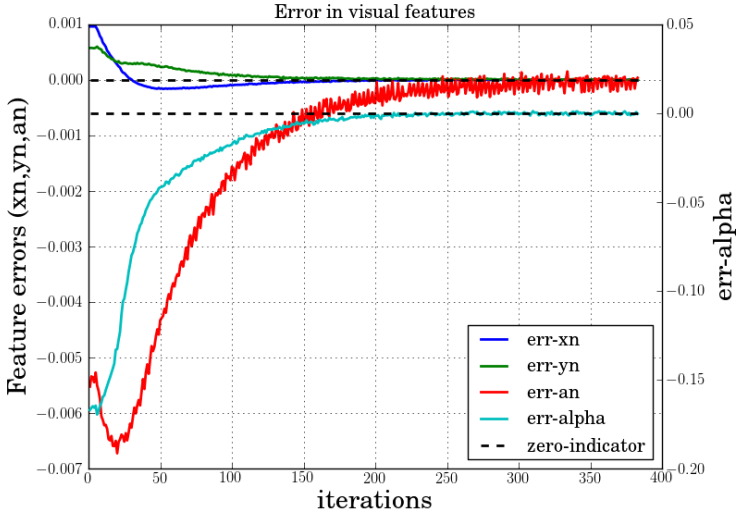


(e) Camera 3-D trajectory

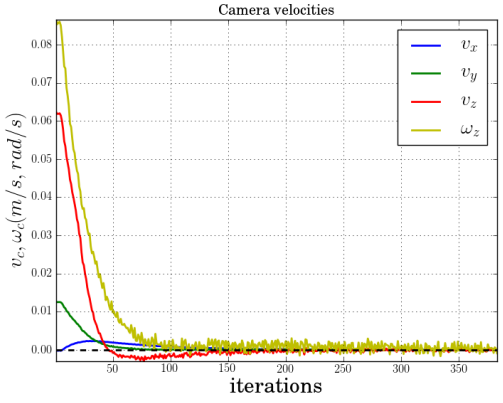
Figure 4.27 – Experiment I : Visual Servoing on the Viper850 robot with weighted photometric moments

(a) Camera view at desired robot pose

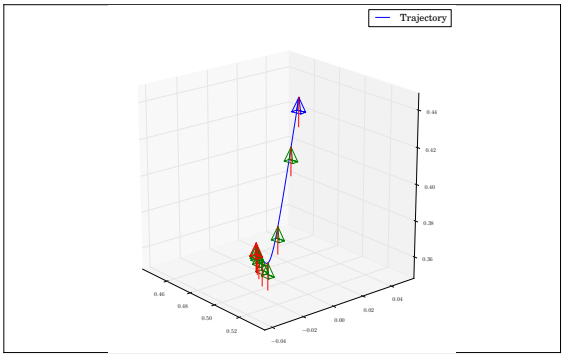
(b) Camera view at initial robot pose



(c) Errors in visual features (y-axis Multi-scale) x_n, y_n, a_n on left and α on right



(d) Camera velocities



(e) Camera 3-D trajectory

Figure 4.28 – Experiment II : Visual Servoing on the Viper850 robot with weighted photometric moments

more noisy than in the previous experiment. This is due to the fact that the noise effect depends on the area under the influence of the weighting function. This can also be attributed in part to the uncontrolled lighting conditions which can contribute to noise in the pixels. The accuracy in this experiment is $[0.84mm, -1.72mm, -0.95mm]$ in translations which is acceptable and an error of only $0.07°$ around the optic axis. The proposed scheme was thus validated by several successful experiments despite the loss in invariance. We recall that our scheme does not require any image matching or visual tracking processes but only the computation of a set of weighted moments on the image. Till now, several results that demonstrated the validity and usefulness of the weighted photometric moments were presented. However, the results were obtained from systems actuated for SCARA type motions involving 4DOF. Next, we shall deal with using weighted photometric moments with fully actuated 6-DOF systems.

4.5 Validation results for 6dof

4.5.1 Simulation Results

In this section, we showcase visual servoing results obtained using all the 6DOF. The desired pose is fixed at $1.2m$ vertically above the object. The desired and initial poses are chosen such that they allow the appearance/disappearance of portions of the scene from the camera image during the servo. A rotational displacement of $[5°, 5°, 25°]$ and a translational displacement of $[5cm, 12.5cm, 10cm]$ is required for convergence.

4.5.1.1 Experiment A

The image learnt from the desired pose is shown in Fig 4.29(b). The image at the robot starting pose is shown in Fig 4.29(a). In this image, it should be observed that a subset of pixels not present in the desired image have appeared.

(a) Camera view at initial robot pose

(b) Camera view at desired robot pose

Figure 4.29 – Initial and desired images pertaining to Experiment 4.5.1.1

The classical visual servoing control law $\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{-1}(\mathbf{s} - \mathbf{s}^*)$ is used. The same set of visual features used in Section 3.5.2 are employed, but are now computed with

the weighted photometric moments. The interaction matrix at the current configuration $\mathbf{L_s}(\mathbf{s}(t), Z(t))$ is used with the depths updated at each iteration of the control law. As mentioned previously for the experiments with the reduced DOF this choice permits the correctness of the modelling from the exponential decrease of the errors resulting from the control law. A gain of $\lambda = 1.5$ was used for all the experiments with the weighted photometric moments. Comparisons are performed between the servoing using visual features built from weighted photometric moments, using non-weighted moments and the pure luminance feature.

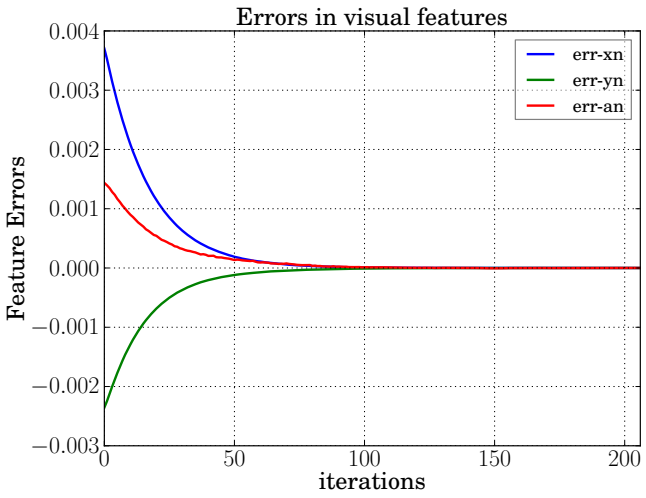
In the case of the weighted photometric moments, the resulting behaviour is very satisfactory. The errors decrease exponentially as shown in Figures 4.30(a) and 4.30(b). This confirms the correctness of the modelling steps used to obtain the interaction matrix of the weighted photometric moments. Naturally, the successful results also imply the correctness of the visual features obtained from the weighted moments. The velocity profiles generated with the weighted photometric moments is shown in Figures 4.31(c). Those profiles are expected since the current interaction matrix ensures that the decrease in errors in the visual features is exponential but does not have direct control over the Cartesian behaviour. Nevertheless, the resulting satisfactory behaviour can be observed from Figure 4.31(d).

Comparison to non-weighted moments : The same visual servoing task was repeated by employing the same control law but with the visual features computed from the non-weighted moments. The system in this case is perturbed by the appearance and disappearance of parts of the scene. The velocities generated are shown in Figure 4.31(a). This perturbation is reflected in the errors shown in Fig 4.30(c) and 4.30(d) respectively. Thus, we see that in this case, the extraneous regions have resulted in the worst case effect namely non-convergence to the desired pose.

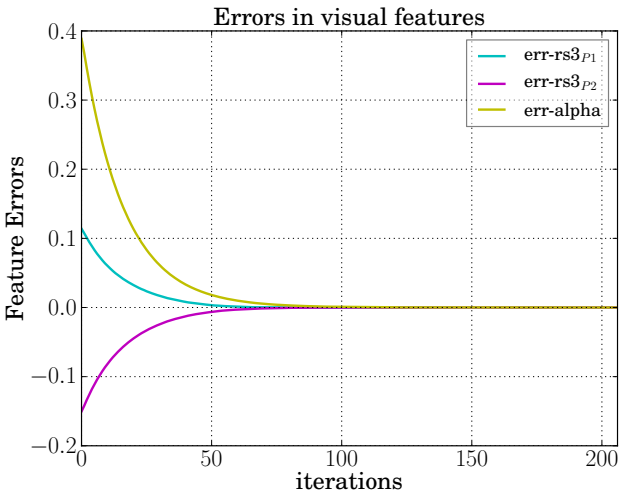
Comparison to pure luminance : Next, we also compared the weighted moments with the pure luminance feature with the same control law. A gain of $\lambda = 35$ was employed. Also in this case, the effect of the extraneous regions is severe and the control law does not converge to the desired pose. The generated velocities (see Fig 4.31(b)) do not regulate the errors satisfactorily. The error $\|\mathbf{I} - \mathbf{I}^*\|$ starts to increase rapidly as shown in Fig 4.30(f). This can be compared to the case of the weighted photometric moments where the error norm decreases exponentially as shown in Figure 4.30(e). Also, as mentioned previously, the visual features are redundant and there is no mapping of individual features to the actuated DOF. The servoing behaviour depends on the profile of the cost function, which is dependent on all the intensities in the acquired image. The appearance and disappearance of scene portions thus also affects the direct visual servoing method. Next, we discuss results obtained from servoing on a scene different from the one used in this experiment.

4.5.1.2 Experiment B

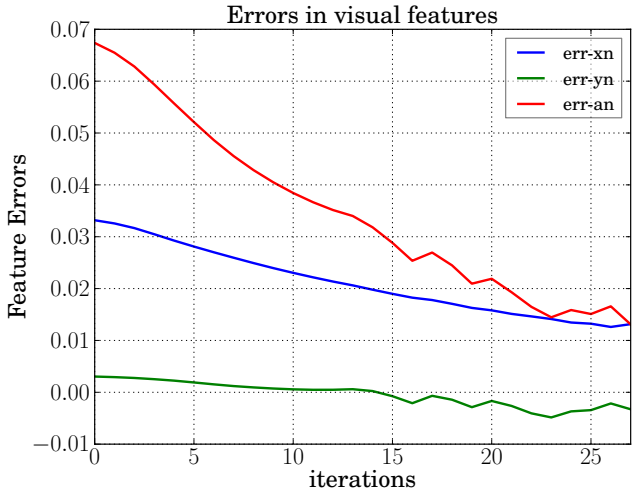
In this section, we performed the same experiments as in 4.5.1.1 but on a different scene similar to the one used in [Collewet 11]. The initial image is shown in Figure 4.32(a) and the image learnt from the desired pose in Figure 4.32(b). Also, the displacements required for convergence are the same as in Experiment 4.5.1.1. Again, we observe that the



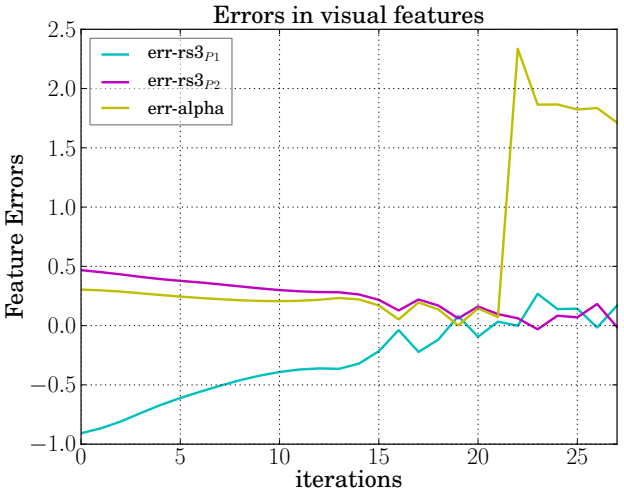
(a) WPM - Errors in features (x_n, y_n, a_n)



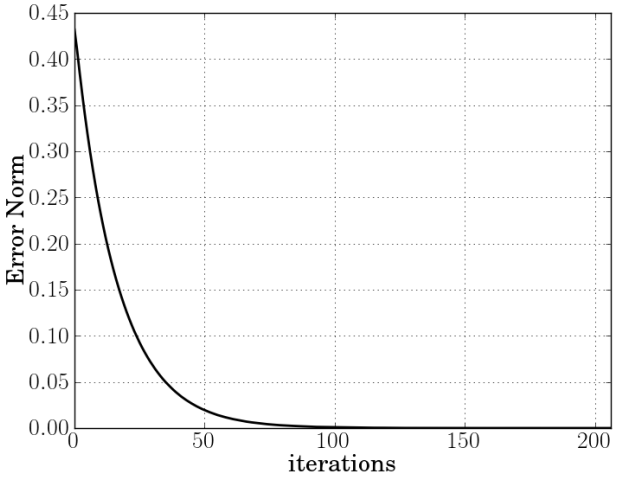
(b) WPM - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$



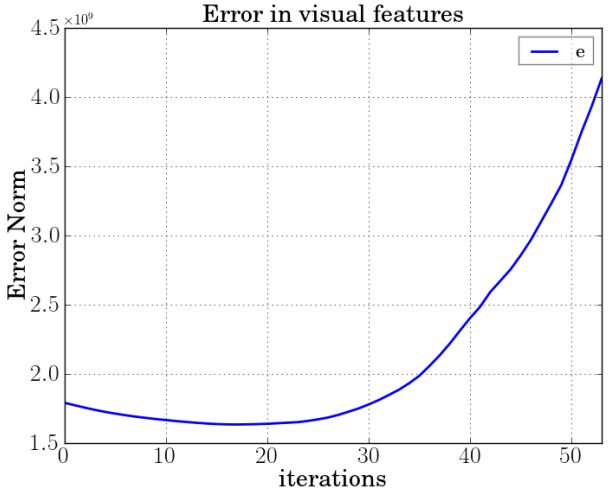
(c) Non-weighted moments - Errors in features (x_n, y_n, a_n)



(d) Non-weighted moments - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$

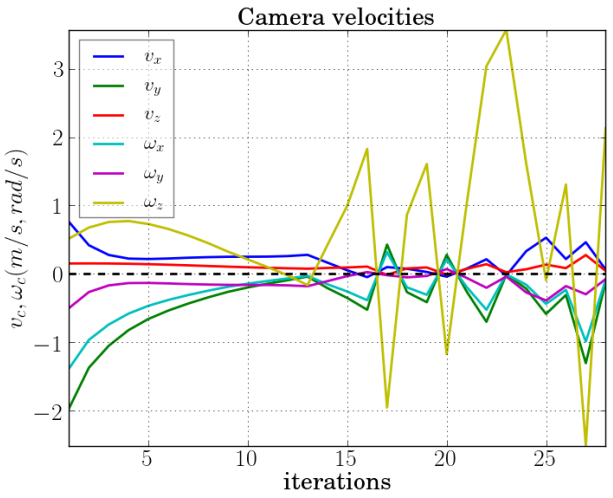


(e) WPM - Visual features error norm $\|\mathbf{s} - \mathbf{s}^*\|$

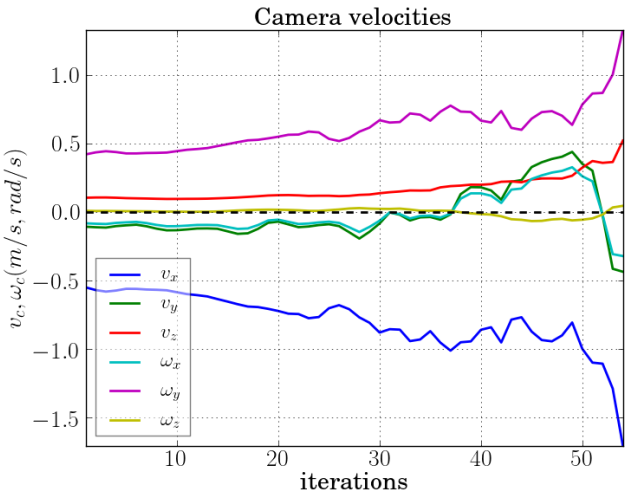


(f) Pure luminance - error norm $\|\mathbf{I} - \mathbf{I}^*\|$

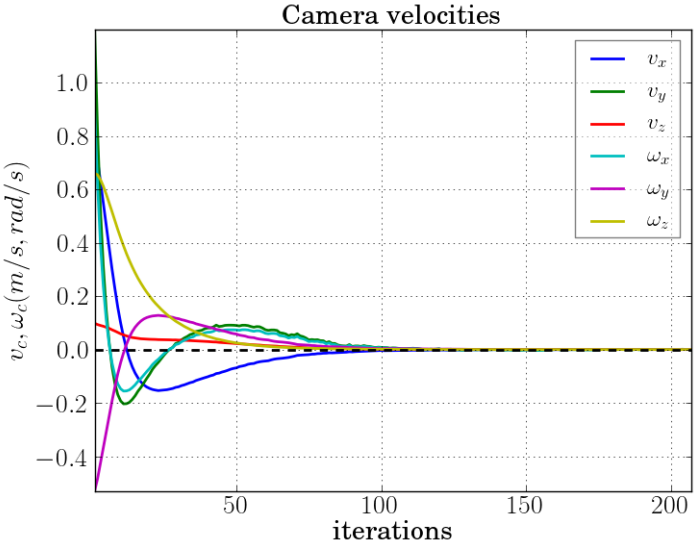
Figure 4.30 – Experiment 4.5.1.1 : 6DOF Visual Servoing with WPM, non-weighted moments and pure luminance feature on a domestic household scene



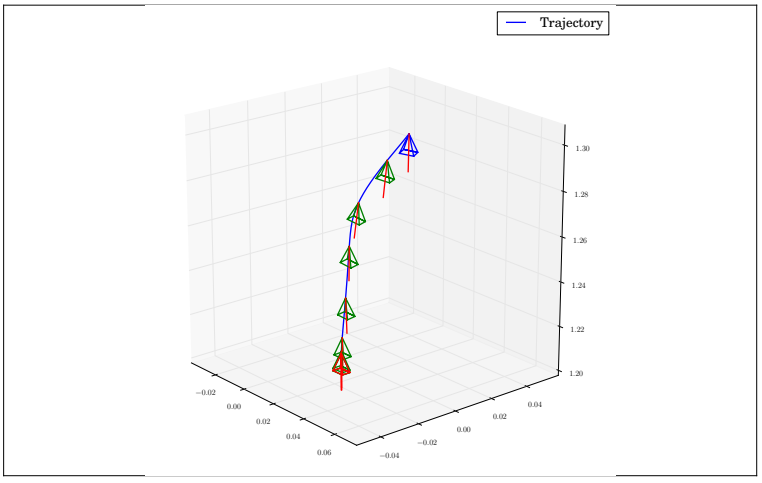
(a) Non-weighted moments - Camera velocities



(b) Luminance feature - Camera velocities



(c) WPM - Camera velocities



(d) WPM - Camera trajectory

Figure 4.31 – Experiment 4.5.1.1 : Control velocities and camera spatial trajectory

(a) Camera view at initial robot pose (b) Camera view at desired robot pose

Figure 4.32 – Initial and desired images pertaining to Experiment 4.5.1.2

weighted photometric moments exhibit a very satisfactory convergence. The exponential decrease in the feature errors are shown in 4.33(a) and 4.33(b). The generated control velocities are shown in Figure 4.34(c) and the camera spatial trajectory shown in Figure 4.34(d).

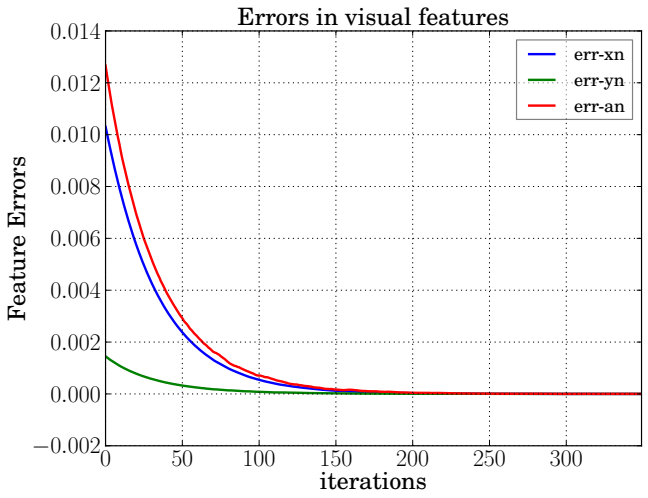
Comparison to non-weighted moments : As for the non-weighted moments, just like the previous experiment, they failed to handle the extraneous regions also in this case. From Fig 4.33(c) and 4.33(d), we can in fact observe that the errors do not decrease satisfactorily. The control is affected severely (see 4.34(a)) and the servo does not converge.

Comparison to pure luminance : In this experiment, the pure luminance features converged to the desired pose although the behaviour is not as satisfactory as the weighted photometric moments. We note that the extraneous regions did not have the worst effect of non-convergence to the desired equilibrium, which occurred in the previous experiment 4.5.1.1 and in the reduced DOF experiments as well. The extraneous regions still affect the control. The errors increase briefly midway (due to the unsatisfactory control velocities) before convergence to 0 as shown in Figure 4.33(f). The generated control velocities are shown in Fig 4.34(b). This results in a spatial trajectory that is more deviant than the case of the weighted photometric moments (see Figure 4.34(d)).

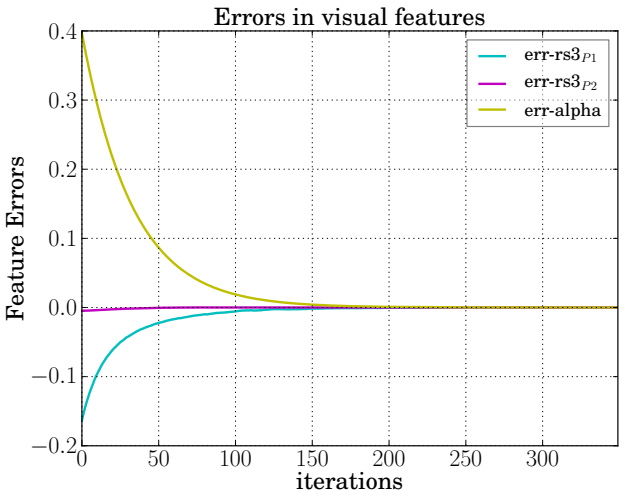
Thus, we see that visual servoing using the weighted photometric moments were more satisfactory than the non-weighted moments and the luminance feature in our experiments. This corroborates evidence from our previous results with reduced DOF about the effectiveness of the weighted photometric moments. From these results, we can conclude that weighted photometric moments are indeed better equipped than moments-based method and the pure luminance feature when it comes to handling the appearance and disappearance issue.

4.5.2 Experimental Results

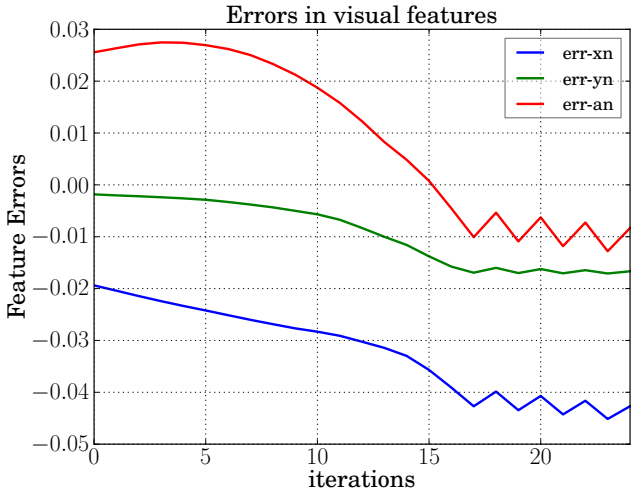
In this section, we present experimental results obtained from the 6dof Viper platform using the weighted photometric moments. For the following experiments, the classical



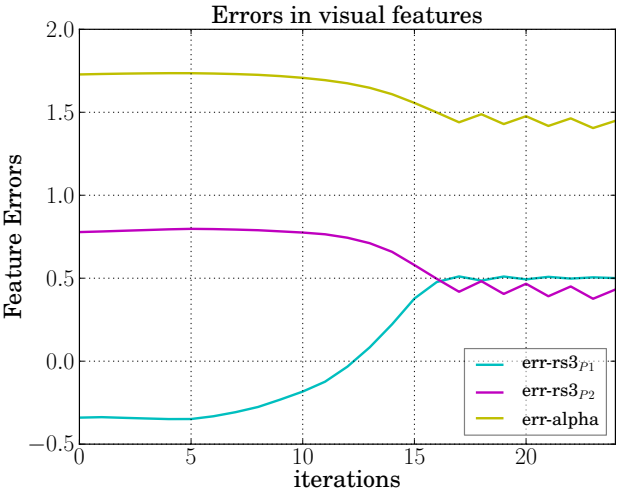
(a) WPM - Errors in features (x_n, y_n, a_n)



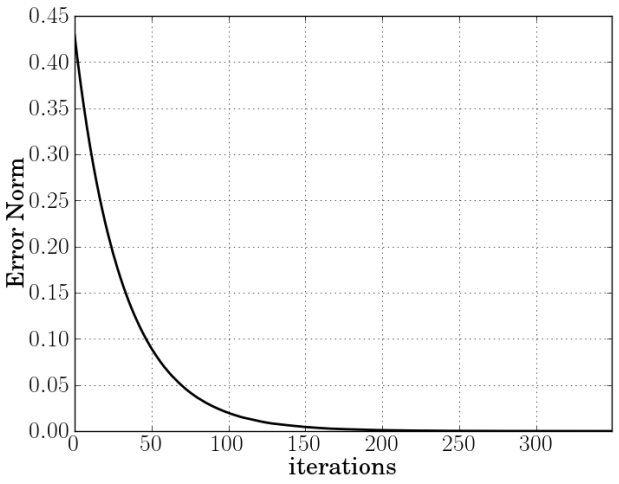
(b) WPM - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$



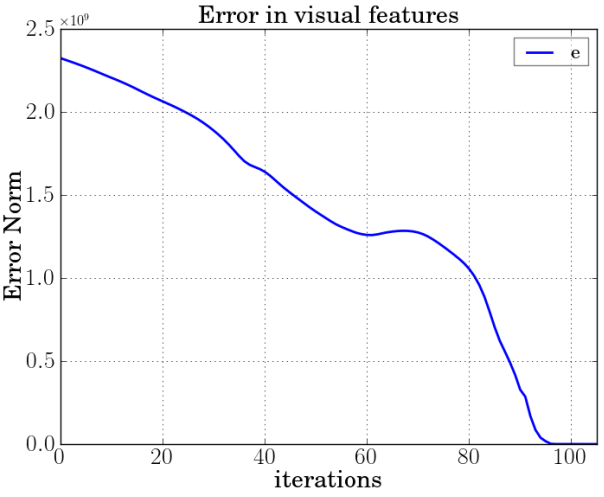
(c) Non-weighted moments - Errors in features (x_n, y_n, a_n)



(d) Non-weighted moments - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$

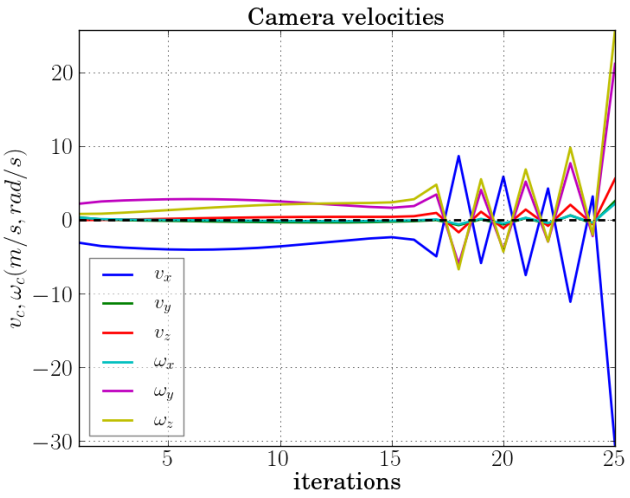


(e) WPM - Visual features error norm $\|\mathbf{s} - \mathbf{s}^*\|$

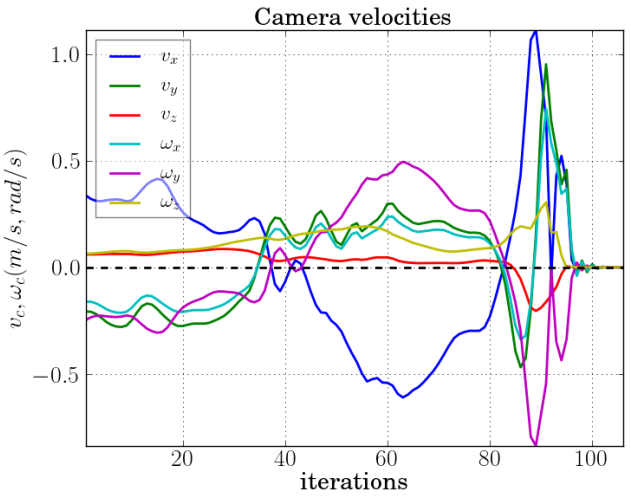


(f) Pure luminance - error norm $\|\mathbf{I} - \mathbf{I}^*\|$

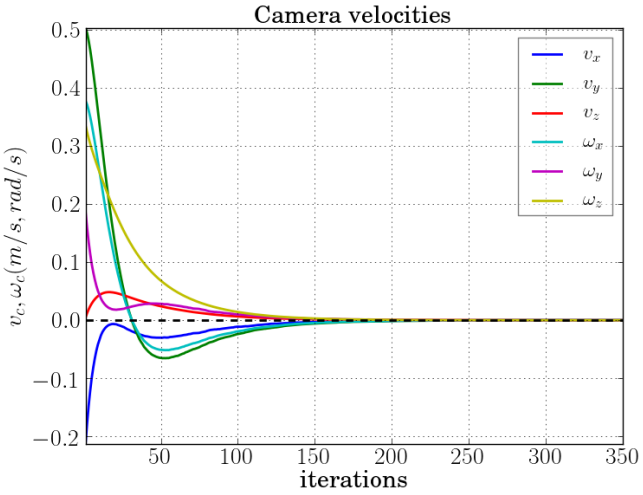
Figure 4.33 – Experiment 4.5.1.2 : 6DOF Visual Servoing with WPM, non-weighted moments and pure luminance feature on Bogart image



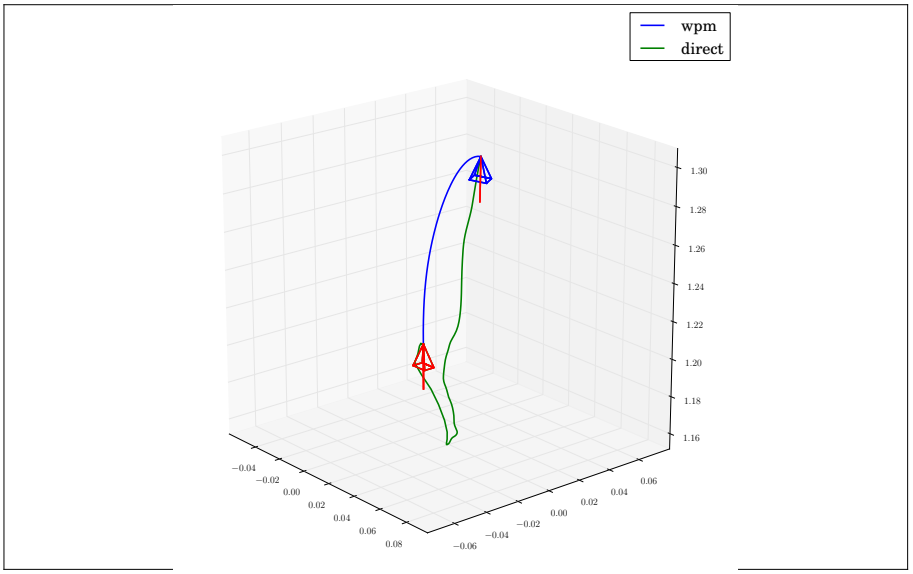
(a) Camera velocities using non-weighted moments



(b) Camera velocities using pure luminance as visual feature



(c) Camera velocities using WPM



(d) Spatial trajectory comparison between WPM and pure luminance

Figure 4.34 – Experiment 4.5.1.2 : Control velocities and camera spatial trajectory

visual servoing control law $\mathbf{v}_c = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^{+}(\mathbf{s} - \mathbf{s}^*)$ has been used again but with the interaction matrix at the mean configuration. As explained previously, this configuration was shown to better approximate the camera displacements with a potentially large convergence domain [Malis 04]. A gain of $\lambda = 0.2$ was used with a roughly approximated depth of $Z^* = 0.5m$. Exactly the same set of visual features based on the weighted photometric moments used for the simulation results in the previous section have again been used here as well. For the experimental results, inspired from [Collewet 11], a target that does not have a very rich texture is chosen.

4.5.2.1 Experiment I

The image learnt from the desired pose of the robot is shown in Figure 4.35(b). The image acquired from the initial pose is shown in Figure 4.35(a). Let us note that *Lauren Bacal* present in the left part of the desired image is completely absent from the initial image. The difference image at the start of the servo is shown in Figure 4.35(c). A translational displacement of $[-0.35cm, -1.13cm, 6.67cm]$ and rotations of $[0.33°, 1.05°, 12.82°]$ are required for convergence to the desired pose. Even with the appearance and disappear-

(a) Camera view at initial robot pose (b) Camera view at desired robot pose

(c) Difference image at start of servo

Figure 4.35 – Initial and desired images pertaining to Experiment 4.5.2.1

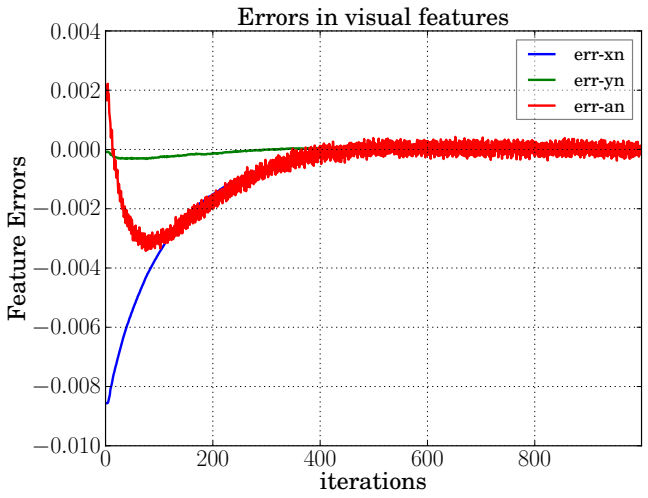
ance of scene portions in the image, the visual servoing converged to the desired pose. The satisfactory Cartesian behaviour can be observed from Figure 4.36(e). The error in features related to control of rotational motions is very satisfactory (see 4.36(b)). On the

other hand, from the error decrease in features related to control of translational motions in Figure 4.36(a). it can be seen that the error in feature a_n is noisy. This feature is based on the area moment m_{00} directly related to the quantity of intensities in the acquired image. As expected, during the robot movement, there are a set of pixels which appear and others that leave from the field of view. This influx of information is severe especially at the border regions and is gracefully handled thanks to the improved modelling used by the weighted photometric moments. At the same time however, the appearance and disappearance phenomenon also occurs in regions other than the borders during the robot motion. This might sometimes contribute to some noise in the features. It is also to be noted that when the interaction matrix is updated at each iteration (for the mean configuration in this case), this noise in the features sometimes make the velocities noisy as well (see Figure 4.36(d)). However, this noise does not affect the satisfactory convergence as evidenced by our results. The difference image at the end of the servo is shown in Figure 4.36(c). The final accuracy in translations is $[-0.05mm, 1.1mm, 0.08mm]$ and for the rotations is $[0.18°, 0.006°, -0.019°]$.

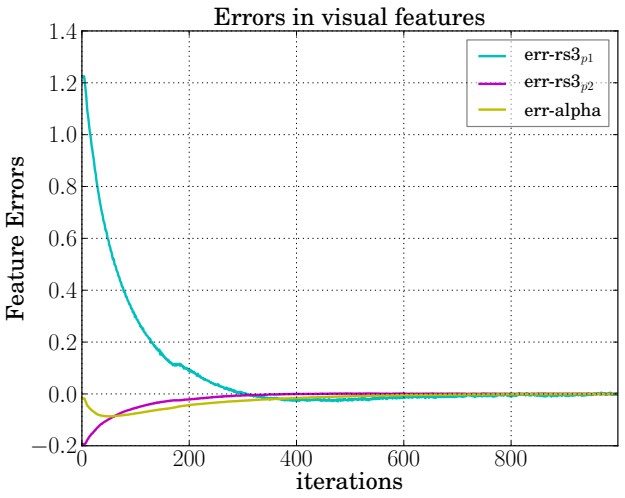
4.5.2.2 Experiments II and III

For the desired pose, the robot is stationed with the camera roughly at $50cm$ in front of the target as shown in Figure 4.37(a). The image learnt from this pose is shown in Fig 4.29(b). For the first experiment, the robot is moved to an initial pose as in Figure 4.37(c). The camera and target planes are not parallel to each other. There is a rotation of the camera around the y axis. At this initial pose, the image in Fig 4.37(d) is observed by the camera. In this case, a translational displacement of $[-11.16cm, -0.04cm, 0.02cm]$ and rotations of $[-0.05°, 11.37°, 0°]$ are required for convergence to the desired pose. The servo converged to the desired pose with decrease in errors as shown in Figures 4.38(a) and 4.38(b). The Cartesian behaviour is shown in 4.38(e). As for the final accuracy in this case, we obtained $[-0.09mm, 1.8mm, 0.28mm]$ and for the rotations $[0.29°, 0.11°, -0.01°]$.

For the second experiment, we chose a different initial pose as shown in Figure 4.37(f). Again, the camera and target planes are not parallel. Let us note that in this case the rotation is a large rotation of $20.1°$ around the x axis. The image at the start of the servo is shown in Figure 4.37(g). Some extraneous scene portions have appeared especially at the right border. The difference image at the beginning of the servo is shown in Figure 4.37(h). In this case, a translational displacement of $[-0.05cm, 17.02cm, 0.07cm]$ and rotations of $[20.1°, 0.12°, -0.02°]$ are required for convergence to the desired pose. The servo converged again in this case. From Figure 4.39(a), the feature a_n is noisy as before, reasons for which have already been previously explained. Also, from Figure 4.38(b), the error in one of the features rs_3 which is intended to control the x rotation increases briefly before decreasing again. This is because of the large rotation of $20°$ around x axis. This causes only a very small rotational motion around the y axis during that brief period, which is negligible compared to the overall motion required for convergence. The physical robot motion is satisfactory and this does not affect at all the convergence. The difference image at the end of the servo is shown in Figure 4.37(h). The final accuracy obtained is $[-0.20mm, 1.7mm, 0.09mm]$ for translations and $[0.18°, 0.02°, 0.03°]$ for the rotations. In both the above cases, the accuracy in the y translation component could be better. As suggested before, one strategy to attain better accuracies is to switch to the pure

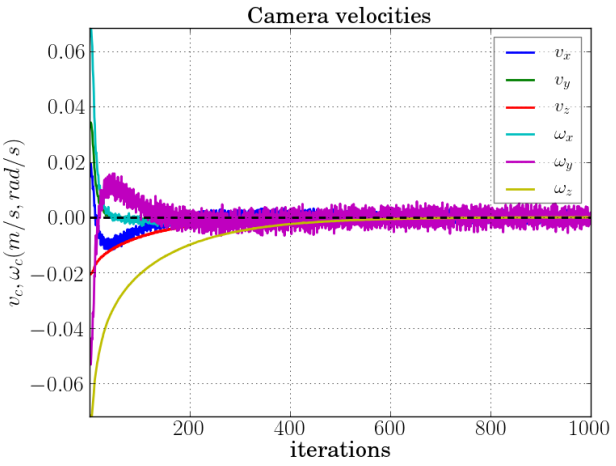


(a) WPM - Errors in features (x_n, y_n, a_n)

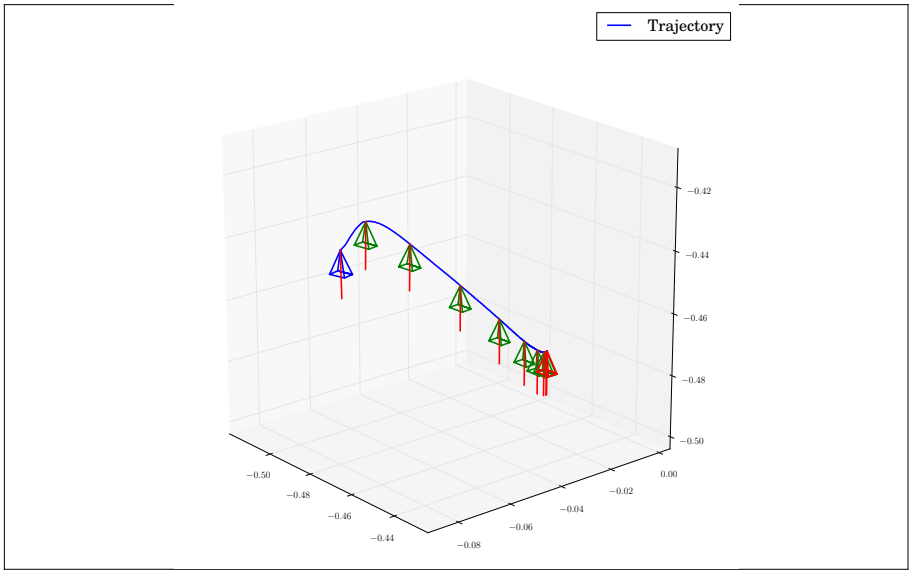


(b) WPM - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$

(c) Difference image at final pose



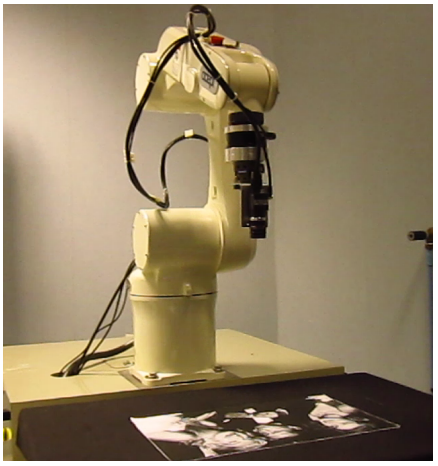
(d) Camera velocities



(e) Spatial trajectory

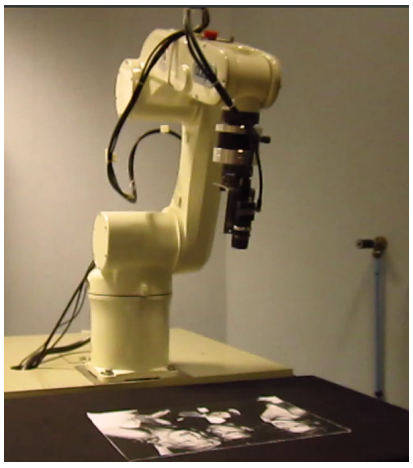
Figure 4.36 – Servoing results pertaining to Experiment 4.5.2.1

luminance based feature at near-convergence.



(a) Viper robot at desired pose

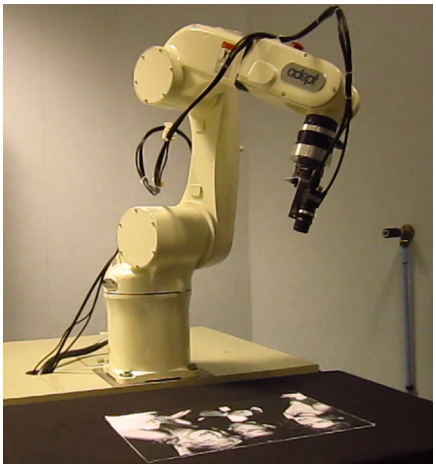
(b) Camera view at desired robot pose



(c) Viper robot at initial pose 1

(d) Camera view at initial robot pose

(e) Difference image at start of servo

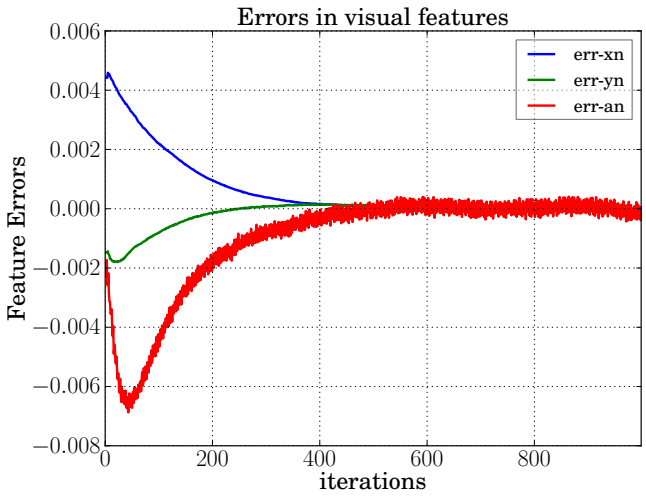


(f) Viper robot at initial pose 2

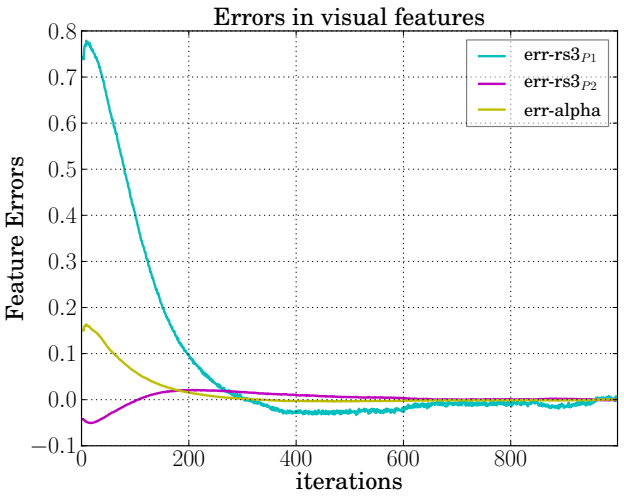
(g) Camera view at initial robot pose

(h) Difference image at start of servo

Figure 4.37 – Robot setups pertaining to Experiment 4.5.2.2

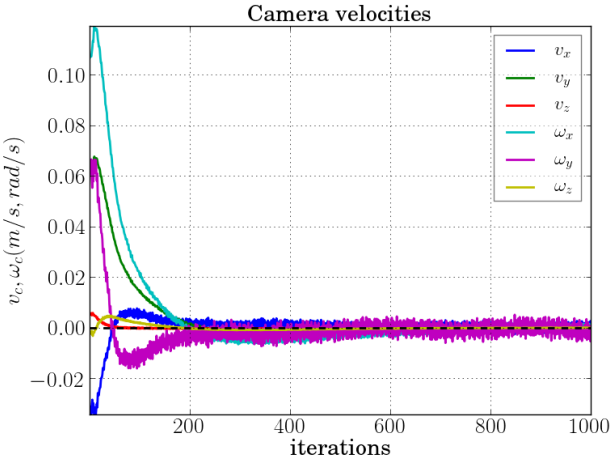


(a) WPM - Errors in features (x_n, y_n, a_n)

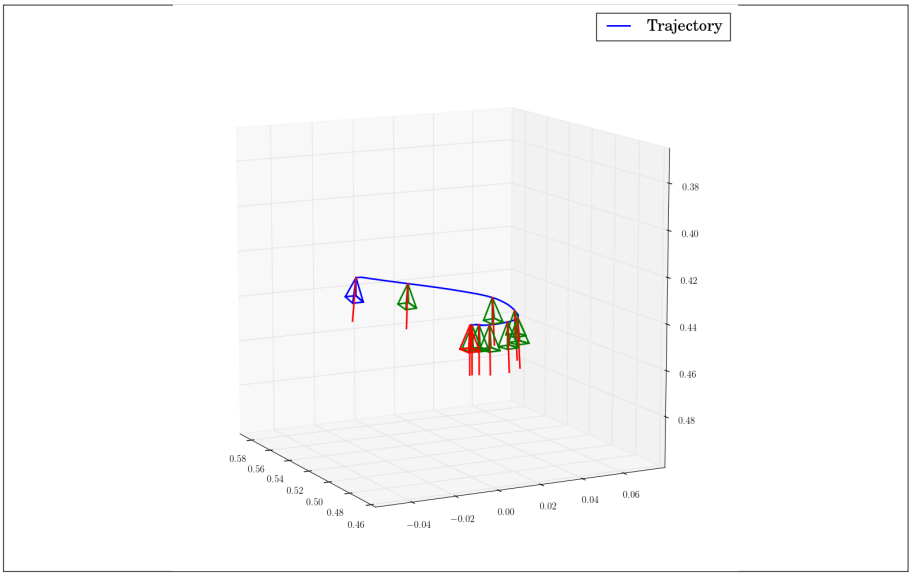


(b) WPM - Errors in $rs3_{p1}, rs3_{p2}, \alpha$

(c) Difference image at final pose

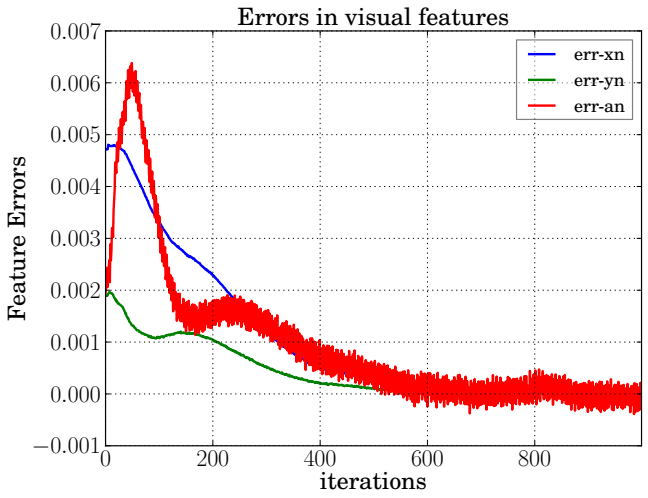


(d) Camera velocities

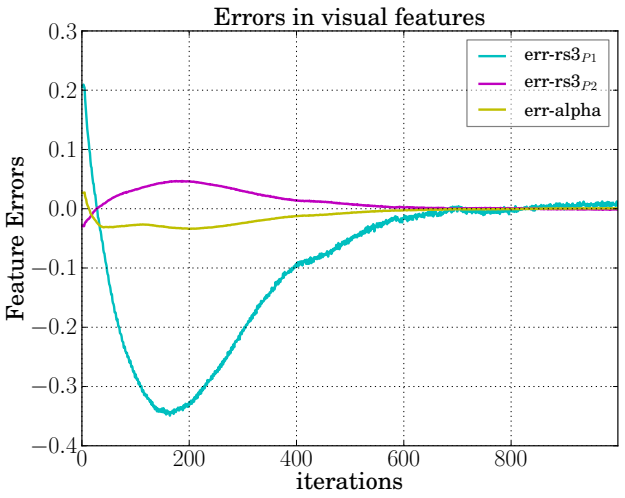


(e) Spatial trajectory

Figure 4.38 – Servoing results pertaining to 4.5.2.2

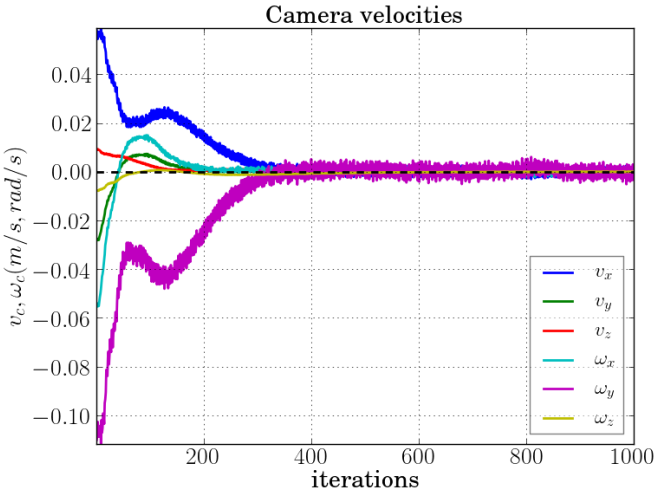


(a) WPM - Errors in features (x_n, y_n, a_n)

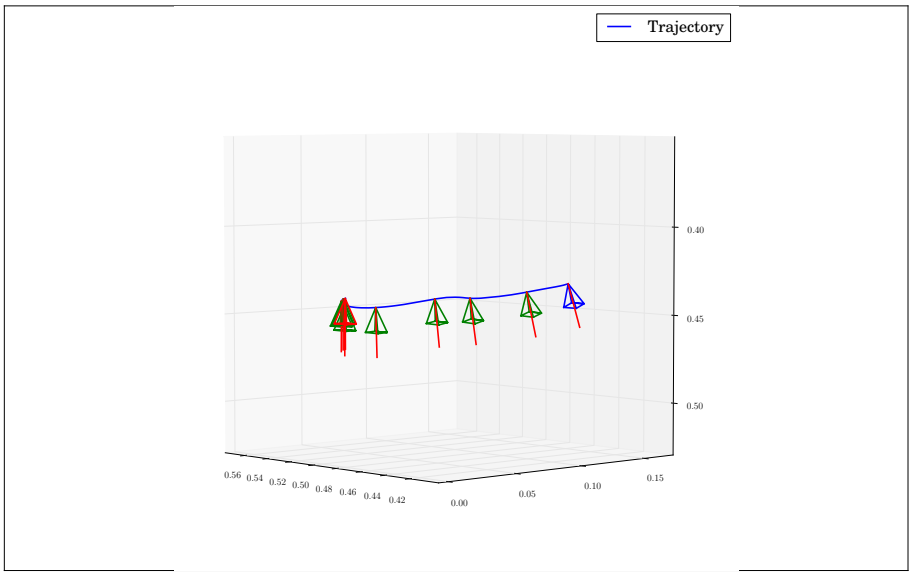


(b) WPM - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$

(c) Difference image at final pose



(d) Camera velocities



(e) Spatial trajectory

Figure 4.39 – Servoing results pertaining to 4.5.2.2

4.5.2.3 Experiment IV

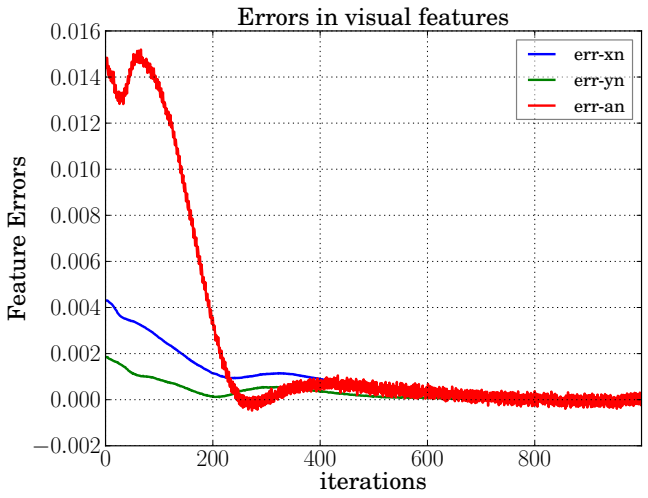
In this section, we present a visual servoing experiment which involved a complex motion to be executed. The initial image is shown in Figure 4.40(a) and desired image in Figure 4.40(b). The large error present can be easily observed from the difference image in Figure 4.40(c). For this experiment, a translational displacement of $[-8.78cm, 9.04cm, -4.34cm]$ and rotations of $[10.12°, -7.52°, -16.66°]$ are required for convergence to the desired pose. The resulting decrease in feature errors can be observed from Figures 4.41(a) and

(a) Camera view at initial robot pose (b) Camera view at desired robot pose

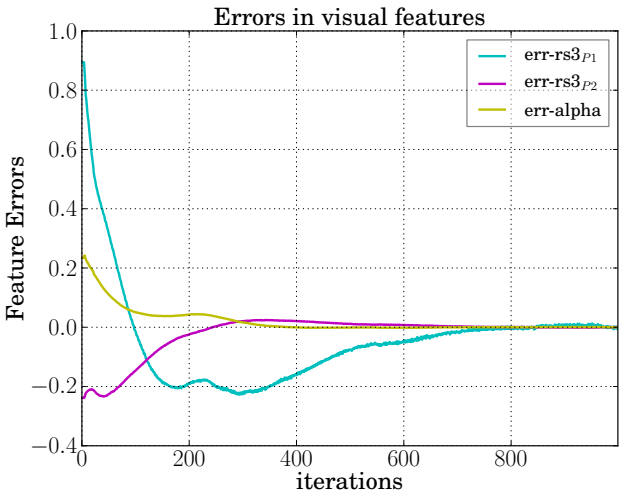
(c) Difference image at start of servo

Figure 4.40 – Initial and desired images pertaining to Experiment 4.5.2.3

4.41(b). We observe that the error profiles are regulated to 0. The generated control velocities are shown in Figure 4.41(d). There is not a completely monotonous decrease because a complex motion is required with a strong effect due to the appearance and disappearance phenomenon. In this case, during the visual servoing, the cost function to be minimized has a complicated surface (difficult to visualize) which is a function of not only the camera movement but also the extraneous regions. Further, we remind also that the interaction matrix does not have a decoupled structure because of the weighting strategy introduced in the moments formulation (this is true in all the previous cases as well). As discussed in the theoretical developments, regaining the invariant properties for the weighted photometric moments can bring more improvements in this situation. The rotational accuracies obtained are less than $[0.24°, 0.23°, 0.0005°]$ while for the translations, we obtained $[-1.8mm, 1.1mm, 0.21mm]$. As discussed previously, a very good strategy

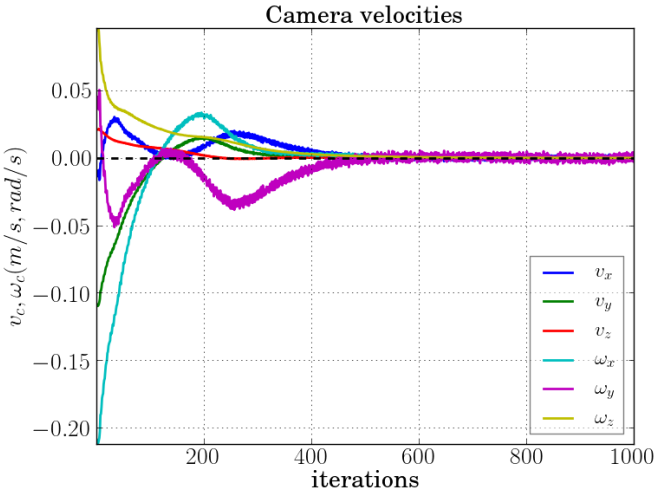


(a) WPM - Errors in features (x_n, y_n, a_n)

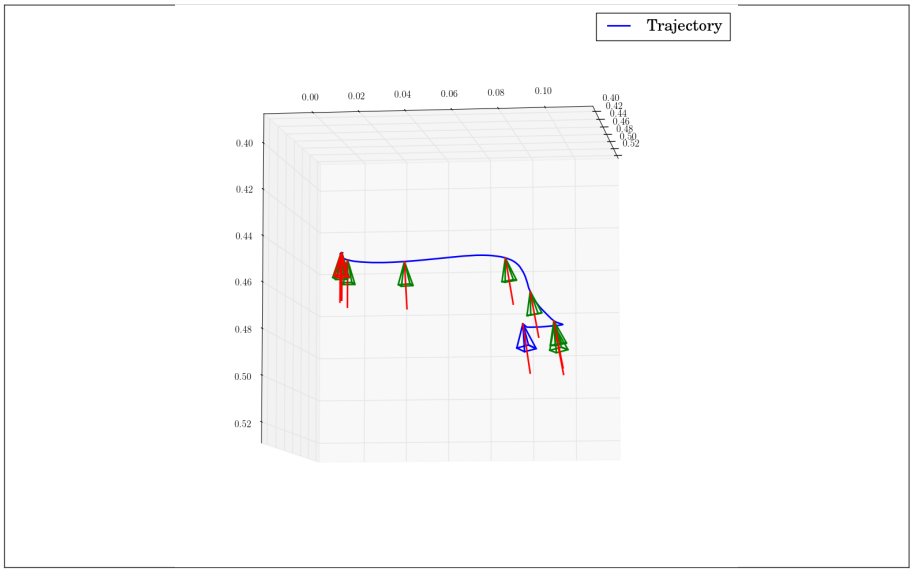


(b) WPM - Errors in $rs_{3_{p1}}, rs_{3_{p2}}, \alpha$

(c) Difference image at final pose



(d) Camera velocities



(e) Spatial trajectory

Figure 4.41 – Servoing results pertaining to 4.5.2.3

would be to combine weighted photometric moments and the pure luminance feature. The servo could be started with visual features built from the weighted photometric moments and at near-convergence, switch to the pure luminance feature. This should in principle result in an enhanced convergence domain due to the use of the moments and simultaneously, better final accuracies resulting from the redundancy in the pure luminance feature.

4.6 Conclusions

In this chapter, the important practical problem of appearance and disappearance of portions of the scene in the camera image during the visual servoing was treated. An modelling scheme for photometric moments that accommodates spatial weights was presented to solve this problem. A carefully selected weighting function was introduced into the formulation of photometric image moments and the interaction matrix obtained in analytical form. The newly formulated moments preserved their properties with respect to rotations around the optic axis, thanks to the ingenious choice of the weighting function whereas invariance with respect to other degrees of freedom were not retained. Both simulations and experimental tests conducted on a real robot confirm the validity and usefulness of the proposed method. Further, convergence tests from different initial poses were conducted and the results demonstrate the effectiveness of the weighted photometric moments in solving this problem. Also, visual servoing was demonstrated without using any visual tracking or image processing steps in the control loop.

One particular theoretical issue that we are aware of is the loss of invariance properties with respect to all motions except for optic axis rotations due to introduction of the spatial weights in the moments definition. To design weighting schemes such that the moments are invariant with respect to specific motions is an open and difficult problem. Despite this issue, we note that the results obtained have been very promising with this technique. Also, when the displacements between the initial and desired poses increase, there is less overlap between the images. The appearance and disappearance in that case is more important. Even if not strong enough to disturb convergence to the equilibrium, the specified exponential decrease in task errors cannot be obtained from the control scheme. In such cases, instead of static spatial weights, a temporally varying weighting function that adapts itself based on the overlap area would be attractive.

Chapter 5

Conclusions and Perspectives

5.1 Conclusions

In this work, the concept of tunable visual features for the simultaneous control of rotations around the x and y axes was introduced. A few selection criteria were proposed to select the features such that the servo is optimal with respect to maximal responsiveness, orthogonality and conditioning in the resulting interaction matrix. The methodology developed finds applications in image-moments based VS methods. More importantly, the proposed approach opens interesting possibilities for future research.

To the best of our knowledge, this work is one of the earliest attempts to capture the image intensities not directly but in the form of image moments. An analytical model was developed for obtaining the interaction matrix of the photometric moments. In these developments, an important assumption about the image borders, known as the zero border assumption (ZBA) was made. The results from the theoretical developments are consistent with the findings in the current state of the art in moments-based VS. This allows the utilisation of several existing results, notably those ones where the invariance properties of image moments-based functions are exploited for visual servoing. As a consequence, decoupled control laws can be designed with the photometric moments and they are endowed with a large convergence domain. With photometric moments, which are global features, spatial segmentation is avoided and the image processing is reduced to a simple and systematic computation of the moments on all the image plane. Thanks to the theoretical developments in this work, even the only image processing step of computing the image gradients necessary in the pure photometric visual servoing method is eliminated. Furthermore, the availability of well-segmented regions or a point set in the observed scene is no longer necessary.

Then, the important practical problem of parts of the scene appearing and leaving from the camera field of view was dealt with. The changes in the image due to these extraneous regions disturb the control in dense VS methods like the pure photometric method and in specific, moments-based approaches. An improved modelling scheme has been proposed in this work that gracefully handles the effects occurring due to the extraneous regions. A

carefully chosen weighting function is introduced into the formulation of the photometric image moments. This function attributes maximum importance to the pixels near the central parts of the image and smoothly reduces this importance towards the minimum for pixels near the periphery of the image. This weighting strategy naturally results in the elimination of the restrictive ZBA assumption discussed earlier. These improved moments called weighted photometric moments, preserve their properties with respect to rotational motions around the optic axis. However, the invariance with respect to motions along the other degrees of freedom is not retained. Nevertheless, the results demonstrate that the weighted photometric moments approach is effective in handling the extraneous regions issue and is quite promising for adoption into future works.

5.2 Perspectives

Alternate Metrics Concerning the tunable visual features proposed in this work, various selection criteria were proposed for their selection. This opens some interesting possibilities for further research. The first obvious research question that can be posed is: Are there alternate criteria that can be used to ensure invariance, best image space behaviour, Cartesian behaviour, etc. Second, this work evaluated the proposed cost criteria with the image learnt from the desired pose. Alternatively, it should be interesting to incorporate into these criteria, information computed from both the current image and the image learnt from the desired pose. Since the former is a snapshot of the current robot state, it can help to design better control laws. The information-theoretic method [Delabarre 12], for instance, modifies the current visual feature using visual information from the desired image.

Search for visual features In this work, with regards to simultaneous control of rotational motions around x and y, only a small subset of the existing moment invariants-based visual features were tested and one of them was adopted. While the results are satisfactory, we still cannot conclude that there is a visual feature set that is perfect for all visual servoing applications or scenarios. Somehow the problem of finding a *perfect* set of 6 visual features such that each of them is associated to a single spatial degree of freedom remains open. In this case, the interaction matrix would be identity $\mathbf{I}_6$, which is an ambitious goal to achieve.

Establishing Invariance with Weighting Strategy We saw that the proposed weighting strategy alters the invariance property of the photometric moments to motions other than the rotation around the optic axis. Removing this shortcoming is an open problem. This is important but difficult to solve at the same time, considering that a considerable advance in theory concerning the representation of the image and modelling the visual features is required.

Combining Photometric Moments with pure photometric visual servoing It could be an advantageous strategy to combine the visual features obtained from the photometric moments with the pure luminance feature. Typically, we can switch the control from

moments to pure luminance after the errors in the system have been regulated to a specific threshold. In this case, we can benefit from an improved convergence domain and excellent accuracies upon convergence.

Image Moments for Visual Tracking To the best of our knowledge, only few works [Tahri 05b] [Komuro 07] investigated image moments towards visual tracking and pose estimation applications. These works tracked the position and orientation of an object whose shape was known apriori. It might be interesting to use photometric moments to define similarity measures and investigate their suitability for pose estimation and visual tracking. Based on the nice results obtained with the weighted photometric moments, we can expect that they can be used for visual tracking at the minimum for simple classes of warps.

Singularity Analysis In the case of visual servoing where a set of 3 image points were used as visual features,[Michel 93] characterized the singularities in the interaction matrix. A similar characterization of singularities is challenging task for image moments-based VS methods, including the photometric moments. In our view, this is also a difficult but important problem.

Combining geometric features with photometric moments A very interesting idea would be to define a hybrid approach combining photometric moments with geometric features. With this type of solution, we can profit from the excellent decoupling properties that can be obtained from the moments. At the same time, our hypothesis (based on experience and intuition gained thorough this work) is that this will allow also to perform visual servoing when there is less image overlap. Typically, when the displacement involved is very large, only a less overlap might be present between the current and the learnt images, especially during the beginning stages of the servo. In such scenarios, this kind of hybrid approach can be an effective solution.

Application to other modalities Television studio automation is an important application area in augmented reality [Nair 10]. In television studio automation, typically high-precision robots are used to control movements of studio equipment during a live broadcast. The studio environments are instrumented with mobile cameras equipped with infrared filters. In these cases, it would be very interesting to apply image moments to perform visual servoing of these cameras. In this case, we can take advantage of the decoupling properties of the image moments to handle huge displacements.

Extension to Omnidirectional Sensors Photometric image moments were applied to monocular camera sensors in this work. But omnidirectional cameras offer a better field of view of the scene. Pure photometric visual servoing has been successfully demonstrated with omnidirectional sensors [Caron 10]. Further, moment-based invariants were used to design decoupled control laws in the case of cameras which respect the unified projection model [Tahri 10b]. Therefore, it should be of interest to investigate using the photometric moments for the omnidirectional case.

Application to Navigation The methods developed in this thesis can be applied to the mobile robot navigation problem. Mobile robot navigation tasks can be classed into model-based and appearance-based methods [Courbon 09] [Šegvić 09] [DeSouza 02]. While model-based methods rely on a 3D model of the navigation space, appearance-based methods aim to work in the sensor space. Similar to visual servoing, this task consists in establishing a relationship between the information from the visual sensor and landmarks in the real-world observed by the camera. A mobile robot equipped with a monocular vision sensor is typically used. In the learning phase, the robot is guided along the required path while recording images at regular time intervals $\mathcal{I} = \{\mathbf{I}(t), \mathbf{I}(t+\Delta t), \mathbf{I}(t+2\Delta t), \ldots\}$ to build an image database. The task for the robot is then to reproduce the taught path. Typically in the navigation phase, a matching procedure matches points between the current image and the next key-image from the database. It would be worthwhile to see if image-moments based features can be used to augment or improve existing environment representations which use a set of tracked image points.

Bibliography

[Abdelkader 05] H.H. Abdelkader, Y. Mezouar, N. Andreff, P. Martinet. – 2 1/2 d visual servoing with central catadioptric cameras. – *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 3572–3577, Aug 2005.

[AbdulHafez 08] AH Abdul Hafez, Supreeth Achar, CV Jawahar. – Visual servoing based on gaussian mixture models. – *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pp. 3225–3230. IEEE, 2008.

[Agarwal 05] Anubhav Agarwal, CV Jawahar, PJ Narayanan. – A survey of planar homography estimation techniques. *Centre for Visual Information Technology, Tech. Rep. IIIT/TR/2005/12*, 2005.

[Allibert 10] Guillaume Allibert, Estelle Courtial, François Chaumette. – Predictive control for constrained image-based visual servoing. *Robotics, IEEE Transactions on*, 26(5):933–939, 2010.

[Aranda 13] M. Aranda, G. López-Nicolás, C. Sagüés. – Angle-based homing from a reference image set using the 1d trifocal tensor. *Autonomous Robots*, 34(1-2):73–91, 2013.

[Assa 13] A Assa, F. Janabi-Sharifi. – Closed-loop uncertainty modeling for visual servoing. – *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 3089–3094, May 2013.

[Baker 04] Simon Baker, Iain Matthews. – Lucas-kanade 20 years on: A unifying framework. *International journal of computer vision*, 56(3):221–255, 2004.

[Bakthavatchalam 13] M. Bakthavatchalam, F. Chaumette, E. Marchand. – Photometric moments: New promising candidates for visual servoing. – *IEEE Int. Conf. on Robotics and Automation, ICRA'13*, pp. 5521–5526, Karlsruhe, Germany, May 2013.

[Bakthavatchalam 14] M. Bakthavatchalam, O. Tahri, F. Chaumette. – Improving moments based visual servoing schemes with tunable visual features.

– *IEEE Int. Conf. on Robotics and Automation, ICRA'14*, Hong Kong, China, June 2014.

[Bakthavatchalam 15] M. Bakthavatchalam, F. Chaumette, O. Tahri. – An improved modelling scheme for photometric moments with inclusion of spatial weights for visual servoing with partial appearance/disappearance. – *IEEE Int. Conf. on Robotics and Automation, ICRA'15*, Seattle, WA, May 2015.

[Benhimane 06] Selim Benhimane, Ezio Malis. – Homography-based 2d visual servoing. – *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp. 2397–2402. IEEE, 2006.

[Benhimane 07] S. Benhimane, E. Malis. – Homography-based 2d visual tracking and servoing. *The International Journal of Robotics Research, IJRR'07*, 26(7):661–676, 2007.

[Caron 10] G. Caron, E. Marchand, E.M. Mouaddib. – Omnidirectional photometric visual servoing. – *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp. 6202–6207, Oct 2010.

[Cazy 14] N. Cazy, C. Dune, P.-B. Wieber, P. Robuffo Giordano, F. Chaumette. – Pose error correction for visual features prediction. – *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'14*, pp. 791–796, Chicago, IL, September 2014.

[Cervera 03] Enrique Cervera, Angel P Del Pobil, François Berry, Philippe Martinet. – Improving image-based visual servoing with three-dimensional features. *The International Journal of Robotics Research*, 22(10-11):821–839, 2003.

[Chaumette 90] François Chaumette. – *La relation vision-commande: théorie et application à des tâches robotiques*. – PhD. Thesis, Université Rennes I, France, 1990.

[Chaumette 98] F. Chaumette. – Potential problems of stability and convergence in image-based and position-based visual servoing. *The confluence of vision and control*, pp. 66–78, 1998.

[Chaumette 00] François Chaumette, Ezio Malis. – 2 1/2 d visual servoing: A possible solution to improve image-based and position-based visual servoings. – *ICRA*, pp. 630–635, 2000.

[Chaumette 04] F. Chaumette. – Image moments: a general and useful set of features for visual servoing. *IEEE Trans. on Robotics*, 20(4):713–723, Aug. 2004.

[Chaumette 06] F. Chaumette, S. Hutchinson. – Visual servo control, part i: Basic approaches. *IEEE Robot. Autom. Mag.*, 13(4):82–90, Dec. 2006.

[Chaumette 07] François Chaumette, Seth Hutchinson et al. – Visual servo control, part ii: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, 2007.

[Chesi 04] Graziano Chesi, Koichi Hashimoto, Domenico Prattichizzo, Antonio Vicino. – Keeping features in the field of view in eye-in-hand visual servoing: A switching approach. *Robotics, IEEE Transactions on*, 20(5):908–914, 2004.

[Chesi 09] G. Chesi. – Visual servoing path planning via homogeneous forms and lmi optimizations. *Robotics, IEEE Transactions on*, 25(2):281–291, April 2009.

[Cideciyan 95] Artur V Cideciyan. – Registration of ocular fundus images. *IEEE Engineering in medicine and biology*, 14(1):52–58, 1995.

[Collewet 08a] C. Collewet, E. Marchand, F. Chaumette. – Visual servoing set free from image processing. – *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, pp. 81–86, Pasadena, California, May 2008.

[Collewet 08b] Christophe Collewet, Eric Marchand. – *Photometric visual servoing*. – Rapport de recherche nRR-6631, INRIA, 2008.

[Collewet 11] C. Collewet, E. Marchand. – Photometric visual servoing. *IEEE Trans. on Robotics*, 27(4):828–834, Aug. 2011.

[Comaniciu 03] Dorin Comaniciu, Visvanathan Ramesh, Peter Meer. – Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564–577, 2003.

[Comport 03] Andrew Comport, Muriel Pressigout, Éric Marchand, François Chaumette. – A visual servoing control law that is robust to image outliers. – *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1, pp. 492–497. IEEE, 2003.

[Comport 04] Andrew I Comport, Éric Marchand, François Chaumette. – Robust model-based tracking for robot vision. – *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, pp. 692–697. IEEE, 2004.

[Comport 06] Andrew I Comport, Eric Marchand, François Chaumette. – Statistically robust 2-d visual servoing. *Robotics, IEEE Transactions on*, 22(2):415–420, 2006.

[Corke 93] Peter Corke. – Visual control of robot manipulators-a review. *Visual Servoing*, pp. 1–32, 1993.

[Corke 96] P.I Corke, M.C. Good. – Dynamic effects in visual closed-loop systems. *Robotics and Automation, IEEE Transactions on*, 12(5):671–683, Oct 1996.

[Corke 11] Peter Corke. – *Robotics, vision and control: fundamental algorithms in MATLAB*. – Springer, vol. 73, 2011.

[Courbon 09] Jonathan Courbon, Youcef Mezouar, Philippe Martinet. – Autonomous navigation of vehicles from a visual memory using a generic camera model. *Intelligent Transportation Systems, IEEE Transactions on*, 10(3):392–402, 2009.

[Dahmouche 12] Redwan Dahmouche, Nicolas Andreff, Youcef Mezouar, Omar Ait-Aider, Philippe Martinet. – Dynamic visual servoing from sequential regions of interest acquisition. *The International Journal of Robotics Research*, 31(4):520–537, 2012.

[Dame 10] Amaury Dame. – *A unified direct approach for visual servoing and visual tracking using mutual information*. – PhD. Thesis, Université Rennes I, France, 2010.

[Dame 11] Amaury Dame, Eric Marchand. – Mutual information-based visual servoing. *Robotics, IEEE Transactions on*, 27(5):958–969, 2011.

[Dani 13] Ashwin Dani, Ghazaleh Panahandeh, Soon-Jo Chung, Seth Hutchinson. – Image moments for higher-level feature based navigation. – *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 602–609. IEEE, 2013.

[Daniilidis 99] Konstantinos Daniilidis. – Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286–298, 1999.

[Deguchi 96] K. Deguchi, T. Noguchi. – Visual servoing using eigenspace method and dynamic calculation of interaction matrices. – *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, vol. 1, pp. 302–306 vol.1, Aug 1996.

[Deguchi 97] Koichiro Deguchi. – A direct interpretation of dynamic images and camera motion for visual servoing without image feature correspondance. *Journal of Robotics and Mechatronics*, 9(2):104–110, 1997.

[Deguchi 98] K. Deguchi. – Optimal motion control for image-based visual servoing by decoupling translation and rotation. – *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, vol. 2, pp. 705–711 vol.2, Oct 1998.

[Deguchi 00] Koichiro Deguchi. – A direct interpretation of dynamic images with camera and object motions for vision guided robot control. *International Journal of Computer Vision*, 37(1):7–20, 2000.

[Delabarre 12] B. Delabarre, E. Marchand. – Visual servoing using the sum of conditional variance. – *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 1689–1694, Oct 2012.

[dePlinval 11] H. de Plinval, Pascal Morin, P. Mouyon, T. Hamel. – Visual servoing for underactuated vtol uavs: A linear, homography-based approach. – *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 3004–3010, May 2011.

[DeSouza 02] Guilherme N DeSouza, Avinash C Kak. – Vision for mobile robot navigation: A survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(2):237–267, 2002.

[Doignon 07] Christophe Doignon. – *An Introduction to Model-Based Pose Estimation and 3-D Tracking Techniques*. – INTECH Open, 2007.

[Durand 13] Petiteville A. Durand, S. Durola, V. Cadenat, M. Courdesses. – Management of visual signal loss during image based visual servoing. – *Control Conference (ECC), 2013 European*, pp. 2305–2310, July 2013.

[Espiau 92] B. Espiau, F. Chaumette, Patrick Rives. – A new approach to visual servoing in robotics. *Robotics and Automation, IEEE Transactions on*, 8(3):313–326, Jun 1992.

[Espiau 94] Bernard Espiau. – Effect of camera calibration errors on visual servoing in robotics. *Experimental Robotics III*, éd. par Tsuneo Yoshikawa, Fumio Miyazaki, pp. 182–192. – Springer Berlin Heidelberg, 1994.

[Fang 02] Y. Fang, D.M. Dawson, W.E. Dixon, M.S. de Queiroz. – Homography-based visual servoing of wheeled mobile robots. – *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 3, pp. 2866–2871 vol.3, Dec 2002.

[Faugeras 01] O. Faugeras, Q.T. Luong, T. Papadopoulo. – *The Geometry of Multiple Images: The Laws that Govern the Formation of Multiple Images of a Scene and Some of Their Applications*. – MIT Press, 2001.

[Feddema 89] John T Feddema, Owen R Mitchell. – Vision-guided servoing with feature-based trajectory generation [for robots]. *Robotics and Automation, IEEE Transactions on*, 5(5):691–700, 1989.

[Feddema 91] J.T. Feddema, C. S G Lee, O.R. Mitchell. – Weighted selection of image features for resolved rate visual feedback control. *Robotics and Automation, IEEE Transactions on*, 7(1):31–47, Feb 1991.

[Flusser 09] J. Flusser, T. Suk, B. Zitov, Inc Ebrary. – *Moments and moment invariants in pattern recognition*. – Wiley Online Library, 2009.

[Folio 08] David Folio, Viviane Cadenat. – Treating image loss using the vision/motion link: a generic framework. *Computer Vision*, pp. 4–26, 2008.

[Gans 03] Nicholas R Gans, Seth A Hutchinson, Peter I Corke. – Performance tests for visual servo control systems, with application to partitioned approaches to visual servo control. *The International Journal of Robotics Research*, 22(10-11):955–981, 2003.

[Gans 07] N.R. Gans, S.A. Hutchinson. – Stable visual servoing through hybrid switched-system control. *Robotics, IEEE Transactions on*, 23(3):530–540, June 2007.

[Gans 11] N. R. Gans, G. Hu, J. Shen, Y. Zhang, W. E. Dixon. – Adaptive visual servo control to simultaneously stabilize image and pose error. *Mechatronics*, 2011.

[GarcíaAracil 05] Nicolás García-Aracil, Ezio Malis, Rafael Aracil-Santonja, Carlos Pérez-Vidal. – Continuous visual servoing despite the changes of visibility in image features. *Robotics, IEEE Transactions on*, 21(6):1214–1220, 2005.

[Ghorbel 06] Faouzi Ghorbel, Stéphane Derrode, Rim Mezhoud, M. Tarak Bannour, Sami Dhahbi. – Image reconstruction from a complete set of similarity invariants extracted from complex moments. *Pattern Recognition Letters*, 27(12):1361–1369, 2006.

[Gonçalves 10] T. Gonçalves, J. Azinheira, Patrick Rives. – Homography-based visual servoing of an aircraft for automatic approach and landing. – *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 9–14, May 2010.

[Gridseth 13] M Gridseth, C.P Quintero, R.Tatsambon Fomena, O.Ramirez, M.Jagersand. – Bringing visual servoing into real world applications. – *Human Robot Collaboration Workshop, Robotics Science and Systems RSS'13*, Berlin, Germany, June 2013.

[Hafez 07] AH Abdul Hafez, Anil Kumar Nelakanti, CV Jawahar. – Path planning approach to visual servoing with feature visibility constraints: a convex optimization based solution. – *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 1981–1986. IEEE, 2007.

[Hartley 00] R.I Hartley, N.Y. Dano. – Reconstruction from six-point sequences. – *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, pp. 480–486 vol.2, 2000.

[Hartley 04] R. I. Hartley, A. Zisserman. – *Multiple View Geometry in Computer Vision*. – Cambridge University Press, ISBN: 0521540518, second édition, 2004.

[Hilbert 93] D. Hilbert, B. Sturmfels. – *Theory of Algebraic Invariants*. – Cambridge University Press, 1993.

[Horaud 95] Radu Horaud, Fadi Dornaika. – Hand-eye calibration. *The international journal of robotics research*, 14(3):195–210, 1995.

[Horn 81] Berthold K. P. Horn, Brian G. Schunck. – Determining optical flow. *ARTIFICAL INTELLIGENCE*, 17:185–203, 1981.

[Hu 62] M.K. Hu. – Visual pattern recognition by moment invariants. *IEEE Trans. on Information Theory*, 8(2):179–187, 1962.

[Hutchinson 96] S. Hutchinson, G. Hager, P. Corke. – A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, octobre 1996.

[Iwatsuki 05] M. Iwatsuki, N. Okiyama. – A new formulation of visual servoing based on cylindrical coordinate system. *IEEE Trans. on Robotics*, 21(2):266–273, 2005.

[Jabbari 12] H. Jabbari, G. Oriolo, H. Bolandi. – Dynamic ibvs control of an underactuated uav. – *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pp. 1158–1163, Dec 2012.

[Jabbari 14] H. Jabbari, Giuseppe Oriolo, Hossein Bolandi. – An adaptive scheme for image-based visual servoing of an underactuated uav. *International Journal of Robotics and Automation*, 29(1), 2014.

[JanabiSharifi 11] F. Janabi-Sharifi, Lingfeng Deng, William J. Wilson. – Comparison of basic visual servoing methods. *Mechatronics, IEEE/ASME Transactions on*, 16(5):967–983, Oct 2011.

[JehanBesson 03] Stéphanie Jehan-Besson, Michel Barlaud, Gilles Aubert. – Dream2s: Deformable regions driven by an eulerian accurate minimization method for image and video segmentation. *International Journal of Computer Vision*, 53(1):45–70, 2003.

[Kallem 07] V. Kallem, M. Dewan, J.P. Swensen, G.D. Hager, N.J. Cowan. – Kernel-based visual servoing. – *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'07*, pp. 1975–1980, San Diego, California, Nov. 2007. IEEE.

[Kallem 09] V. Kallem, The Johns Hopkins University. – *Vision-based Control on Lie Groups with Application to Needle Steering*. – The Johns Hopkins University, 2009.

[Kazemi 10] Moslem Kazemi, Kamal Gupta, Mehran Mehrandezh. – Path-planning for visual servoing: A review and issues. *Visual Servoing via Advanced Numerical Methods*, éd. par Graziano Chesi, Koichi Hashimoto, pp. 189–207. – Springer London, 2010.

[Khalil 02] Wisama Khalil, Etienne Dombre. – *Modeling, Identification and Control of Robots*. – Taylor & Francis, Inc., 3rd édition, Bristol, PA, USA, 2002.

[Komuro 07] T. Komuro, M. Ishikawa. – A moment-based 3d object tracking algorithm for high-speed vision. – *Robotics and Automation, 2007 IEEE International Conference on*, pp. 58–63, April 2007.

[Kyrki 06] Ville Kyrki, Danica Kragic, Henrik I Christensen. – Measurement errors in visual servoing. *Robotics and Autonomous Systems*, 54(10):815–827, 2006.

[Lapresté 04] Jean-Thierry Lapresté, Youcef Mezouar. – A hessian approach to visual servoing. – *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, pp. 998–1003. IEEE, 2004.

[Lazar 08] Corneliu Lazar, Adrian Burlacu. – Predictive control strategy for image based visual servoing of robot manipulators. – *Proc. of 9th International Conference on Automation and Information, Bucharest*, pp. 91–97, 2008.

[Li 13] Baoquan Li, Yongchun Fang, Xuebo Zhang. – Uncalibrated visual servoing of nonholonomic mobile robots. – *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 584–589, Nov 2013.

[Liberzon 99] D. Liberzon, A.S. Morse. – Basic problems in stability and design of switched systems. *Control Systems, IEEE*, 19(5):59–70, 1999.

[Mahony 12] R. Mahony, V. Kumar, P. Corke. – Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *Robotics Automation Magazine, IEEE*, 19(3):20–32, Sept 2012.

[Malis 98] E. Malis. – *Contributions à la modélisation et à la commande en asservissement visuel*. – PhD. Thesis, Université de Rennes 1, Mention traitement du signal et télécommunications, novembre 1998.

[Malis 00] E. Malis, F. Chaumette, S. Boudet. – 2-1/2d visual servoing. *Robotics and Automation, IEEE Transactions on*, 15(2):238–250, 2000.

[Malis 02] Ezio Malis, François Chaumette. – Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods. *Robotics and Automation, IEEE Transactions on*, 18(2):176–186, 2002.

[Malis 04] E. Malis. – Improving vision-based control using efficient second-order minimization techniques. – *IEEE Int. Conf. on Robotics and Automation, 2004. Proceedings. ICRA'04*, vol. 2, pp. 1843–1848. IEEE, 2004.

[Mansard 07] N. Mansard, F. Chaumette. – Task sequencing for high-level sensor-based control. *Robotics, IEEE Transactions on*, 23(1):60–72, Feb 2007.

[Mansard 09] N. Mansard, A Remazeilles, F. Chaumette. – Continuity of varying-feature-set control laws. *Automatic Control, IEEE Transactions on*, 54(11):2493–2505, Nov 2009.

[Marchand 02] Éric Marchand, François Chaumette. – Virtual visual servoing: a framework for real-time augmented reality. – *Computer Graphics Forum*, vol. 21, pp. 289–297. Wiley Online Library, 2002.

[Marchand 04] E. Marchand, A Comport, F. Chaumette. – Improvements in robust 2d visual servoing. – *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, pp. 745–750 Vol.1, April 2004.

[Marchand 05] E. Marchand, F. Spindler, F. Chaumette. – ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robot. Autom. Mag.*, 12(4):40–52, Dec. 2005.

[Marey 08] M. Marey, F. Chaumette. – Analysis of classical and new visual servoing control laws. – *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, pp. 3244–3249, Pasadena, California, May 2008.

[Martinet 99] Philippe Martinet. – Comparison of visual servoing techniques: Experimental results. – *Proceedings of the European Control Conference, ECC'99, paper 1059-4, Karlsruhe, Germany*, 1999.

[Mebarki 10] R. Mebarki, Alexandre Krupa, F. Chaumette. – 2-d ultrasound probe complete guidance by visual servoing using image moments. *Robotics, IEEE Transactions on*, 26(2):296–306, April 2010.

[Mezouar 02] Y. Mezouar, F. Chaumette. – Path planning for robust image-based control. *Robotics and Automation, IEEE Transactions on*, 18(4):534–549, Aug 2002.

[Michel 93] Henri Michel, Patrick Rives. – *Singularities in the determination of the situation of a robot effector from the perspective view of 3 points*. – Research Report nRR-1850, INRIA, 1993.

[Mukundan 98] Ramakrishnan Mukundan, KR Ramakrishnan. – *Moment functions in image analysis: theory and applications*. – World Scientific, vol. 100, 1998.

[Murray 94] Richard M Murray, Zexiang Li, S Shankar Sastry, S Shankara Sastry. – *A mathematical introduction to robotic manipulation*. – CRC press, 1994.

[Nair 10] Suraj Nair, Thorsten Röder, Giorgio Panin, Alois Knoll. – Visual servoing of presenters in augmented virtual reality tv studios. – *Proceedings of the IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems*. IEEE, 2010.

[Nayar 96] S.K. Nayar, S.A. Nene, H. Murase. – Subspace methods for robot vision. *Robotics and Automation, IEEE Transactions on*, 12(5):750–758, Oct 1996.

[Nelson 95] Bradley J Nelson, Pradeep K Khosla. – Strategies for increasing the tracking region of an eye-in-hand system by singularity and joint limit avoidance. *The International journal of robotics research*, 14(3):255–269, 1995.

[Nematollahi 12] Eissa Nematollahi, Aleksandar Vakanski, Farrokh Janabi-Sharifi. – A second-order conic optimization-based method for visual servoing. *Mechatronics*, 22(4):444–467, 2012.

[Palmer 99] Stephen E Palmer. – *Vision science: Photons to phenomenology*. – MIT Press, 1999.

[Paragios 06] Nikos Paragios, Yunmei Chen, Olivier D Faugeras. – *Handbook of mathematical models in computer vision*. – Springer, 2006.

[Paulk 13] Angelique Paulk, S Sean Millard, Bruno van Swinderen. – Vision in drosophila: seeing the world through a model's eyes. *Annual review of entomology*, 58:313–332, 2013.

[Petit 13] A. Petit, E. Marchand, K. Kanani. – A robust model-based tracker for space applications: combining edge and color information. – *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'2013*, pp. 3719–3724, Tokyo, Japan, November 2013.

[Phong 75] Bui Tuong Phong. – Illumination for computer generated pictures. *Commun. ACM*, 18(6):311–317, juin 1975.

[Prokop 92] Richard J Prokop, Anthony P Reeves. – A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graphical Models and Image Processing*, 54(5):438 – 460, 1992.

[Reddi 81] S. S. Reddi. – Radial and angular moment invariants for image identification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-3(2):240–242, March 1981.

[Reiss 91] T.H. Reiss. – The revised fundamental theorem of moment invariants. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(8):830–834, Aug 1991.

[Ruszczyński 06] Andrzej P Ruszczyński. – *Nonlinear optimization*. – Princeton university press, vol. 13, 2006.

[Samson 91a] Claude Samson, K Ait-Abderrahim. – Feedback control of a nonholonomic wheeled cart in cartesian space. – *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pp. 1136–1141. IEEE, 1991.

[Samson 91b] Claude Samson, Bernard Espiau, Michel Le Borgne. – *Robot Control: The Task Function Approach*. – Oxford University Press, 1991.

[Sanderson 80] AC Sanderson, LE Weiss. – Image-based visual servo control using relational graph error signals. *Proc. ieee*, 1074, 1980.

[Šegvić 09] Siniša Šegvić, Anthony Remazeilles, Albert Diosi, François Chaumette. – A mapping and localization framework for scalable appearance-based navigation. *Computer Vision and Image Understanding*, 113(2):172–187, 2009.

[Shademan 10] A Shademan, M. Jagersand. – Three-view uncalibrated visual servoing. – *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp. 6234–6239, Oct 2010.

[Shannon 48] Claude Shannon. – A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.

[Sharma 97] Rajeev Sharma, Seth Hutchinson. – Motion perceptibility and its application to active vision-based servo control. *Robotics and Automation, IEEE Transactions on*, 13(4):607–617, 1997.

[Silveira 06] G. Silveira, E. Malis, Patrick Rives. – Visual servoing over unknown, unstructured, large-scale scenes. – *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp. 4142–4147, May 2006.

[Silveira 07a] G. Silveira, E. Malis. – Direct visual servoing with respect to rigid objects. – *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 1963–1968, Oct 2007.

[Silveira 07b] G. Silveira, E. Malis. – Real-time visual tracking under arbitrary illumination changes. – *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1–6, June 2007.

[Silveira 08] G. Silveira, E. Malis, Patrick Rives. – An efficient direct approach to visual slam. *Robotics, IEEE Transactions on*, 24(5):969–979, Oct 2008.

[Silveira 09] G. Silveira, E. Malis. – Visual servoing from robust direct color image registration. – *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 5450–5455, Oct 2009.

[Silveira 10] Geraldo Silveira, Ezio Malis. – Unified direct visual tracking of rigid and deformable surfaces under generic illumination changes in grayscale and color images. *International journal of computer vision*, 89(1):84–105, 2010.

[Silveira 12] G. Silveira, E. Malis. – Direct visual servoing: Vision-based estimation and control using only nonmetric information. *IEEE Transactions on Robotics*, 28(4):974–980, 2012.

[Soatto 11] Stefano Soatto. – Steps towards a theory of visual information: Active perception, signal-to-symbol conversion and the interplay between sensing and control. *arXiv preprint arXiv:1110.2053*, 2011.

[Sonka 14] Milan Sonka, Vaclav Hlavac, Roger Boyle. – *Image processing, analysis, and machine vision*. – Cengage Learning, 2014.

[Swensen 10] JohnP. Swensen, Vinutha Kallem, NoahJ. Cowan. – Empirical characterization of convergence properties for kernel-based visual servoing. *Visual Servoing via Advanced Numerical Methods*, éd. par Graziano Chesi, Koichi Hashimoto, pp. 23–38. – Springer London, 2010.

[Tahri 04] O. Tahri. – *Utilisation des moments en asservissement visuel et en calcul de pose*. – PhD. Thesis, Université Rennes I, France, 2004.

[Tahri 05a] O. Tahri, F. Chaumette. – Point-based and region-based image moments for visual servoing of planar objects. *IEEE Trans. on Robotics*, 21(6):1116–1127, Dec. 2005.

[Tahri 05b] Omar Tahri, François Chaumette. – Complex objects pose estimation based on image moment invariants. – *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 436–441. IEEE, 2005.

[Tahri 08] Omar Tahri, Youcef Mezouar. – On the efficient second order minimization and image-based visual servoing. – *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pp. 3213–3218. IEEE, 2008.

[Tahri 10a] Omar Tahri, Youcef Mezouar. – On visual servoing based on efficient second order minimization. *Robotics and Autonomous Systems*, 58(5):712–719, 2010.

[Tahri 10b] Omar Tahri, Youcef Mezouar, François Chaumette, Peter Corke. – Decoupled image-based visual servoing for cameras obeying the unified projection model. *Robotics, IEEE Transactions on*, 26(4):684–697, 2010.

[Tamtsia 13a] A. Tamtsia, Yeremou. – *Nouvelles contributions à l'application des moments en asservissement visuel*. – PhD. Thesis, Université Blaise Pascal - Clermont-Ferrand II, France, 2013.

[Tamtsia 13b] A. Tamtsia, Yeremou, O. Tahri, Y. Mezouar, H. Djalo, T. Tonye. – New results in image moments-based visual servoing. – *IEEE International Conference on Robotics and Automation 2013*, Karlsruhe, Germany, May 2013.

[Tatsambon 13] Fomena.R. Tatsambon, C. Perez Quintero, M. Gridseth, M. Jagersand. – Towards practical visual servoing in robotics. – *Computer and Robot Vision (CRV), 2013 International Conference on*, pp. 303–310, 2013.

[Teh 88] C. H. Teh, R. T. Chin. – On image analysis by the method of moments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 10(4):496–513, 1988.

[Teuliere 14] C. Teuliere, E. Marchand. – A dense and direct approach to visual servoing using depth maps. *Robotics, IEEE Transactions on*, 30(5):1242–1249, Oct 2014.

[Trivedi 07] Mohan M Trivedi, Tarak Gandhi, Joel McCall. – Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety. *Intelligent Transportation Systems, IEEE Transactions on*, 8(1):108–120, 2007.

[TUM 12] TUM. – Hand-eye calibration. – `http://campar.in.tum.de/Chair/HandEyeCalibration`, 2012. Technical University of Munich, Centre for Augmented Reality.

[VanGool 95] Luc Van Gool, Theo Moons, Eric Pauwels, André Oosterlinck. – Vision and lie's approach to invariance. *Image and vision computing*, 13(4):259–277, 1995.

[Wang 08] J. Wang, H. Cho. – Micropeg and hole alignment using image moments based visual servoing method. *IEEE Trans. on Industrial Electronics*, 55(3):1286–1294, 2008.

[Weiss 87] Lee E Weiss, Arthur C Sanderson, Charles P Neuman. – Dynamic sensor-based control of robots with visual feedback. *Robotics and Automation, IEEE Journal of*, 3(5):404–417, 1987.

[Wilson 96] William J. Wilson, C.C. Williams Hulls, G.S. Bell. – Relative end-effector control using cartesian position based visual servoing. *Robotics and Automation, IEEE Transactions on*, 12(5):684–696, Oct 1996.

List of Publications

International Conferences

- M. Bakthavatchalam, F. Chaumette., O. Tahri. An Improved Modelling Scheme for Photometric Moments with Inclusion of Spatial Weights for Visual Servoing with Partial Appearance/Disappearance. *IEEE/RSJ Int. Conf. on Robotics and Automation, ICRA 2015*, Seattle, USA, May 2015.

- M. Bakthavatchalam, O. Tahri., F. Chaumette. Improving Moments-based Visual Servoing with Tunable Visual Features. In *IEEE/RSJ Int. Conf. on Robotics and Automation, ICRA 2014*, HongKong, China, June 2014.

- M. Bakthavatchalam, F. Chaumette., E. Marchand. Photometric moments: New promising candidates for visual servoing. In *IEEE/RSJ Int. Conf. on Robotics and Automation, ICRA 2013*, Karlsruhe, Germany, May 2013.

Résumé

Cette thèse s'intéresse à l'asservissement visuel, une technique de commande à retour d'information visuelle permettant de contrôler le mouvement de systèmes équipées de caméras tels que des robots. Pour l'asservissement visuel, il est essentiel de synthétiser les informations obtenues via la caméra et ainsi établir la relation entre l'évolution de ces informations et le déplacement de la caméra dans l'espace. Celles-ci se basent généralement sur l'extraction et le suivi de primitives géométriques comme des points ou des lignes droites dans l'image. Il a été montré que le suivi visuel et les méthodes de traitement d'images restent encore un frein à l'expansion des techniques d'asservissement visuel. C'est pourquoi la distribution de l'intensité lumineuse de l'image a également été utilisée comme caractéristique visuelle. Finalement, les caractéristiques visuelles basée sur les moments de l'image ont permis de définir des lois de commande découplées. Cependant ces lois de commande sont conditionnées par l'obtention d'une région parfaitement segmentée ou d'un ensemble discret de points dans la scène.

Ce travail propose donc une stratégie de capture de l'intensité lumineuse de façon indirecte, par le biais des moments calculés sur toute l'image. Ces caractéristiques globales sont dénommées moments photométriques. Les développements théoriques établis dans cette thèse tendent à définir une modélisation analytique de la matrice d'interaction relative aux moments photométriques. Ces derniers permettent de réaliser une tâche d'asservissement visuel dans des scènes complexes sans suivi visuel ni appariement. Un problème pratique rencontré par cette méthode dense d'asservissement visuel est l'apparition et la disparition de portions de l'image durant la réalisation de la tâche. Ce type de problème peut perturber la commande, voire dans le pire des cas conduire à l'échec de la réalisation de la tâche. Afin de résoudre ce problème, une modélisation incluant des poids spatiaux est proposée. Ainsi, la pondération spatiale, disposant d'une structure spécifique, est introduite de telle sorte qu'un modèle analytique de la matrice d'interaction peut être obtenue comme une simple fonction de la nouvelle formulation des moments photométriques.

Une partie de ce travail apporte également une contribution au problème de la commande simultanée des mouvements de rotation autour des axes du plan image. Cette approche définit les caractéristiques visuelles de façon à ce que l'asservissement soit optimal en fonction de critères spécifiques. Quelques critères de sélection basées sur la matrice d'interaction ont été proposés. Ce travail ouvre donc sur d'intéressantes perspectives pour la sélection d'informations visuelles pour l'asservissement visuel basé sur les moments de l'image.

Mots-clés : Asservissement visuel, Commande référencée vision, Moments photométriques, Asservissement visuel basé intensité lumineuse, Robotique, Vision par ordinateur.

Summary

This thesis is concerned with *visual servoing*, a feedback control technique for controlling camera-equipped actuated systems like robots. For visual servoing, it is essential to synthesize visual information from the camera image in the form of visual features and establish the relationship between their variations and the spatial motion of the camera. The earliest visual features are dependent on the extraction and visual tracking of geometric primitives like points and straight lines in the image. It was shown that visual tracking and image processing procedures are a bottleneck to the expansion of visual servoing methods. That is why the image intensity distribution has also been used directly as a visual feature. Finally, visual features based on image moments allowed to design decoupled control laws but they are restricted by the availability of a well-segmented regions or a discrete set of points in the scene.

This work proposes the strategy of capturing the image intensities not directly, but in the form of moments computed on the whole image plane. These global features have been termed photometric moments. Theoretical developments are made to derive the analytical model for the interaction matrix of the photometric moments. These moments enable to perform visual servoing on complex scenes without visual tracking or image matching procedures. A practical issue encountered in such dense VS methods is the appearance and disappearance of portions of the scene during the visual servoing. Such unmodelled effects can disturb the control and in the worst case, result in complete failure to convergence. To handle this important practical problem, an improved modelling scheme for the moments that allows for inclusion of spatial weights is proposed. Then, spatial weighting functions with a specific structure are exploited such that an analytical model for the interaction matrix can be obtained as simple functions of the newly formulated moments.

A part of this work provides an additional contribution towards the problem of simultaneous control of rotational motions around the image axes. The approach is based on connecting the design of the visual feature such that the visual servoing is optimal with respect to specific criteria. Few selection criteria based on the interaction matrix was proposed. This contribution opens interesting possibilities and finds immediate applications in the selection of visual features in image moments-based VS.

Keywords : Visual Servoing, Vision-based control, Image moments, Photometric moments, Intensity-based visual servoing, Robotics, Computer Vision