الجمهورية الجزائرية الديمقراطية الشعبية

**République Algérienne Démocratique et Populaire**

وزارة التعليم العالي والبحث العلمي

**Ministère de l'enseignement supérieur et de la recherche scientifique**

**Université Mohamed Khider - Biskra-**

**Faculté des sciences et technologie**

**Département de Génie Électrique**

**Référence: ……..**

**جامعة محمد خيضر – بسكرة-**

**كلية العلوم والتكنولوجيا**

**قسم الهندسة الكهربائية**

**المرجع: .....**

Thèse présentée en vue de l'obtention
Du diplôme de :

**Doctorat LMD en Génie Électrique**

Spécialité : Automatique et informatique industrielle
Option :

Intitulé :

# Analyse Multi échelle des Images pour le Problème d'Asservissement Visuel

Présentée par :

**ROUAG Fadi Elislam**

Soutenue publiquement le :

Devant le jury composé de :

| | | | |
|---|---|---|---|
| **Président:** | **Pr. Zine-Eddine BAARIR** | **Prof** | **Université de Biskra** |
| **Superviseur:** | **Pr. Nadjiba TERKI** | **Prof** | **Université de Biskra** |
| **Examinateur:** | **Pr. Salim SBAA** | **Prof** | **Université de Biskra** |
| **Examinateur:** | **Dr. Djalloul ACHOUR** | **MCA** | **Université de Chlef** |

**2024-2025**

الجمهورية الجزائرية الديمقراطية الشعبية

**People's Democratic Republic of Algeria**

وزارة التعليم العالي والبحث العلمي

**Ministry of Higher Education and Scientific Research**

**Mohamed Khider University - Biskra-**

**Faculty of Science and Technology**

**Department of Electrical Engineering**

**Ref: ……..**

**جامعة محمد خيضر – بسكرة-**

**كلية العلوم والتكنولوجيا**

**قسم الهندسة الكهربائية**

**المرجع: .....**

Thesis presented with a view to obtaining
From the diploma of:

**LMD PhD in Electrical Engineering**

Specialty: Automation and industrial computing
Option:

Entitled:

# Multiscale Image Analysis for the Visual Servoing Problem

Presented by:

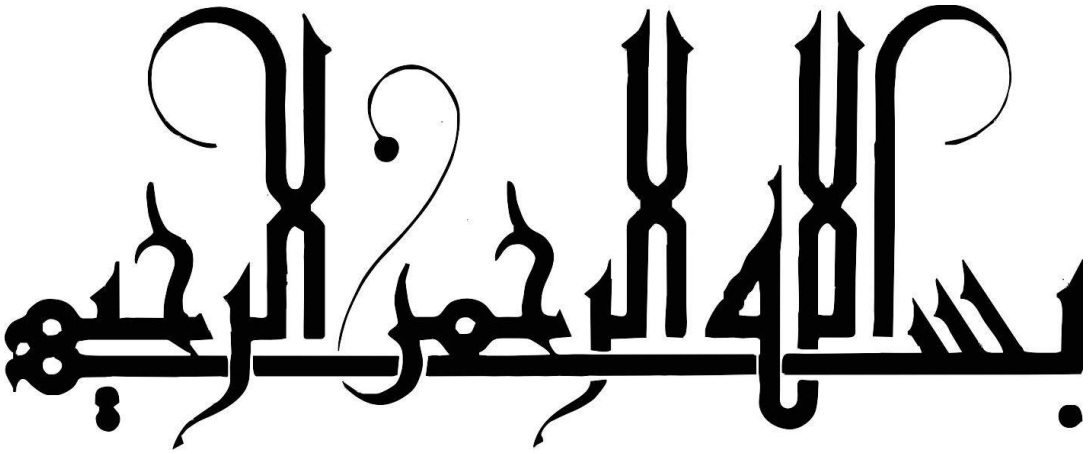
**ROUAG Fadi Elislam**

Publicly supported on:

In front of the jury composed of:

| | | | |
|---|---|---|---|
| **President:** | **Pr. Zine-Eddine BAARIR** | **Prof** | **University of Biskra** |
| **Supervisor:** | **Pr. Nadjiba TERKI** | **Prof** | **University of Biskra** |
| **Examiner:** | **Pr. Salim SBAA** | **Prof** | **University of Biskra** |
| **Examiner:** | **Dr. Djalloul ACHOUR** | **MCA** | **University of Chlef** |

**2024 / 2025**



# DEDICATION

To my parents, your unwavering belief in my abilities has served as a beacon of hope for me as I continue to learn. You gave me an insatiable curiosity and a thirst for knowledge. I will always be thankful for your sacrifices, both visible and invisible, which have made it possible for me to pursue this doctorate.

My admirable supervisor, **Pr. Nadjiba TERKI**, has always been an inspiration to me because of your unwavering faith in my abilities. I have the strength to overcome obstacles and pursue excellence because of your unwavering support and unrelenting enthusiasm. This journey has been even more rewarding because of you in my life.

Last but not least, I would like to express my gratitude to everyone who has helped me along the way, whether directly or indirectly, with this academic endeavor. Your positive feedback, deeds of kindness, and faith in my abilities have been a huge help.

## ACKNOWLEDGEMENTS

I want to start by sincerely thanking Allah, the Most Merciful and Compassionate, for giving me the courage, direction, and insight I needed to complete my doctoral studies.

My profound gratitude goes out to my supervisor, Pr. Nadjiba TERKI, for their unwavering support, priceless advice, and boundless tolerance. Their knowledge, commitment, and support have greatly influenced my research and academic development.

For their time, knowledge, and critical assessment of my thesis, I would like to thank the distinguished jury members:

**-1 Pr. Zine-Eddine BAARIR** professor at the University of Biskra.

**-2 Pr. Salim SBAA** professor at the University of Biskra.

**-3 Dr. Djalloul ACHOUR** professor at the University of Chlef.

Last but not least, I would like to sincerely thank my parents for always being a source of love, support, and encouragement. My accomplishments have been fueled by their sacrifices and unwavering faith in my abilities.

## Abstract

Visual servoing enables the control of robotic systems directly from visual feedback and has become a fundamental technique in vision-based robotics. Among existing approaches, Photometric Visual Servoing (PVS) relies on raw image intensities as visual features, avoiding explicit feature extraction and matching. While PVS offers high accuracy and a compact formulation, its performance is strongly affected by illumination variations, limiting its robustness in realistic operating conditions.

This thesis proposes an enhanced photometric visual servoing framework based on multiresolution image analysis, referred to as Lifting Wavelet Transform-based Photometric Visual Servoing (LWT-PVS). The method integrates the lifting wavelet transform into the PVS formulation to exploit spatial-frequency image representations while preserving the direct nature of photometric control. By selecting appropriate wavelet sub-bands, the proposed approach reduces sensitivity to illumination changes and photometric disturbances, while maintaining stable convergence properties.

A unified mathematical formulation of LWT-PVS is developed, including the definition of photometric visual features, error functions, interaction matrices, and discrete-time control laws. The complete control architecture is implemented using the ViSP (Visual Servoing Platform) simulator, ensuring consistency with established visual servoing tools.

Extensive simulation studies are conducted for multiple scenarios involving different initial camera poses and both nominal and varying illumination conditions. The performance of PVS and LWT-PVS is evaluated in terms of error convergence, camera velocity behavior, and robustness to lighting variations. The results show that LWT-PVS significantly improves robustness to illumination changes compared to classical PVS, while preserving convergence accuracy and stability.

The contributions of this work demonstrate the relevance of multiresolution representations for photometric visual servoing and provide a robust control framework suitable for real-world robotic applications operating under challenging illumination conditions.

**Keywords:** Lifting Scheme; Visual servoing; Wavelet Transform; Interaction matrix; Photometric visual servoing ; Illumination variations; ViSP simulator.

**الملخص**

تُمكّن تقنية التحكم البصري من التحكم في الأنظمة الروبوتية مباشرةً من خلال التغذية الراجعة البصرية، وقد أصبحت تقنية أساسية في مجال الروبوتات القائمة على الرؤية. من بين الأساليب الحالية، يعتمد التحكم البصري الضوئي (PVS) على شدة الإضاءة الخام للصورة كخصائص بصرية، متجنبًا بذلك استخراج الخصائص ومطابقتها بشكل صريح. على الرغم من أن PVS يتميز بدقة عالية وصيغة مختصرة، إلا أن أداءه يتأثر بشدة بتغيرات الإضاءة، مما يحد من متانته في ظروف التشغيل الواقعية.

تقترح هذه الأطروحة إطار عمل مُحسّنًا للتحكم البصري الضوئي قائمًا على تحليل الصور متعددة الدقة، ويُشار إليه باسم التحكم البصري الضوئي القائم على تحويل المويجات الرافعة (LWT-PVS). يدمج هذا الأسلوب تحويل المويجات الرافعة في صيغة PVS لاستغلال تمثيلات الصورة المكانية الترددية مع الحفاظ على الطبيعة المباشرة للتحكم الضوئي. من خلال اختيار نطاقات فرعية مناسبة للمويجات، يقلل الأسلوب المقترح من الحساسية لتغيرات الإضاءة والاضطرابات الضوئية، مع الحفاظ على خصائص تقارب مستقرة.

من خلال اختيار نطاقات فرعية مناسبة للمويجات، يقلل الأسلوب المقترح من الحساسية لتغيرات الإضاءة والاضطرابات الضوئية، مع الحفاظ على خصائص تقارب مستقرة. تم تطوير صياغة رياضية موحدة لنظام التحكم البصري الضوئي LWT-PVS، تشمل تعريف الخصائص البصرية الضوئية، ودوال الخطأ، ومصفوفات التفاعل، وقوانين التحكم في الزمن المتقطع. تم تطبيق بنية التحكم الكاملة باستخدام محاكي ViSP (منصة التحكم البصري)، مما يضمن التوافق مع أدوات التحكم البصري المعروفة.

أُجريت دراسات محاكاة مكثفة لسيناريوهات متعددة تتضمن أوضاعًا ابتدائية مختلفة للكاميرا وظروف إضاءة اسمية ومتغيرة. تم تقييم أداء نظامي PVS وLWT-PVS من حيث تقارب الخطأ، وسلوك سرعة الكاميرا، ومقاومة تغيرات الإضاءة. تُظهر النتائج أن نظام LWT-PVS يُحسّن بشكل ملحوظ مقاومة تغيرات الإضاءة مقارنةً بنظام PVS التقليدي، مع الحفاظ على دقة التقارب والاستقرار.

تُبرز مساهمات هذا العمل أهمية تمثيلات متعددة الدقة للتحكم البصري الضوئي، وتوفر إطار تحكم قويًا مناسبًا لتطبيقات الروبوتات في العالم الحقيقي التي تعمل في ظروف إضاءة صعبة.

**الكلمات المفتاحية:** مخطط الرفع؛ التوجيه البصري؛ تحويل الموجات؛ مصفوفة التفاعل؛ التوجيه البصري الضوئي؛ تغيرات الإضاءة؛ محاكي ViSP.

## LIST OF FIGURES

# List OF ACRONYMS

| | |
|---|---|
| VS | : Visual Servoing |
| IBVS | : Image-based Visual Servoing |
| PBVS | : Pose-based Visual Servoing |
| PVS | : Photometric Visual Servoing |
| LWT-PVS | : Lifting Wavelet Transform-based Photometric Visual Servoing |
| 6-dof | : Six degrees of freedom |
| WT | : Wavelet Transform |
| LWT | : Lifting Wavelet Transform |
| CWT | : Continuous Wavelet Transform |
| DWT | : Discrete Wavelet Transform |
| GMM | : Gaussian mixture model |
| ViSP | Visual Servoing Platform |

# TABLE OF CONTENTS

## CHAPTER2. LIFTING WAVELET TRANSFORM



## CHAPTER3. LIFTING WAVELET TRANSFORM-BASED PHOTOMETRIC VISUAL SERVOING



## CHAPTER4. SIMULATION RESULTS



## GENERAL CONCLUSION AND PERSPECTIVES

# GENERAL INTRODUCTION

**Contents**

## 1 General Introduction

Visual Servoing is a multidisciplinary topic that lies computer vision, robotics and automatic control. It is a feasible technique for robotic closed-loop control that uses information extracted from images, obtained by one or more sensors [1][2][3]. This field has advanced considerably in recent years with applications such as medical robotics [4], autonomous drones [5], etc. In general, visual servoing is designed to minimize the difference between a vector $s(t)$ of current features and a vector $s^*$ of the desired features in order to accomplish a positioning or tracking task. Points, lines, and geometric moments were among the first features of the geometric type taken into consideration in visual servoing [6][7][8]. For these geometric methods, the method's capability to identify, extract, and track visual features during the robot control process is essential.

Historically, key point coordinates or line parameters have been extensively utilized for the purpose of computing various visual features that are derived from image data. This complex process demands a highly accurate extraction and a meticulous tracking process that cannot be overlooked. Geometric shapes, including rectangles, circles, and ellipses, must be efficiently identified, monitored, and consistently followed across time in a spatiotemporal manner. This continuous tracking is essential to maintain the integrity and positional relevance of these shapes within different frames of reference. In the highly specialized field of visual servoing, such precision in tracking becomes absolutely critical not only for the functionality of real-time control systems but also for enhancing the overall performance of automated tasks. Here, the accurate detection combined with the reliable tracking of features plays a crucial role in effectively guiding and maneuvering robotic systems through their designated tasks and operational environments. The precision required for these activities ensures that robotic systems can adapt to changing conditions and execute their functions with the highest level of efficiency and reliability. This capacity is fundamentally important for complex applications where even slight deviations can lead to significant errors or failures in operation [9][10][11].

## 2 Context

Recently, the rapid development of innovative methodologies, such as Photometric Visual Servoing (PVS), has significantly evolved, utilizing pixel intensities as the primary visual features for enhancing precision in robotic applications and control systems [12]. Furthermore, there are many more complex features, including mutual information [13], the total of the image's conditional variances [14], mixtures of Gaussians [15], or photometric moments [16]. The convergence domain of these direct visual servoing methods is smaller, but they are often

considered to be more precise than approaches that use geometric visual features. These advanced techniques significantly enhance the precision of control and increase the resilience of the system by employing global descriptors. Among these descriptors, photometric moments play a crucial role as they are utilized to compute the interaction matrix. This innovative approach enables the creation of visual feedback, which is then effectively leveraged to drive and refine robotic movements in a more sophisticated manner [17][18][19]. Nonetheless, variations in lighting conditions and dynamic scene elements, which include factors such as occlusions or unexpected changes in the object's appearance, can significantly undermine the overall effectiveness of these techniques. These variations pose substantial challenges to their robustness in various practical applications, making it difficult to achieve consistent results across different situations and environments [20].

To effectively tackle this complex issue, a substantial number of rigorous studies have thoroughly investigated the diverse application of frequency-domain image transformations in various contexts. Among these notable methods, two of the most significant are the Discrete Cosine Transform (DCT) and the Fast Fourier Transform (FFT) [21][22]. These advanced techniques play a crucial role in significantly enhancing visual servoing systems, which rely heavily on sophisticated and complex image processing for achieving highly accurate control, along with seamless automation in a broad range of applications. The ongoing development and continuous refinement of such innovative image transformation methods remain an essential and vibrant area of research, contributing to impressive advancements in technology and thereby improving overall system performance in various fields and industries [19]. However, these methods often fail to adequately account for the inherently temporal nature of signals, particularly when dealing with non-stationary image data, such as real-time video streams that are continuously changing. In contrast, wavelet transforms have gained considerable prominence due to their unique dual ability to capture both frequency and temporal information effectively. This remarkable feature makes them particularly effective and well-suited in managing the complexities of non-stationary signals encountered in various applications. Wavelet-based techniques outperform traditional frequency-domain methods, providing better robustness against lighting fluctuations and offering improved spatial resolution [23][24][25][26].

The development of a direct visual servoing method using multi-scale representations that describe the image's spatial and frequency information is the main objective of the work presented in this thesis. In order to create multi-scale representations of an image, high-pass and low-pass filters are applied to the image successively, followed by subsampling. Each filtering level then contains continuously smaller details over the decompositions. These

methods not only enable the creation of multiple levels of detail, but they also enable the creation of a sparse representation of the image, meaning that most of the coefficients have values near zero, which reflects the essential information of the image. Multi-scale representations gained importance as a result of harmonic analysis. The representation of signals as a series of waves is the primary interest of this field, which has its roots in Fourier analysis. An optimal wave system is constructed to describe the structure of an image in order to develop an interesting multiscale representation. One of the main fields of research in modern applied mathematics is the development of such systems.

Duflot et al. [27] introduced a noteworthy enhancement to wavelet-based techniques specifically aimed at managing illumination changes effectively. This was achieved by employing both shearlet and wavelet coefficients to model the intricate features of input images in a more robust manner. Despite these promising advancements in the field, the computational cost that is associated with calculating the necessary interaction matrices, which utilize such sophisticated coefficients, continues to pose a significant hurdle in achieving efficient real-time applications [28][29]. As such, while wavelet-based methods demonstrate considerable promise in various applications, their overall applicability is frequently constrained by notably high computational demands, which can significantly hinder their potential for real-time deployment in practical scenarios [30].

## 3 Motivations and Objectives

The lifting scheme presents an innovative and highly effective solution to the various computational challenges that are often inherent in wavelet transforms [31]. By utilizing a non-Fourier approach, it provides an exceptionally efficient method for constructing biorthogonal wavelets, which considerably reduces the computational cost when compared to traditional methods [32]. The lifting scheme is particularly beneficial in the specific context of visual servoing, where real-time processing capabilities are essential, and achieving computational efficiency is paramount for successful applications. Through the lifting approach, second-generation wavelets are constructed directly in the spatial domain, offering a more flexible, adaptable, and computationally efficient alternative to first-generation wavelets that operate in the frequency domain [33]. This adaptiveness highlights the lifting scheme's potential to address the varied needs of modern computational tasks, enhancing performance while maintaining speed and reducing resource utilization.

In the realm of signal and image processing, lifting schemes facilitate a broad array of applications, encompassing tasks such as denoising, compression, feature extraction, and classification [34]. Furthermore, the ability of lifting schemes to effectively segment signals

into various frequency bands while ensuring high computational efficiency renders them exceptionally well-suited for real-time applications in dynamic environments. These noteworthy advantages have led to the extensive adoption of lifting schemes across diverse fields, particularly in audio and image processing, where maintaining high computational efficiency is of utmost importance [35]. Such versatility and effectiveness also invite further exploration into innovative uses of lifting schemes in emerging technologies.

This study introduces a novel and innovative Photometric Visual Servoing (PVS) approach that effectively utilizes the lifting wavelet transform (designated as LWT-PVS). The primary aim of this research is to comprehensively address the significant shortcomings and limitations found in conventional PVS methods, particularly in complex scenarios that involve considerable and varying lighting conditions. By deriving and extracting visual feedback from both the approximation coefficients and the detail coefficients obtained from the lifting wavelet transform, this advanced approach allows for the formulation of more precise and accurate control laws. The key advantage and significant strength of this methodology lie in its remarkable ability to represent specific and essential spatial features, including edges, contours, and textures, directly from the processed image data. The application of the lifting wavelet transform directly in the spatial domain facilitates the real-time identification and extraction of features, which is a critical capability required in applications that demand precise localization. This includes areas such as robotic navigation, autonomous vehicles, and sophisticated image-based control systems, where accurate and rapid responses to changing environments are essential for successful operation and functionality [36][37][38][39][40].

In addition to significantly improving computational efficiency, the lifting scheme also provides a reversible transformation, which means that the inverse wavelet transform can be computed by systematically reversing the steps of the forward transform [41]. This particular feature is especially useful in systems where maintaining data integrity and achieving lossless recovery are crucial and of utmost importance. Furthermore, the versatility of the lifting scheme extends remarkably to non-uniform data sampling, making it a highly powerful tool for various scenarios where traditional Fourier-based methods might not be applicable or ideal. The adaptability of this approach enables better handling of diverse data structures while effectively preserving essential characteristics during the transformation process.

## 4 Contributions and Thesis Outline

The contributions of this research are as follows:

1. **New Approach to Visual Servoing:** In contrast to the method proposed by Mendoza et al. [42], which relied on raw luminance values for visual feedback, the LWT-PVS approach

utilizes approximation and detail coefficients derived from the lifting wavelet transformation. This represents a shift towards utilizing spatial-domain features that are more robust to illumination changes, significantly enhancing control precision.

2. **Reduction in Dimensionality and Robustness to Illumination Variations:** By using approximation coefficients, the approach provides a simplified, low-dimensional representation of the image. The detail coefficients encapsulate high-frequency content that remains invariant under changes in illumination, offering better robustness compared to previous PVS methods [43].

3. **Mathematical Development of Interaction Model:** The interaction matrix for the LWT-PVS method is derived by integrating the optical flow constraint equation (OFCE) with the lifting wavelet transform. This interaction model serves as the backbone of the proposed control law, incorporating both approximation and detail features in a unified framework [44].

4. **Method Validation:** Through simulation-based experiments, the proposed LWT-PVS method is shown to outperform conventional PVS techniques in scenarios with varying illumination, demonstrating its practical applicability in real-time visual servoing systems.

The structure of this thesis is as follows:

Chapter 1 - **Visual Servoing**: The chapter presents a detailed introduction to background in visual servoing, a brief survey of the currently existing visual servoing methods. The traditional geometric primitives-based IBVS, PBVS and Hybrid VS methods are first presented. This is followed by presentation of current state of the art in visual servoing methods that use intensity.

Chapter 2 - **Lifting Wavelet Transform Overview:** In this chapter, an overview on the classical wavelet transform with its drawbacks is presented. Then, the second generation WT, the lifting wavelet transform with its advantages is introduced.

Chapter 3 - **The Proposed Features**: is dedicated to LWT-PVS, the central theme of this thesis. The formulation of LWT-PVS is introduced and detailed mathematical developments necessary for obtaining the interaction matrix are presented.

Chapter 4 - **Simulation Results:** in this chapter, proposed visual features are compared to the pure luminance feature under illumination variations, and the results obtained were presented. Finally, conclusions are drawn based on these results.

**General conclusion and perspectives:** Lastly, a conclusion on our work and perspectives for future work are presented.

CHAPTER 1

# VISUAL SERVOING

## Contents

## 1.1 Introduction

In recent years, the rapid advancements in computing capabilities have made visual servoing an integral area of research and industry. Visual servoing refers to a feedback control system that relies on a computer vision unit, where the information provided by a vision sensor is used to guide the movements of a dynamic system. This system can either be physical, such as in robotics, or virtual, such as in the animation of artificial entities or augmented reality.

Among the various visual servoing techniques, those belonging to direct visual servoing schemes are often considered to be more accurate and robust than methods based on geometric feature extraction. These techniques have found significant application in ultrasound-guided medical robots, as traditional methods requiring visual feature extraction struggle with ultrasound images. They are also widely used in fields such as medical robotics (especially in surgery), surgical micro-robotics, and assistive robots for people with disabilities.

Regardless of the sensor configuration ranging from onboard cameras mounted on a robot's end effector to multiple remote cameras, it is essential to select the most relevant visual information. Subsequently, a control law is developed to manage the appropriate degrees of freedom to ensure that the visual data reaches a desired value, which ultimately defines the task's success. These techniques can be applied to a wide variety of tasks, such as positioning a system relative to its environment or tracking moving objects, by controlling one or more degrees of freedom. This allows for compensation of inaccuracies in either the sensor or the system being controlled.

Visual servoing, when utilizing a vision sensor that provides 2D information, offers a wealth of potential visual data. For example, it is possible to use 2D data such as the coordinates of key points in an image, or even 3D data derived from a localization module that processes 2D measurements. The challenge in visual servoing lies in selecting the most relevant information from this abundant visual data, ensuring that the system performs effectively. This requires the development of real-time image processing algorithms capable of extracting and tracking measurements in video-frequency image sequences.

In essence, visual servoing involves controlling actuated systems through visual feedback, with a broad range of closed-loop approaches. The goal of a visual servoing task is to control the actuator in such a way that the error between the current visual features, $s\ t$ , and the desired features, $s^*$, approaches zero, i.e., the difference $s\ t\ - s^*$ becomes negligible.

## 1.2 The Basic Components of Visual Servoing

Vision-based control techniques aim to control a system's behavior using visual information, generally captured through cameras or other imaging devices. In such techniques, the primary

objective is to minimize an error signal, denoted as which represents the discrepancy between the desired visual features and the actual visual features at a given time. This error, $e\ t\ $, is mathematically defined as:

$$e\ t\ = s\ m\ t\ ;a\ - s^{*} \tag{1.1}$$

The formulation discussed is broad and adaptable, enabling its application across various vision-based control methods. The image measurements, denoted as $m\ t\ $, represent a set of features extracted from the camera's current image at time *t*. These measurements can include quantities such as the image coordinates of points of interest, the centroid of an object, or other geometric features like edges and corners. Based on these measurements, a vector of $k$ visual features, $s\ m\ t\ ;a\ $, is computed. This computation may also utilize additional system-specific parameters, $a$, which represent prior knowledge such as the intrinsic parameters of the camera (e.g., focal length, pixel size, or principal point) or 3D models of objects within the scene. The desired state of the system is captured by the vector $s^{*}$, which contains the target values for these visual features. For example, $s^{*}$ may define the desired pose of an object in the image frame or the target pose of a robotic manipulator relative to an object. The error signal, $e\ t\ = s\ m\ t\ ;a\ - s^{*}$, quantifies the difference between the current and desired states. The primary goal of vision-based control is to minimize this error over time, ensuring the system accurately converges to the target state. Through this formulation, a wide range of tasks, such as object tracking, navigation, or robotic manipulation, can be addressed using either Image-Based Visual Servoing (IBVS) or Pose- Based Visual Servoing (PBVS). This approach provides an effective framework for leveraging visual information to achieve autonomous control.

For this discussion, we focus on a specific case where the desired pose ($s^{*}$) is fixed, and the target remains immobile. In this scenario, $s^{*}$ is constant, and any changes in the visual features $s$ are solely due to the motion of the camera. Furthermore, controlling the camera's motion using six degrees of freedom (6-dof) is considered, corresponding to a camera mounted on the end effector of a robotic arm with six independent movements: three translational (along *x*, *y*, and *z* axes) and three rotational (roll, pitch, and yaw). The goal in this setup is to adjust the robotic arm's motion so that the current visual features s align with the desired visual features $s^{*}$. This is achieved by minimizing the error $e\ t\ = s\ m\ t\ ;a\ - s^{*}$, where $e\ t\ $ represents the discrepancy between the current and target states.

A key distinction in visual servoing techniques lies in the design of the visual features $s$. Classical approaches to visual servoing can be categorized into two main types. The first

approach is IBVS, where $\boldsymbol{s}$ is defined as a set of features that are directly extracted from the image, such as pixel coordinates, geometric shapes (e.g., points, lines), or image moments. The second approach is PBVS, where $s$ represents the pose (position and orientation) of the target relative to the camera. In this case, the pose is computed indirectly by extracting image measurements and using them in conjunction with additional system knowledge, such as the intrinsic parameters of the camera or a 3D model of the target. In earlier literature, PBVS was referred to as pose-based visual servoing rather than pose-based, but both terminologies are now widely accepted and used interchangeably.

More advanced methods extending visual servoing beyond classical techniques will be explored. These methods build on the foundations of IBVS and PBVS but address challenges such as robustness to occlusions, dynamic targets, and improving the flexibility and accuracy of control in complex, real-world scenarios. By exploring these different approaches, a comprehensive understanding of visual servoing techniques and their applications in a variety of contexts is intended to be provided.

Once the visual features $s$ are defined, the design of a control scheme becomes relatively straightforward. Among the various options, designing a velocity controller is one of the simplest and most commonly used approaches in visual servoing. To achieve this, it is essential to establish the mathematical relationship between the camera's spatial velocity and the temporal variation of the visual features $s$. This relationship defines how changes in the camera's motion affect the visual features over time.

Let the camera's spatial velocity be represented by $\nu_c = (v_c; \omega_c)$, where $\nu_c$ is the instantaneous linear velocity of the camera's origin (i.e., the translational velocity), and $\omega_c$ is the instantaneous angular velocity of the camera's frame (i.e., the rotational velocity). The time derivative of the visual features, represented by $s$, is directly related to the spatial velocity of the camera through the interaction matrix (also referred to as the image Jacobian or the feature Jacobian) denoted by $L_s$. This relationship is expressed as:

$$s = L_s \nu_c \tag{1.2}$$

The matrix $L_s \in \mathbb{R}^{k\times 6}$ is referred to as the interaction matrix associated with $s$ [45]. Through the use of Equations (1.1) and (1.2), a direct connection can be established between the camera's velocity and the rate of change of the error over time:

$$e = L_e \nu_c \tag{1.3}$$

In this context, $L_e = L_s$. If $v_c$ is used as the input to the robot controller and an exponential, independent reduction of the error is desired (i. e., $e = -\lambda e$), the controller can be derived using Equation (1.3) as follows:

$$v_c = -\lambda L_e^+ e \tag{1.4}$$

where $L_e^+ \in \mathbb{R}^{6\times k}$ represents the Moore-Penrose pseudoinverse of $L_e$, which is, $L_e^+ = (L_e L_e^\top)^{-1} L_e^\top$ with $k \geq 6$ and $L_e$ is of full rank 6. It is possible to invert $L_e$ when $k = 6$ if $\det L_e \neq 0$, and the control gives $\boldsymbol{v}_c = -\lambda L_e^{-1}\boldsymbol{e}$. If $k \leq 6$ and $L_e$ is of full rank $k$, $L_e^+$ is given by $L_e^+ = L_e^\top (L_e L_e^\top)^{-1}$. If $L_e$ is not full rank, the singular value decomposition of $L_e$ can be used to obtain the numerical value of $L_e^+$. In all situations, control scheme (1.4) permits minimal $\left\|e - \lambda L_e \widehat{L_e^+} e\right\|$ and $\|v_c\|$. The desired behavior $\boldsymbol{e} = -\lambda \boldsymbol{e}$ is achieved only if $L_e L_e^+ = \mathbf{I}_k$, where $\mathbf{I}_k$ is the $k \times k$ identity matrix, that is, only if $L_e$ is of full rank $k$, $k \leq 6$.

In real-world visual servo systems, it is not feasible to exactly determine $L_e$ or its pseudo-inverse $L_e^+$. As a result, one must rely on approximations or estimates of these matrices. Henceforth, the symbol $\widehat{L_e^+}$ is used to denote both the pseudo-inverse of the approximated interaction matrix and the approximation of the pseudo-inverse itself. Using this representation, the control law can then be expressed as follows:

$$v_c = -\lambda \widehat{L_e^+} e = -\lambda \widehat{L_s^+}(s - s^*) \tag{1.5}$$

Assuming the robot controller can achieve perfect $\boldsymbol{v_c}$, and closing the loop by inserting (1.5) into (1.3):

$$e = -\lambda L_e \widehat{L_e^+} e \tag{1.6}$$

This equation illustrates the closed-loop system's real behavior, which is different from the desired one ($\boldsymbol{e} = -\lambda \boldsymbol{e}$) once $L_e \widehat{L_e^+} \neq \mathbf{I}_k$. Also, it is the basis for the Lyapunov theory-based stability analysis of the system.

The majority of visual servo controllers use the fundamental concept that have been presented above. Schematic of a robotic visual servoing system is illustrated in Fig.1.1.

The next stage involves elaborating on the intricate details of the process. One of the primary questions is: how should the task-specific variable, $s$, be appropriately chosen to ensure optimal

system performance? Following this, it is essential to determine the precise form and structure of the interaction matrix, $L_s$, which plays a critical role in visual servoing control. Another pressing concern is the accurate estimation of the pseudo-inverse of the interaction matrix, denoted as $\widehat{L_e^+}$. The answers to these questions are central to the development of efficient and robust visual servoing algorithms and will be systematically addressed in the subsequent sections of this chapter.

Additionally, the chapter will provide a comprehensive overview of the two fundamental strategies in the field of visual servoing: IBVS and PBVS. These two classical approaches were originally conceptualized more than two decades ago [46] and have since served as the backbone for numerous advancements in this domain. Building upon their foundational principles, modern research has introduced various extensions and enhancements to these methodologies, overcoming historical limitations and improving system reliability, robustness, and adaptability. In the later sections, these contemporary solutions, which leverage advances in computational techniques and sensor technologies, will be presented in detail.

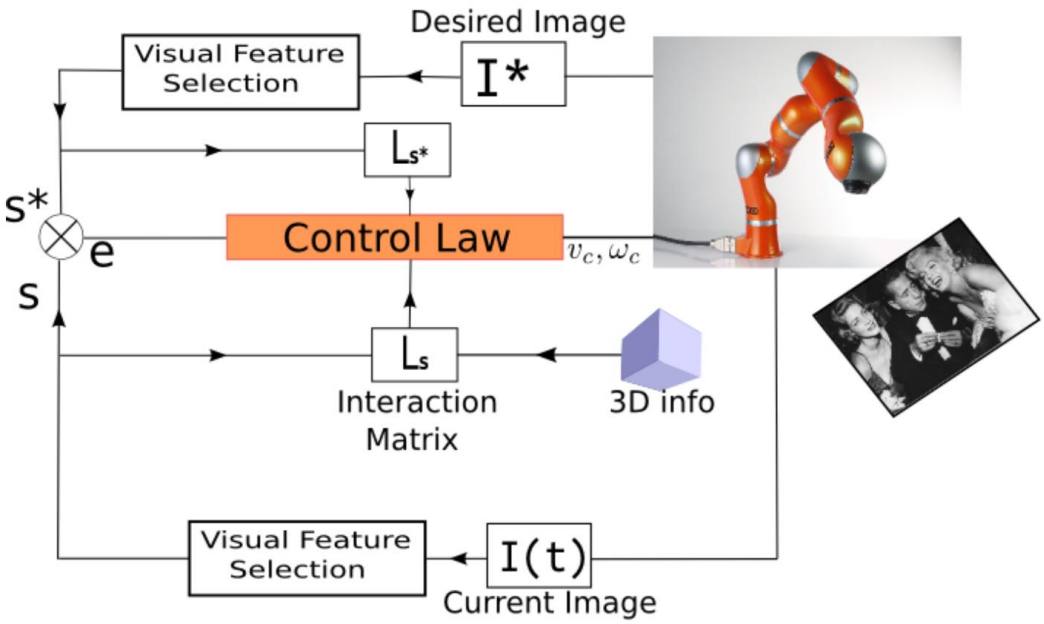



**Figure 1.1:** Schematic of a robotic visual servoing system with eye-in-hand configuration [47]

### 1.3 Camera-robot configurations

In visual servoing applications, two primary configurations are commonly employed to integrate cameras with robotic systems for perception and control. The first configuration, referred to as the "eye-in-hand" setup, involves mounting one or more cameras directly onto

the robot's end effector. This placement allows the cameras to move in conjunction with the robot, enabling close observation of the workspace or tools attached to the end effector. To interpret and utilize the camera data effectively in this configuration, a fixed and well-calibrated transformation is defined between the coordinate frame of the camera and that of the robot's end effector. This transformation facilitates the conversion of motion and visual data between the camera frame, the end effector frame, and other reference frames associated with the robot system.

The second configuration, known as the "eye-to-hand" setup, involves positioning one or more static cameras in the robot's workspace. These cameras are used to observe the robot's end effector, the target objects, or both from an external viewpoint. Unlike the eye-in-hand configuration, in eye-to-hand systems, it is necessary to compute the relationship (transformation) between the robot's coordinate frame and the static camera frame at each iteration. This dynamic transformation is crucial for maintaining spatial awareness and ensuring the accuracy of visual servoing operations. The eye-to-hand configuration is particularly useful when a broader perspective of the workspace is needed or when multiple targets and objects must be tracked simultaneously.

The eye-in-hand configuration is often the preferred choice in tasks where it is important to keep the target or object of interest, as well as the tool attached to the end effector, within the camera's field of view throughout the operation. An example of this is object-grasping tasks, where the camera must monitor both the object being manipulated and the gripper mounted on the end effector for precise alignment and control [48]. However, there are also instances where hybrid configurations, combining elements of both eye-in-hand and eye-to-hand setups, are employed to exploit the advantages of each approach [49]. These hybrid systems can offer supplementary observations or redundancy for enhanced accuracy and flexibility in complex manipulation tasks.

In the work described in this thesis, the focus is on a visual servoing system utilizing the eye-in-hand configuration. This setup provides the advantage of a camera perspective tightly coupled to the motions of the robot's end effector, making it well-suited for tasks requiring precise monitoring and manipulation of objects within a localized environment.

### 1.3.1 Eye-in-hand configuration

The integration of the camera with the robot's end-effector adopts an eye-in-hand configuration, as depicted in Fig. 1.2. Within this framework, visual servoing control strategies are employed to define the appropriate velocity vectors, denoted as $v$ or $q$, depending on

whether the desired motion is to be regulated in the camera space or the robot's joint space. These velocity commands are transmitted to the robot controller, enabling precise execution of the corresponding movements.

To enable accurate motion control, it is essential to determine several key parameters critical to the system's operation. These include the camera's intrinsic parameters, the robot's Jacobian matrix ( ${}^{e}J_{q}$), and the transformation matrix ( ${}^{c}M_{b}$) that describes the spatial relationship between the camera's coordinate frame ($\mathcal{F}_{c}$) and the robot's end-effector frame ($\mathcal{F}_{e}$). The process for identifying these parameters involves performing camera calibration, robot calibration, and calibration of the relative pose between the camera and the end-effector. These calibration steps ensure that the geometrical relationships and system dynamics are well-characterized and aligned for control purposes, following standard methodologies as outlined, for instance, in Tsai's method [50].

One important aspect of visual servoing systems is their inherent robustness to calibration inaccuracies. Even in the presence of small errors in the calibration parameters, the control algorithms used in visual servoing are designed to adapt and perform reliably, which adds a layer of resilience to the overall system. This feature makes visual servoing particularly advantageous for robotic applications in dynamic or imperfect environments where precise calibration may not always be feasible or may degrade over time.

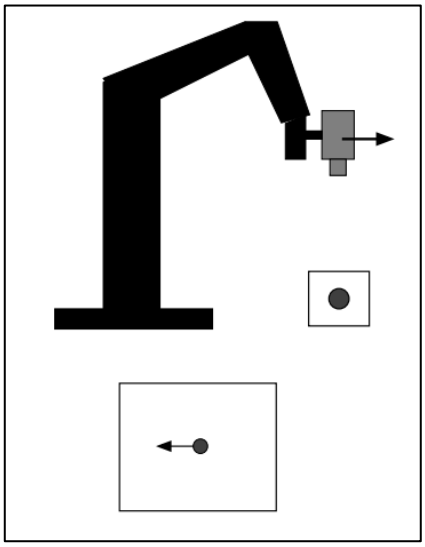


**Figure 1.2:** Eye-in-hand camera-robot configuration

### 1.3.2 Eye-to-hand configuration

In the eye-to-hand configuration, the camera is mounted in a fixed position within the robot's workspace, as illustrated in Fig. 1.3. Unlike the eye-in-hand configuration, where the camera is

attached to the end effector and thus captures image changes resulting from both the movements of the target and the end effector, the eye-to-hand configuration limits image variation to the motion of the target alone. This separation simplifies certain aspects of system kinematics and control by decoupling the camera's viewpoint from the robotic manipulator's movements.

A critical component of this configuration is the transformation matrix $^{c}\boldsymbol{M_b}$ which describes the static relationship between the base coordinate system of the robot $\mathcal{F}_b$ and the camera's coordinate frame $\mathcal{F}_c$. Since the camera is stationary, $^{c}\boldsymbol{M_b}$ is computed once during system setup and remains fixed throughout operation. This matrix forms the foundation for calculating the relative positions and orientations between various coordinate frames within the system, particularly the relationship between the camera frame and the robot's end effector frame. By systematically applying frame transformations, the kinematic relationships between the robot, camera, and target can be established and utilized for control or perception tasks.

The eye-to-hand configuration offers several advantages, such as simplifying the modeling of image-based feedback control systems and eliminating the need to recalibrate for camera motion. As a result, it is well-suited for robotics applications where precise monitoring of a target is required without directly coupling the camera to the manipulator's dynamics. For further details about camera and robot configurations, readers can refer to the foundational works on visual servoing and robotic vision, such as those by Hutchinson et al. [1], Kragic and Christensen [51], and Chaumette and Hutchinson [52]. Their contributions provide in-depth discussions on the principles, advantages, and use cases for different camera-robot configurations.

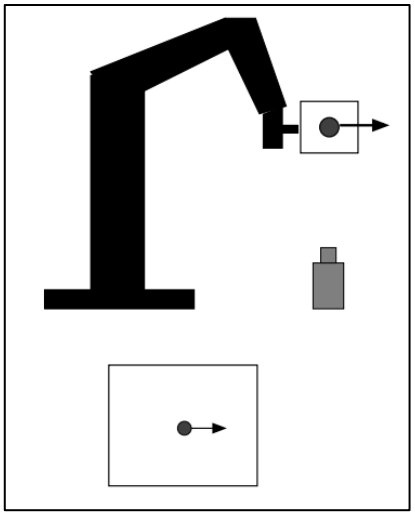


**Figure 1.3:** Eye-to-hand camera-robot configuration

### 1.4 Geometric Feature-based methods

Visual servoing (VS) is a control technique that utilizes visual information from a camera to guide a robot's motion. The core objective of this approach is to enable the robot to interact with its environment by processing the visual data obtained from images of objects within the scene. In feature-based visual servoing methods, a set of image-based features or primitives, denoted as $m_i \in \mathcal{M}_{2D}$, is extracted from the image, where $\mathcal{M}_{2D}$ represents the collection of all measurable quantities in the image space. These image features are derived using a variety of image-processing techniques, such as object detection, feature matching, visual tracking, and segmentation, all of which are critical for isolating and interpreting the relevant visual cues within the image.

In most cases, these features correspond to the projections of certain geometric entities ($P_i$) that exist in the three-dimensional scene, such as points, edges, straight lines, or other geometric structures. These 3D entities belong to the spatial domain ($P_i \in \mathcal{S}_{3D}$) and are projected onto the 2D image plane through the camera's perspective projection model. The extraction of such features creates a link between the information in the image and the physical objects in the environment, allowing real-time control of the robotic system.

Depending on how the extracted image-based measurements are used to define visual features, feature-based visual servoing can be categorized into three main types: IBVS, PBVS and hybrid or 2.5D approaches. IBVS operates directly in the image space, utilizing image features to compute control signals without explicit reliance on the scene's 3D geometry. In contrast, PBVS relies on estimating the pose (position and orientation) of the object in 3D space relative to the robot, typically requiring additional computations to reconstruct 3D information. Hybrid approaches, often referred to as 2.5D visual servoing (introduced by [53]), blend elements of both IBVS and PBVS, leveraging the advantages of both methodologies while mitigating their respective limitations. Specifically, hybrid techniques aim to combine the responsiveness and simplicity of 2D visual servoing with the enriched spatial awareness of 3D approaches.

Each of these schemes offers distinct advantages and trade-offs depending on the application and system requirements. For instance, IBVS is generally robust to calibration inaccuracies but can face challenges related to image-space singularities. PBVS, on the other hand, provides better spatial interpretation but is more sensitive to errors in camera calibration and pose estimation. Hybrid approaches attempt to balance these strengths and weaknesses by providing a flexible framework that preserves the benefits of both the 2D and 3D domains. Details regarding the advantages, limitations, and practical considerations of these methods can be

found in the works of Chaumette (1998) [54] and Chaumette and Hutchinson (2006) [1], which serve as foundational references in the field.

### 1.4.1 Image-Based Visual Servoing (IBVS)

Image-based control schemes use measurements of 2D primitives observed in the image-plane to define the task function [55][47].

$$e\ t\ = s\ m(I\ t\ )\ - s^{*} \tag{1.7}$$

The most commonly employed visual features in visual servoing applications are the coordinates of n specific points located within the image plane, represented as $s \in \mathbb{R}^{2n}$ [55][47]. These points, which are extracted from the camera's captured image, serve as essential components for guiding the visual servoing process. Each feature point is expressed as part of a coordinate set, denoted as $s = (\mathbf{x}_1; \mathbf{x}_2; \mathbf{x}_3; ....; \mathbf{x}_n)$, where $\mathbf{x}_i$ corresponds to the 2D coordinates of the i-th point in the image frame. These points are visual primitives detected by the camera, as illustrated in Figs. 1.3(a) and 1.3(b), and their locations form the basis for defining the robot's control objectives.

To execute a visual servoing task, a reference or desired set of image features must first be specified, which corresponds to the visual features that the system should achieve when the robot reaches its goal pose. This desired feature set is denoted as $s^*$, where $s^*$ represents the image-space coordinates of the points when the object and camera are in their target relative positions. The control objective in visual servoing is to ensure the current set of features s converges to the desired feature set $s^*$, thereby guiding the robot to accomplish the intended task.

The visual servoing control law is designed to minimize the error between the current feature set s and the desired feature set $s^*$. This error, commonly expressed as $e = s - s^*$, represents the deviation of the system's current state from the goal. The control strategy generates velocity commands for the robot such that e asymptotically approaches zero, hence driving the robot to the desired pose. Depending on the type of visual servoing method employed—image-based visual servoing (IBVS), pose-based visual servoing (PBVS), or hybrid methods—different formulations of the control law are employed to optimize the trajectory and ensure system stability. For instance, in IBVS, this control law operates directly in the 2D image space, while in PBVS, it is mapped into the 3D space of the robot's pose.

The use of point-based features is widely popular due to their computational simplicity, ease of detection, and universality across various applications. However, selecting appropriate visual

features and establishing a robust correspondence between $s$ and $s^*$ are critical for achieving stable and accurate visual servoing performance. This process often relies on advanced computer vision techniques to track, match, and refine the detected points in noisy or dynamic environments. Future sections (or subsequent chapters) will delve into how these principles are applied and adapted in specific visual servoing tasks to address challenging scenarios, such as occlusions, calibration errors, and system constraints.

$$v_c = -\lambda \widehat{L_s^+} \begin{bmatrix} x_1 - x_1^* \\ x_2 - x_2^* \\ \cdot \\ \vdots \\ x_n - x_n^* \end{bmatrix} \tag{1.8}$$

The error $e$ in Equation (1.7) must be reduced to zero in an exponential manner. Achieving this requires constructing the interaction matrix corresponding to an image point, which forms the cornerstone for all subsequent computations involving visual features. The interaction matrix plays a critical role in linking the dynamics of the image features to the dynamics of the robotic system, laying the groundwork for effective control strategies.

Among the many approaches proposed for robotic vision-based control, IBVS has emerged as a dominant technique due to its practical benefits. This method continues to attract significant interest in the field of robotics and computer vision research. The appeal of IBVS is rooted in its simplicity of implementation, its capacity to handle uncertainties in camera calibration, and its robustness against image noise. These characteristics make IBVS a reliable and adaptable solution for real-world robotic applications, even in unstructured or dynamically changing environments. Its resilience to errors and inaccuracies has solidified its reputation as a dependable approach for achieving precise image-driven robotic motion.

Furthermore, IBVS's ability to directly operate in the 2D image space without requiring a precise 3D reconstruction of the environment is a key advantage. This eliminates the dependency on complex calibration workflows while improving overall system efficiency.

In general, before IBVS can be used practically, a few issues need to be resolved. Among the existing problems are the following:

1. IBVS technique, while widely used, is not without its challenges. One notable issue is the potential occurrence of image singularity, which can lead to unstable control behavior and numerical difficulties [54][56][57].

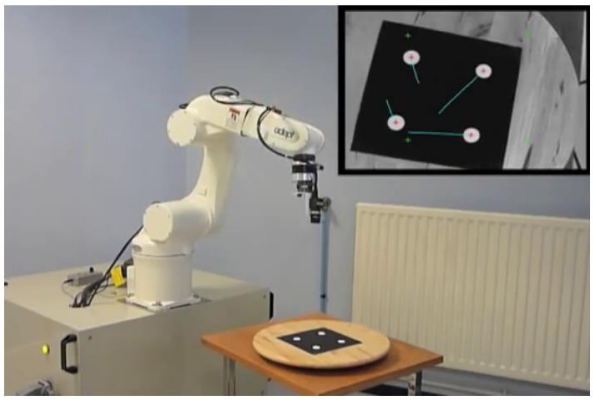
2. The retreat problem that is when the camera moves back and forth unnecessarily when it should only rotate around the optical axis [54].

3. Despite having perfect knowledge of $L$, the global asymptotic stability of IBVS technique cannot be guaranteed [2][58]. Instead, only local asymptotic stability can be achieved, and this is limited to a small neighborhood surrounding the desired pose.

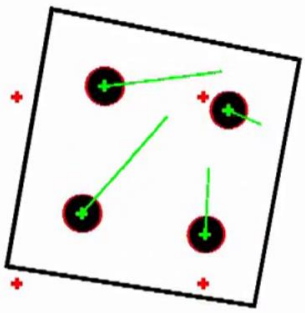
4. One approach to mitigating the singularity problem in IBVS is to incorporate additional feature points. However, using more feature points ($k > 3$) requires relying on the pseudo-inverse of the interaction matrix ($L^{+}$) or its transpose ($L^{\top}$), which introduces a higher risk of being trapped in a local minimum. This occurs when the error vector lies in the kernel of $L^{+}$ or $L^{\top}$, preventing further progress toward the desired configuration [54][56][57].

5. Since the precise interaction matrix is unavailable [58], its approximations necessitate estimating the object's depth [59]. Numerical issues in the computation could result from the error created here.

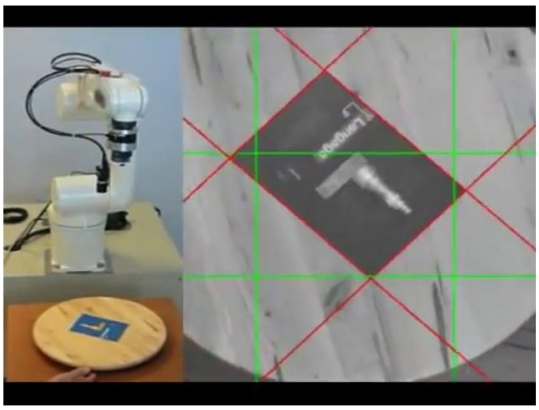
As a result, in recent years, a lot of work has been attempted to enhance IBVS performance [59][60].



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



(b) Camera view with feature trajectories



(c) Visual servoing with lines

**Figure 1.4:** Image-Based Visual Servoing with image points and straight lines as visual features [96]

#### 1.4.1.1 Interaction Matrix for an image point

Suppose that $\mathbf{X} = (X; Y; Z)$ is a 3D point in the scene whose coordinates are expressed in the camera frame. Suppose that $\boldsymbol{a} = (u_0; v_0; f; \alpha)$ is the parameters of a pin-hole camera imaging the scene, where $u_0$ and $v_0$ are the coordinates of the principal point in the image, $f$ is the camera focal length and $\alpha$ is the ratio of the pixel dimensions. The 3D scene point $\mathbf{X}$ then projects in the image plane as a 2D point $\mathbf{x}$ under perspective projection, which is given by:

$$\begin{cases} x = \frac{X}{Z} = \frac{u-u_0}{f\alpha} \\ y = \frac{Y}{Z} = \frac{v-v_0}{f\alpha} \end{cases} \tag{1.9}$$

Camera velocity is linked to the velocity of the 3D point by $\dot{\mathbf{X}} = -v_c - \omega_c \mathbf{X}$.

The construction of the interaction matrix is based on this fundamental kinematics relation, which links changes in a point in the real world with camera motions [56][2]. By expanding this relation, we get:

$$\begin{cases} \dot{X} = -v_x - \omega_y Z + \omega_z Y \\ \dot{Y} = -v_y - \omega_z Z + \omega_z Z \\ \dot{Z} = -v_z - \omega_x Z + \omega_z X \end{cases} \tag{1.10}$$

After differentiating Equation (1.9), the following is obtained:

$$\begin{cases} x = \frac{X - xZ}{Z} \\ y = \frac{Y - yZ}{Z} \end{cases} \tag{1.11}$$

After applying Equation (1.10) in Equation (1.11), we get

$$\begin{cases} \dot{x} = -\frac{v_x}{Z} + \frac{xv_z}{Z} + xy\omega_x - \ 1 + x^2 \ \omega_y + y\omega_z \\ \dot{y} = -\frac{v_y}{Z} + \frac{yv_z}{Z} + \ 1 + y^2 \ \omega_x - xy\omega_y - x\omega_z \end{cases} \tag{1.12}$$

This allows for the direct writing of the interaction matrix of an image point (that satisfies $\dot{\mathbf{x}} = L_{\mathbf{x}} v_c$):

$$L_{\boldsymbol{x}} = \begin{bmatrix} L_x \\ L_y \end{bmatrix} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \tag{1.13}$$

As a result, $L_{\mathbf{x}}$ depends on the intrinsic parameters of the camera, $\hat{\boldsymbol{a}}$, depth $Z$, and visual features $s$. Using the result of Equation (1.13), which connects the motion of the image points to the motion of the camera, the interaction matrices of intensity-based features are developed.

The interaction matrices of each individual point are stacked to produce the interaction matrix for a set of $n$ points.

$$L_s = \begin{bmatrix} L_{x_1}(x_1, Z_1, A) \\ L_{x_2}(x_2, Z_2, A) \\ \cdot \\ \vdots \\ L_{x_n}(x_n, Z_n, A) \end{bmatrix} \tag{1.14}$$

Similarly, [61] used the tracked points as visual features while transforming them to the cylindrical coordinate system $s = (\rho;\ \theta)$ with $\rho = \sqrt{x^2 + y^2}$ and $\theta = \arctan(\frac{x}{y})$. A notable inaccuracy in the rotational pose around the optical axis caused the camera to move backward along this axis. Nevertheless, this issue was successfully addressed in previous research [54], ensuring that the unintended retreat was mitigated. Beyond basic point features, various geometric elements such as ellipses, contours, and straight lines (refer to Fig. 1.4(c)) can be effectively leveraged within the image plane to represent three-dimensional structures like spheres and cylinders. These higher-level visual features provide a richer set of constraints for the servoing process, as demonstrated in prior works [46][62]. However, implementing such methods requires robust algorithms for detecting and continuously tracking these features throughout the visual servoing operation to maintain stability and accuracy.

By formulating the regulation error $e$ directly in the image space, the system ensures that the desired visual features follow well-defined trajectories at the pixel level. This approach guarantees effective convergence of the control law in the image domain. However, a limitation of this method is that it does not inherently provide direct regulation over the motion or final positioning of the end-effector in Cartesian space. Consequently, the physical path followed by the end-effector may exhibit deviations from an optimal geodesic trajectory, which can introduce inefficiencies in certain applications. Addressing these trajectory distortions remains an area of interest, with potential solutions involving model-based predictive control or hybrid strategies that integrate image-based and pose-based servoing.

#### 1.4.1.2 Image Moments-Based Visual Servoing

The application of image moments in visual servoing was first introduced in [63], where they were utilized as a compact and informative representation of image intensity distributions [64]. These statistical descriptors encapsulate essential geometric and structural characteristics of an image, making them particularly useful for controlling robotic systems through visual feedback. One of the key advancements in this domain was the derivation of the analytical form of the interaction matrix in [63]. This study demonstrated that leveraging visual features based on image moments significantly improved numerical stability while ensuring a strong decoupling effect in the servoing process.

Building on this foundation, [65] later proposed an innovative set of visual features derived from moment invariants, further enhancing robustness to variations in viewpoint and transformations. The computation of image moments can be performed either by segmenting a well-defined region within the image or by using a dispersed set of points. In the latter approach, the system operates without explicitly extracting or matching individual features, thereby reducing computational overhead and increasing efficiency. This capability makes image moments particularly attractive for real-time visual servoing applications where speed and robustness are crucial.

#### 1.4.1.3 Visual Servoing Utilizing Gaussian Mixture Models

A Gaussian mixture model (GMM) is formed by taking a convex combination of $\mathcal{N}$ Gaussian probability density functions (pdfs). The image's feature points $\boldsymbol{X}$ were extracted and then modelled as a GMM in [66]. The covariance matrix $\Sigma_k$ and mean vector $\mu_k$ of each of those $\mathcal{N}$ pdfs parameterize a GMM. Given $\boldsymbol{X}$, the EM (Expectation Maximization) technique can estimate it. The Gaussian mixture model (GMM) obtained from the feature points at both the desired and current positions is expressed as:

$$s = f\ \boldsymbol{X},\ \xi\ = \sum_{i=1}^{N} \alpha_i \mathcal{N}(\boldsymbol{X}; R\mu_i + t, R\sum_i R^T) \tag{1.15}$$

$$s^* = g\ \boldsymbol{X}\ = \sum_{i=j}^{N} \beta_i \mathcal{N}(\boldsymbol{X}; \mu_i^* + \sum_i^*) \tag{1.16}$$

Visual servoing aims to minimize the cost function, defined as the $\mathrm{L}_2$ norm between the GMMs computed for the desired and current poses.

$$\mathcal{C} = \|s - s^*\| = \int [f\ \boldsymbol{X},\ \xi\ - g\ \boldsymbol{X}\ ]^2 d\xi \tag{1.17}$$

To minimize the Lyapunov function $\mathcal{L} = \frac{1}{2}(\mathcal{C})^2$, the following control input is selected:

$$\boldsymbol{u} = -(\mathcal{C})\frac{\partial \mathcal{C}}{\partial \xi} \tag{1.18}$$

A closed-form expression for the gradient is available $\frac{\partial \mathcal{C}}{\partial \boldsymbol{\xi}}$ along with the $\mathrm{L}_2$ norm of the error provides valuable analytical insights, aiding in the optimization and stability analysis of the control system. However, achieving full six-degree-of-freedom (6-dof) control of the robot requires precise knowledge of the depth distribution associated with the selected feature points. Experimental findings from [66] indicate that the trajectory traced by the camera does not conform to a geodesic path between the initial and target poses. Additionally, the generated control signals exhibit non-smooth variations, which may introduce undesired dynamic effects during execution.

A key limitation of the algorithm is the uncertainty surrounding its convergence properties when dealing with estimation errors in the Gaussian Mixture Model (GMM) parameters $\alpha_i$ and $\beta_j$ specifically, it remains unclear whether discrepancies where $\alpha_i \neq \beta_j$ could adversely impact the stability or convergence rate of the method. Moreover, while the extraction of image points $\boldsymbol{X}$ is an inherent requirement of this approach, it operates without the need for continuous visual tracking, potentially simplifying computational demands but also introducing challenges in feature association and localization.

### 1.4.2 Pose-Based Visual Servoing (PBVS)

Due to the fact that the error in PBVS [6][67] is defined in pose space instead of image space, it is termed 3D visual servoing. The error that needs to be regulated to zero is directly linked to the robot's pose.

$$s = s\ m(\boldsymbol{I}\ t\ , \boldsymbol{A}, \boldsymbol{X} \tag{1.19}$$

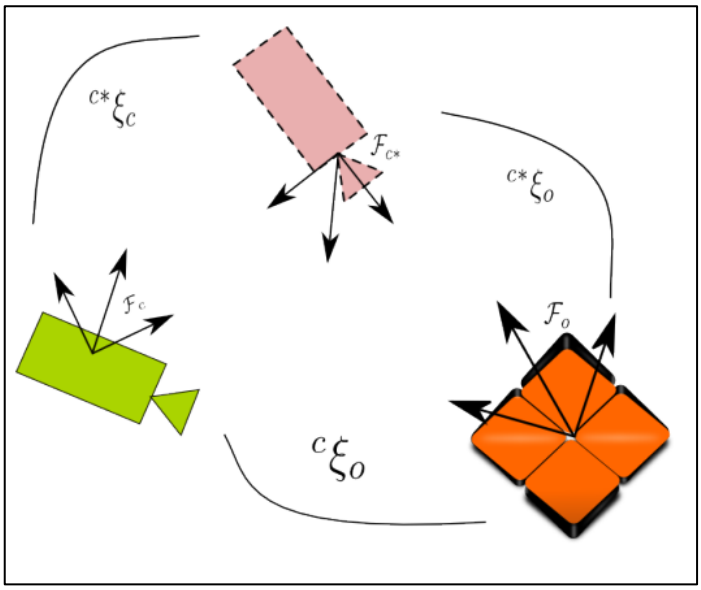



**Figure 1.5:** Pose-based visual servoing scheme

In PBVS, the accurate estimation of the camera pose ${}^{c}\xi_o(t)$ from captured images is a crucial step. This estimation depends on prior knowledge of the camera's intrinsic parameters and a predefined 3D model of the observed object. By leveraging these elements, the camera pose can be determined from a set of visual measurements captured in a single image. The challenge of determining 3D spatial coordinates from 2D projections is a well-established problem in computer vision, commonly referred to as 3D localization. While an in-depth discussion of this problem lies beyond the scope of this chapter, various approaches addressing it have been extensively documented in the literature [6][67].

Once the camera pose is estimated, its chosen parameterization is typically employed to derive the visual feature vector $s$. It is important to emphasize that the parameters $\boldsymbol{A}$ used in defining $s$, as outlined in Equation (1.19), these parameters correspond to both the intrinsic characteristics of the camera and the 3D structural representation of the object being tracked. They are essential for ensuring precise control and stabilization within the PBVS framework, as they directly affect the accuracy of the computed pose and the overall servoing performance. In robotic vision and control, it is often useful to define and work with multiple coordinate frames. Typically, three such frames are used in this context:: the reference frame $\mathcal{F}_o$, which is fixed to the object of interest; the desired camera frame $\mathcal{F}_{c*}$, and the current camera frame $\mathcal{F}_c$.

The standard convention for representing a coordinate system involves attaching a superscript to indicate the specific frame in which a given set of coordinates is defined. Consequently, the coordinates of the object's origin can be represented as vectors ${}^{c}t_o$ and ${}^{c*}t_o$,

where the former is expressed relative to the desired camera frame $\mathcal{F}_{c*}$, and the latter is in terms of the current camera frame $\mathcal{F}_{c}$.

To model the orientation of the current camera frame relative to the desired frame, the rotation matrix $\boldsymbol{R}$, defined as $\boldsymbol{R} = {}^{c*}\boldsymbol{R}_{c}$ is used, where ${}^{c*}\boldsymbol{R}_{c}$ represents the transformation from the current camera frame to the desired one. Rotation matrices in the Special Orthogonal group SO(3) can be expressed in different forms depending on the chosen parameterization. In this work, the axis-angle parameterization is employed, where $\theta u$ denotes the rotation around the axis $\boldsymbol{u}$, and $\boldsymbol{t}$ represents the translation vector.

In this framework, the control loop for the robotic system necessitates an algorithm capable of estimating the pose of the visual sensor relative to an observable target. For instance, the Extended Kalman Filter (EKF) is one common approach for pose estimation, as outlined in [6]. Other techniques for solving the pose estimation problem include model-based methods, such as those described in [68] and [69], as well as virtual visual servoing (VVS), as explored in [70]. A thorough discussion on pose estimation methods, particularly in the context of robotic vision, can be found in [71].

PBVS can be divided into two variants according to whether the control is implemented in the desired or current camera frames [52]. "If the object pose relative to the current sensor frame is selected for use, $s = ({}^{c}t_{o}; \theta u)$, $s^{*} = ({}^{c*}t_{o}; 0)$, then $e = ({}^{c}t_{o} - {}^{c*}t_{o}; \theta u)$ to perform the control. Then, the interaction matrix can be derived as

$$L_{\boldsymbol{s}} = \begin{bmatrix} -I_3 & [{}^{c}t_{o}]_{\times} \\ 0 & L_{\theta u} \end{bmatrix} \tag{1.20}$$

where

$$L_{\theta u} = I_3 + \frac{\theta}{2}[u]_{\times} + \left(1 - \frac{sinc\,\theta}{sinc^2\frac{\theta}{2}}\right)[u]_{\times}^{2} \tag{1.21}$$

with $[\mathbf{v}]_{\times}$ represents the skew-symmetric matrix of $\mathbf{v}$. sinc $\theta = \sin\theta / \theta$ is the sinus cardinal. The matrix $L_{\theta u}$ contains the property ${L_{\theta u}}^{-1}\ \theta u = \theta u$. When $\theta \neq 2k\pi$, where $k$ is a non-zero integer, the $L_{\theta u}$ matrix is non-singular. The translational velocity screw can then be computed as follows:

$$v_{c} = -\lambda\ (\ {}^{c*}t_{o} -\ {}^{c}t_{o}) + [\ {}^{c}t_{o}]_{\times}\theta u \tag{1.22}$$

As an alternative, the pose of the current sensor frame expressed in the desired sensor frame can be used to determine the control. Since $s^* = 0$, we have $s = [\,^{c*}t_c \;\; \theta u]$ and $e = s$ in this instance.

Pose estimation algorithms typically produce the target's relative pose as represented in the sensor frame $^{c}\boldsymbol{X}_{o}$. Next, the sensor's relative pose represented in the desired sensor frame is obtained using the following relation:

$$^{c*}\boldsymbol{X}_{c} = \; ^{c*}\boldsymbol{X}_{o} \; ^{c}\boldsymbol{X}_{o}^{-1} \tag{1.23}$$

The structure of the interaction matrix in this case is as follows:

$$L_{\boldsymbol{s}} = \begin{bmatrix} ^{c*}\boldsymbol{R}_{c} & 0 \\ 0 & L_{\theta u} \end{bmatrix} \tag{1.24}$$

The translational and rotational motions are decoupled, as may be seen from Eq (1.24). Here, the translational velocity screw is obtained as

$$v_c = -\lambda \; ^{c*}\boldsymbol{R}_{c} \; ^{c*}t_c \tag{1.25}$$

The matrix representing interactions for rotational movements is identical in Equations (1.20) and (1.24) [53][2]. The corresponding control law for these degrees of freedom is expressed as:

$$\omega_c = -\lambda\theta\mathbf{u} \tag{1.26}$$

In PBVS, the control laws are designed to ensure that errors in the rotational pose diminish exponentially along a geodesic path, leading to smooth convergence. This results in a more efficient alignment of the camera's pose with the target, as the system progressively reduces the rotational error over time. On the other hand, when using control law (1.24), the camera follows a straight-line trajectory, which contrasts with the behavior seen in PBVS using control law (1.20), where the camera's movement is more curvilinear and dynamic in nature.

It is important to note the differences in how depth *Z* affects the system in PBVS and IBVS. In PBVS, the depth *Z* directly influences the pose features, meaning that any change in depth alters the camera's positional features in the image plane. In contrast, IBVS simplifies this by making the features independent of depth, with the interaction matrix being the only component

that depends on $Z$. This distinction significantly influences the overall behavior and performance of visual servoing systems, especially in terms of control and trajectory planning.

Another important consideration in visual servoing is the potential for features to leave the field of view. Many studies have highlighted this limitation, noting that PBVS has limited control over trajectories in the feature space, as the features may move out of the camera's sight. However, [72] demonstrated that, despite this concern, image trajectories can indeed be controlled indirectly within certain constraints. This holds true unless major assumptions are made to simplify the interaction matrix, in which case some of the control over the trajectory might be lost. This finding challenges the traditional view of PBVS's limitations and opens avenues for more refined control strategies in visual servoing applications.

This version provides more context, expands on the distinctions between PBVS and IBVS, and further elaborates on the findings of [72].

**Stability in PBVS:**

The stability of a general dynamical system, denoted as $\mathbf{x} = f(\mathbf{x}; \mathbf{u})$, can be examined through the application of Lyapunov stability theory. The system state x is determined by the function $f$, which may be either linear or nonlinear and might not be fully characterized. The evolution of the state is influenced by the control inputs **u** and the intrinsic properties of the system, which define the form of $f$ and its interactions with the environment. According to Lyapunov, $\mathbf{x} = 0$ is an equilibrium point of the system if and only if there exists a function $\mathcal{L}$ such that $\mathcal{L} > 0$ for all **x**, and $\mathcal{L}(0) = 0$. This function is indicative of the system's energy. The energy function, referred to as the Lyapunov function of the system, should exhibit a decreasing trend along the trajectories of the system state. A straightforward Lyapunov function will be selected for the visual servoing system:

$$\mathcal{L} = \frac{1}{2}\|e\|^2 \tag{1.27}$$

Let us note that $\mathcal{L} > 0$ and $\mathcal{L} = 0$ if and only if $e = 0$. Differentiating $\mathcal{L}$ with respect to time gives:

$$\mathcal{L} = e^\top e \tag{1.28}$$

Using (1.3) in (1.28):

$$\dot{\mathcal{L}} = e^\top L_s v_c \tag{1.29}$$

Substituting (1.6) in (1.29) yields to:

$$\mathcal{L} = -\lambda e^{\top} L_s \widehat{L_s^{+}} e = -\lambda e^{\top} Q e \tag{1.30}$$

where $Q = (L_s \widehat{L_s^{+}})$. According to the fundamental theorems of Lyapunov stability, if the Lyapunov function $\mathcal{L}$ ($e$; $t$) > 0 and its derivative $\mathcal{L}$ ($e$; $t$) < 0 in a neighborhood around the equilibrium point, then the equilibrium point (desired system state) is considered locally stable in the Lyapunov sense. By definition, Equation (1.31) is positive. Therefore, for the system to achieve stability, Equation (1.30) must be negative. This condition is satisfied if and only if the matrix $Q$ is positive definite (PD), denoted $Q \succ 0$. However, it is insufficient to merely assert the condition $L_s \widehat{L_s^{+}} \succ 0$ and draw conclusions about stability without examining the specifics of the number of features and the neighborhood where the condition is applicable. The stability criterion can be expressed by

$$L_s \widehat{L_s^{+}} \succ 0 \quad \textit{when} \;\; k \leq n \tag{1.31}$$

where $n$ represents the number of actuated degrees of freedom. If this condition is met, then, as previously stated, the derivative of the Lyapunov function will be negative for all system states except at the desired equilibrium where $e$ = 0. In other words, the following condition is satisfied:

$$\mathcal{L} < 0 \qquad \forall e \neq 0 \tag{1.32}$$

If condition (1.32) is satisfied for configurations throughout the robot workspace, global asymptotic stability (GAS) is guaranteed. Conversely, if condition (1.32) is valid only within a vicinity of the equilibrium state, this is referred to as local asymptotic stability. Although the size of this vicinity has not been analytically determined, it is notably large for visual servoing methods [2].

Since $L_{\mathbf{s}} \widehat{L_s^{+}} = \mathbf{I}_6$, the stability requirement in Equation (1.31) for PBVS can be easily ensured, assuming that the pose is precisely estimated [2]. This holds true for every $\boldsymbol{\theta} \neq 2k\pi$ as previously stated in Equations (1.20), and (1.24), when $L_{\theta u}$ is non-singular. Visual feature measurements in the image are used to estimate the pose features used in the control law. The above strong assumption is broken when these measurements are inaccurate since the pose estimation is not precise. Therefore, inaccuracies in pose estimation have a negative impact on the PBVS method.

A concise overview of IBVS and PBVS, including their relative advantages and stability characteristics, is provided in [2], to which the reader is referred. A formal, comprehensive treatment is provided in [72], where a common framework for comparing IBVS and PBVS is given from the perspectives of performance in image and Cartesian spaces, stability, robustness, and sensitivity to modeling errors.

### 1.4.3 Hybrid Visual Servoing

In hybrid visual servoing, the measurement vector **x** integrates both three-dimensional spatial information and image-based primitives. This approach represents a balanced compromise between two-dimensional (2D) and three-dimensional (3D) visual servoing, aiming to simultaneously control both the camera's trajectory within its workspace and the movement of certain image-based primitives. Hybrid visual servoing allows for more flexibility by combining the advantages of both 2D and 3D techniques, ensuring that both the camera and the features it observes are actively managed during the task.

The first scheme of hybrid visual servoing, known as 2D1/2, was introduced by Malis in 1998 [73]. This method provides a partial reconstruction of the object's pose without requiring prior knowledge of the full 3D model of the object. Instead, it relies on the estimation of the homography matrix, or a homogeneous transformation matrix, which describes the relationship of the object to a reference plane. This approach allows for pose estimation by leveraging available image information, reducing the computational burden associated with full 3D model reconstruction.

In the 2D1/2 scheme, the task function is designed with two distinct components. The first component involves the image coordinates of a point, along with a term that accounts for the relative depth of the point between the current and desired camera poses. This depth component is crucial for tracking the 3D position of the point in the image space. The second component of the task function addresses the rotational aspect, representing the rotation between the current and desired camera poses. This dual-component structure enables the hybrid visual servoing system to manage both the motion in the image plane and the 3D trajectory of the camera, achieving precise control over both aspects simultaneously.

This expanded explanation provides more insight into the hybrid visual servoing method, detailing the functionality of the 2D1/2 approach and its components. $s$ is then defined as follows: $s = [s_t; \theta\mathbf{u}]$ where $s_t$ is a set of 2D primitives allowing the control of translational movements and θu is the rotation between the current and desired poses, typically used in PBVS approaches. In hybrid visual servoing, the visual information vector $s$ is constructed by combining both 2D and 3D data, enabling the system to leverage the advantages of both

domains. This combination provides a more comprehensive representation of the visual task, allowing for the control of both image-based features and spatial trajectories. The interaction matrix corresponding to this task function takes on a simplified and partially decoupled structure, which offers analytical benefits, such as facilitating formal stability analysis in the presence of modeling errors. This feature makes hybrid visual servoing appealing in situations where a balance of accuracy and efficiency is required, particularly when dealing with imperfect models or noisy data.

However, one of the challenges with this approach is that it requires a partial reconstruction of the camera displacement at each iteration of the control loop. This step, although necessary for accurate tracking and control, is inherently unstable, especially as the system approaches convergence. The instability is particularly pronounced when the camera nears the desired pose, where small errors in the pose estimation can have a significant impact on the system's behavior, leading to oscillations or even divergence if not carefully controlled.

Additionally, maintaining the visibility of all the required image primitives for estimating the homography matrix is not trivial. The system relies on the availability of certain features in the image, and ensuring that these features remain within the camera's field of view throughout the operation is a key challenge. This becomes even more complex in dynamic environments or situations where the object's pose changes rapidly, as the camera may lose sight of essential features, leading to inaccuracies in the homography estimation. Therefore, careful planning of the camera's trajectory and the selection of robust visual features are critical for ensuring the success of the hybrid visual servoing method.

The PBVS control law was found to produce rotational errors that follow a geodesic, as described in Section 1.4.2. The following visual features were suggested by the 2.5D approach [53]:

$$s = \begin{bmatrix} s_t \\ \theta u \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ log\, Z \\ \theta u \end{bmatrix} \tag{1.33}$$

$$s = \begin{bmatrix} L_v & L_\omega \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = L_v v_c + L_\omega \omega_c \tag{1.34}$$

where the feature set responsible for controlling the translational motions is denoted by $\mathbf{s}_t$. It includes $Z \in \mathbb{R}$, which is the depth of that image point, and $\mathbf{x} \in \mathbb{R}^2$, which is an image point used in IBVS (as in Section 1.4.1). The rotational pose used in PBVS is $\theta\mathbf{u} \in \mathbb{R}^3$. The interaction matrix for the feature log $Z$ is given by

$$L_{log\,Z} = \frac{1}{Z}[0 \quad 0 \quad -1 \quad -y \quad x \quad 0] \tag{1.35}$$

By combining (1.13) and (1.35), the translational control input's interaction matrix can be expressed as follows:

$$L_{s_t} = \left[ \frac{1}{Z} \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} xy & -(1+x^2) & y \\ 1+y^2 & -xy & -x \\ -y & x & 0 \end{bmatrix} \right] \tag{1.36}$$

The 3 × 3 matrix on the left has a block triangular structure. The diagonal values indicate a significant correlation between the relevant translational degree of freedom and the selected features. Equation (1.20), therefore, shows that the rotations and translations are decoupled. The 2.5D VS is globally asymptotically stable, assuming that the pose estimation algorithm is precise and there are no coarse modeling errors [74].

### 1.5 Photometric Visual Servoing Methods

According to Sonka [75], a camera image is essentially a spatial sampling of brightness variations, which are influenced by the geometry of the scene, the ambient lighting conditions, and the reflectance properties of the objects in view. In feature-based visual servoing approaches, after image processing, the original image is typically reduced to a set of discrete points or geometric primitives. This reduction leads to a loss of rich information present in the image, such as intricate intensity patterns, which are discarded and not utilized in further processing.

In contrast, photometric visual servoing methods exploit the intensity information directly from the image. These techniques focus on the pixel intensity values rather than geometric features like points, straight lines, or homographies, and are sometimes referred to as "correspondenceless" visual servoing methods. The term "correspondenceless" arises because these methods do not rely on the need for feature tracking or matching, which simplifies the process and eliminates the computational complexity typically associated with these tasks.

One of the main advantages of photometric visual servoing is its robustness. It does not require precise depth information to calculate the interaction matrix, and it is less sensitive to errors in coarse depth approximations. Additionally, photometric techniques are more resilient to partial occlusions of the scene, as the intensity values remain available even when some

features are obscured. This makes them particularly advantageous in real-world applications where occlusions and imprecise depth data are common challenges.

In general, methods that utilize the image's intensity information are grouped under the category of photometric visual servoing. These methods allow for more flexible and efficient control of robotic systems, as they avoid the need for explicit feature extraction and matching, offering a practical solution in scenarios with limited or challenging visual features.

### 1.5.1 Direct Visual Servoing

Direct visual servoing techniques do not rely on tracking or matching intermediate geometric entities in the image. Instead, the control law directly utilizes pixel intensity. Determining appropriate control inputs for visual servoing essentially involves solving a nonlinear optimization problem.

$$\boldsymbol{\xi} = arg\min_{\boldsymbol{\xi}} \mathcal{C}\ \boldsymbol{I}\ \boldsymbol{\xi}\ , I^* \tag{1.37}$$

with

$$\mathcal{C}\ \boldsymbol{\xi}\ = \tfrac{1}{2}\| e\ \boldsymbol{I}\ \boldsymbol{\xi}\ , I^*\ \|^2 \tag{1.38}$$

Equation (1.38) defines the cost function to be minimized, which aligns with the Lyapunov function used to ensure stability. Here, $\xi$ signifies the robot pose, while $e$ denotes the error measure computed from the current image and the image associated with the desired pose ($I^* = I(\xi^*)$). In direct visual servoing techniques, the error $e$ is not based on measurements derived from geometric features extracted from the image. Instead, it is a photometric error, defined by the intensity distribution of the image

#### 1.5.1.1 Pure Photometric Visual Servoing

The difference in intensities (luminance) between the images captured at the desired and present poses is the direct definition of the error function in pure photometric visual servoing [Collewet 11]: $e = I(\xi) - I(\xi^*)$. The cost function to be optimized is the following:

$$\xi = arg\min_{\xi} \sum_{\boldsymbol{x}} [I\ \mathbf{x}(\boldsymbol{\xi})\ - I\ \mathbf{x}(\boldsymbol{\xi}^*)\ ]^2 \tag{1.39}$$

where $\mathbf{x} = (x; y) \in \mathbb{R}^2$ denotes the pixel indexing variable. The KLT technique uses the SSD (Sum of the Squared Differences) function for image registration, and this function is exactly

the error norm [76].

The total number of pixels observed within the image, or a subset of them, is a subset of visual features.

$$s = I = \begin{bmatrix} I(x_1) \\ I(x_2) \\ \cdot \\ \vdots \\ I(x_N) \end{bmatrix} \tag{1.40}$$

where $N$ is the number of pixels. This subset must be made available frame-by-frame when only a particular subset of the image's pixels are used. The derivative of $s$ can be used to determine the interaction matrix associated with pixel illumination.

$$\boldsymbol{I} = L_I v_c = [L_I \; x_1 \; , \ldots , L_I(x_N)]^T v_c \tag{1.41}$$

The interaction matrix of an individual image pixel's luminance was developed in [12] using the traditional brightness constancy assumption [77]. This constraint states that a moving image point's brightness remains constant during a short interval of time $\delta t$.

$$I(x; y; t) = I(x + \delta x; y + \delta y; t + \delta t) \tag{1.42}$$

Taylor's expansion of this equation at the first order yields:

$$\nabla I_x \mathbf{x} + \nabla I_y \mathbf{y} \; + \; I = 0 \tag{1.43}$$

where $\nabla I^T = [\nabla I_x \quad \nabla I_y]^T = \left[\frac{\partial I}{\partial x} \quad \frac{\partial I}{\partial y}\right]^T$ is the gradient of the image pixel at $(x, y)$.

After that, it is simple to derive the interaction matrix of the luminance of pixel $(x, y)$.

$$L_I \; \mathbf{x} \; = -\nabla I^T L_{\mathbf{x}} \tag{1.44}$$

Here, $L_{\mathbf{x}}$ represents the interaction matrix for the image point, as derived in Section 1.4.1 and given in Equation (1.13). An important observation is that the depth $Z$ of the image point, which is transferred from $L_{\mathbf{x}}$ to $L_I$, remains present. Additionally, when surfaces and light sources move relative to one another, the brightness constancy assumption may no longer hold, even

for Lambertian surfaces. In such cases, the process of deriving the interaction matrix becomes more complex. To address this, the Phong Illumination Model [78] is employed to derive the interaction matrix, as detailed in [79].

In non-linear optimization problems, the convergence to an optimal solution is heavily influenced by the initial conditions and the distance between the starting point and the target state. This principle also applies to visual servoing, which, depending on the method chosen to tackle the problem, presents both advantages and challenges. A variety of iterative non-linear optimization techniques are available for addressing problems like those formulated in Equation (1.38) [80]. These methods generally revolve around selecting a specific direction of descent within the solution space, thereby guiding the process towards achieving the optimization goal. The control mechanism employed in this context typically takes the following form:

$$\mathrm{v} = \lambda \mathrm{d}(\boldsymbol{\xi}) \tag{1.45}$$

where $\boldsymbol{\xi}$ indicates robot posture and $\mathbf{d}(\boldsymbol{\xi})$ is the direction of descent. The following control law is derived using the Levenberg-Marquardt method of descent.

$$v = -\lambda \left( \boldsymbol{H} + \mu dig(\boldsymbol{H})^{-1} \widehat{L_I^+} (I\ \boldsymbol{\xi}\ - I\ \boldsymbol{\xi}^*\ ) \right)$$

(1.46)

where the scalars $\lambda$ and $\mu$ are positive. $\boldsymbol{H}$ is an approximation of the Hessian matrix that is $\boldsymbol{H} = L_I{}^{\mathbf{T}} L_I$, where $L_I$ is the interaction matrix at any pose $\boldsymbol{\xi}$.
This approach has high accuracies at convergence since it uses a redundant feature set and does not use intermediate measures from the image.

The degree of planar rotation error between the images of the initial and final robot poses decreases the convergence domain [12]. The significant degree of non-linearity in Equation (1.44) helps to explain this. Conversely, simpler solutions (such as the Gauss-Newton method) are adequate for the servoing in the case of classical visual servoing methods.

The study in [12] demonstrated that a robot motion could be controlled by the pixel intensity of the image. Additionally, visual servoing was shown without the use of feature tracking or detection. However, the gradients in (1.44) had to be calculated as part of the image processing step. Only local stability may be demonstrated, much like with other IBVS methods. Additionally, the method has a problem with lighting variations, which directly affect the cost function's minimum.

## 1.6 Conclusion

This chapter has presented a comprehensive overview of the visual servoing control paradigm, with particular emphasis on its fundamental configurations, modeling tools, and visual feature representations. The two main camera configurations—eye-in-hand and eye-to-hand—were examined, highlighting their respective advantages and limitations depending on task requirements, workspace constraints, and precision needs.

The chapter also detailed the role of the interaction matrix, a central element in visual servoing that links the temporal evolution of visual features to the camera's kinematic motion. Understanding this relationship is essential for designing effective control laws capable of driving robotic systems toward desired visual configurations using real-time feedback.

Different categories of visual features were reviewed, distinguishing between geometric approaches, which rely on explicit feature extraction and tracking, and direct approaches, which exploit global image information without intermediate processing. While geometric methods can offer high accuracy, they often require complex and computationally demanding image-processing pipelines. Direct methods, by contrast, provide a simpler and more efficient alternative, albeit with increased sensitivity to photometric disturbances.

Particular attention was given to photometric visual servoing, which leverages raw image intensities to achieve high precision through data redundancy. Despite its effectiveness under stable lighting conditions, its sensitivity to illumination variations limits robustness in dynamic environments. Alternative strategies, such as Gaussian mixture-based methods, were also discussed, demonstrating improved convergence properties in situations involving lighting changes or partial occlusions.

Overall, this chapter has highlighted that the choice of visual features remains a critical and unresolved issue in visual servoing, directly impacting stability, convergence, and robustness. Addressing this challenge is especially important in complex and unstructured environments. The concepts and limitations identified here establish the foundations for the subsequent chapters, which investigate advanced modeling and control strategies aimed at improving robustness while preserving the advantages of direct visual servoing approaches.

CHAPTER 2

# LIFTING WAVELET TRANSFORM

**Contents**

## 2.1 Introduction

Wavelet transforms are essential tools for signal and image analysis. They enable the decomposition of a signal into distinct frequency components, thus facilitating multi-resolution analysis. Typically, classical wavelet constructions employ Fourier transformations to provide basis functions, which are based on the translation and dilation principles. These approaches have issues with irregularly sampled data, adaptive processing, and computational efficiency. The Lifting Wavelet Transform (LWT) solves these limitations by offering a more flexible and computationally efficient framework. Initially developed to optimize existing wavelet transforms, the Lifting Scheme evolved into a method for constructing second-generation wavelets, which depart from strict translation and dilation constraints. By dividing wavelet transforms into elementary lifting steps, this method preserves excellent reconstruction qualities while lowering computing complexity.

The first section of this chapter considers generalizations about the wavelet transform, including historical wavelet families, types of wavelets, and the paramount question of selecting the optimal wavelet. The second section of this chapter considers the conception of lifting technique, which is a way to generate wavelets, and that includes the phases.

### 2.2 Wavelet Transform

The term wavelet refers to a function that oscillates during a “given time” (if the variable is time) or over a finite length interval (if the variable is of spatial type). Beyond that, the function decreases very quickly towards zero. Wavelets are primarily employed for the decomposition of functions. This process involves representing a function as a weighted sum of functions derived from basic operations performed on a primary function known as the mother wavelet. These operations encompass translations and dilatations of the variable.Depending on whether these translations and dilatations are chosen to be continuous or discrete, the resulting transform is referred to as a continuous or discrete wavelet transform, respectively.

#### 2.2.1 Continuous Wavelet Transform

A wavelet transform is said to be continuous when the structural parameters of the functions used (i.e. the translations and dilations) can take any value from the set of real numbers R (the dilations must nevertheless be positive). Wavelets were introduced by Grossmann and Morlet [83] as a family of functions $\Psi_{a,b}$ of $\boldsymbol{L^2(\mathbf{R})}$. They are generated by translations and dilations of a finite energy basis function (belongs to $\boldsymbol{L^2(\mathbf{R})}$) called the mother wavelet *y*.

$$\psi_{a,b}\ t\ = \frac{1}{\sqrt{|a|}}\psi\ \frac{t-b}{a}\ , \qquad a, b\ \epsilon\ \mathbb{R}, a \neq 0 \tag{2.1}$$

The coefficients *a* and *b* denote respectively the scale factor for the dilation of *y*, and the translation coefficients. If the function ψ (x) is integrable and sufficiently oscillating to be of zero integral:

$$M_k = \int_{-\infty}^{+\infty} t^k \psi\ t\ dt = 0 \qquad , 0 \leq k \leq N \tag{2.2}$$

The continuous wavelet transform (CWT) of a function f is defined by:

$$W_f(a, b) = \langle f\ ,\ \psi\ a, b \rangle \tag{2.3}$$

where the symbol $\langle . \rangle$ corresponds to the scalar product in $L^2()$.

If the wavelet ψ satisfies the admissibility condition

$$C_\psi = \int_0^\infty \frac{|\widehat{\psi(\omega)}|^2}{\omega} d\omega < +\infty \tag{2.4}$$

Then the continuous wavelet transform *W* (*a*, *b*) admits an inverse

$$x\ t\ = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^{+\infty} X_{WT}\ a, b\ \frac{1}{a^2} \psi\ \frac{x-a}{b}\ dbda \tag{2.5}$$

From the admissibility condition, $\widehat{\psi}$ (0)= 0 is observed, implying that ψ must oscillate, hence its name "wavelet". It should be noted that the CWT converts a one-variable function into a two-variable function. The representation of a function by its CWT is redundant and the inverse transform is not always unique. Moreover, not all functions *W*(*a*, *b*) are necessarily the CWT of the function *f*.

The advantage of this characterization over the Fourier transform is that it not only gives information on the type of singularity but also its location in time.

### 2.2.2 Properties of a first generation wavelets

- **Zero moments:** this is the most important property for a wavelet. a wavelet ψhas N zero moments if:

$$M_k = \int_{-\infty}^{+\infty} t^k \psi\ t\ dt = 0 \qquad , 0 \leq k \leq N$$

According to the last equation $M_k$, any wavelet must have at least one zero moment (the case where $k = 0$).

- **Compact stand:** as long as the wavelet has less zero moments and its support is compact, a more exact high frequency analysis is possible. Wavelets like Daubechies' family exhibit compact support, making them highly localized in time and computationally efficient.
- **Regularity:** The regularity of a wavelet is the property allowing to locate singularities in a signal. It can be noted that there is a link between regularity and null moments. As much as we have null moments as much the signal is regular.
- **Symmetry:** like the number of zero moments, the symmetry of the wavelet conditions its regularity over an interval.
- **Orthogonality:** The orthogonality of a wavelet is the property allowing eliminate information redundancy. Many wavelet bases, such as Haar and Daubechies wavelets, are orthogonal, which ensures energy conservation and the ease of reconstruction, hence stable computational results.

### 2.2.3 Discrete Wavelet Transform DWT

The Discrete Wavelet Transform (DWT) has been demonstrated to offer a computationally efficient alternative to the CWT by discretizing the scale and translation parameters. In the discrete case, the values of the parameters *a* and *b* are restricted to a discrete grid. In this case a dilation step $a_0$>1 and $b_0$>0 is fixed and by setting $a = a_0^{\mathrm{m}}$ and $b = \mathrm{n}b_0a_0^{\mathrm{m}}$ with m and $n \in$Z. The wavelet bases are defined by the functions [84].

$$\psi_{m,n}\ x\ = a_0^{-\frac{m}{2}}\psi\ a_0^{-m}x - nb_0 \tag{2.6}$$

Therefore, the wavelet decomposition and the inverse transform are determined by the following equations:

$$W_f(m,n) = \langle f, \psi_{m,n} \rangle = \int_R f(x)\psi^*_{m,n}(x)dx \tag{2.7}$$

$$f = \sum\nolimits_{m,n\ \in Z^2} W_f\ m,n\ \psi_{m,n}\ \ cover\ in\ \ L^2(R) \tag{2.8}$$

Thus, the wavelet transform associates with the function $f \in L^2(\mathrm{R})$ a discrete set of coefficients $W_f$ (m, n). [85] constructed a dyadic wavelet transform for which $a_0$=2 and $b_0$=1, showed that for particular functions $y$ of $L^2(\mathrm{R})$, the family:

$$\psi_{m,n}\ x\ = 2^{-\frac{m}{2}}\psi\ 2^{-m}x - n \qquad \forall\ m,n\ \in Z^2 \tag{2.9}$$

$\psi_{\mathrm{m},n}$ Constitutes an orthonormal basis of $\mathrm{L}^2(\mathrm{R})$.

### 2.3 Multi-resolution analysis

The multi-resolution analysis method is a mathematical structure formalized Mallat on the finite real axis [84][86][21][87]. In this regard, wavelets provide an effective instrument. The fundamental concept of multi-resolution analysis involves the hierarchical representation of a function by a lower-resolution component, which is characterized by the wavelet coefficients necessary to restore the original function [88].

#### 2.3.1 Theoretical framework

The notion of multiresolution wavelet analysis of a function $f \in L^2(\mathrm{R})$ consists in its projection onto function bases, giving increasingly less fine approximations of the original function. A sequence $\{V_j\}_{j\in Z}$ of closed subspaces of $L^2(\mathrm{R})$ is a multiresolution approximation if it verifies the following properties: [89][90]

1- $V_j$ is invariant under translation of length $2^j$

$$\forall (j,k) \in Z^2, f(t) \in V_j \iff f(t-2^j k) \in V_j \tag{2.10}$$

2- Causality property: the spaces $\{V_j\}_{j\in Z}$ define a sequence of nested spaces.

$$V_{k+1} \subset V_k \tag{2.11}$$

3- The details between $V_{j+1}$ and $V_j$ are increased by a factor of 2.

$$\forall j \in Z,\ f(t) \in V_j \iff f(t/2) \in V_{j+1} \tag{2.12}$$

4- The intersection of the nested spaces contains only the zero element.

$$\lim_{j\to+\infty} V_j = \bigcap_{j\to-\infty}^{+\infty} V_j \tag{2.13}$$

5- The union of nested spaces $\{V_j\}_{j\in Z}$ is dense in $L^2(\mathrm{R})$. In other words, the adherence of this union is identified with the space $L^2(\mathrm{R})$.

$$\lim_{j\to+\infty} V_j = \overline{\bigcup_{j\to-\infty}^{+\infty} V_j} = L^2(R) \tag{2.14}$$

6- There exists $f$ such that $\{\phi(t-n)\}_{n\in Z}$ is a Riesz basis

Condition (6) implies that an orthonormal basis of $V_j$ can be deduced from the basis of $V_0$ from a single function $\phi(t)$, the scaling function, by constructing the family:

$$\phi_{j,n}(t) = 2^{-\frac{j}{2}}\phi(2^{-j}t - n) \tag{2.15}$$

Since operations are performed in nested spaces, any function in $V_j$ can be expressed as a linear combination of the functions in $V_{j-1}$ . In particular $\phi \in V_0 \subset V_{-1}$, and subsequently the existence of a sequence $h \in L^2(\mathrm{R})$ such that

$$\phi(t) = \sqrt{2}\sum_{n=-\infty}^{+\infty} h(n)\phi(2t-n) \tag{2.16}$$

where the $h(n)$ are the coordinates of $\phi(t)$ in the basis $\{2\phi(2t-k), k \in Z\}$, or the coefficients of the coarse discrete filter. These $h(n)$ verify the following two properties, for an orthonormal basis:

$$\phi(t) = \sqrt{2}\int_{-\infty}^{+\infty} h(n)\phi(2t-n) \quad with \quad \sum_{n=-\infty}^{+\infty} h^2(n) = 1 \tag{2.17}$$

Mallat combines the concept of multiresolution analysis and wavelet theory by considering wavelets as functions containing the details needed to move from a coarse resolution to a finer resolution.

The orthogonal complement of $V_j$ in $V_{j-1}$ is defined by the vector space $W_j$

$$V_{j-1} = V_j \oplus W_j \tag{2.18}$$

The wavelets $(\psi_{j,n})_{n\in Z}$ belong to the space $W_j$. Under certain conditions, they constitute an orthonormal basis of $W_j$, the orthogonal projection of a function $f$ on $V_{j-1}$ can be written:

$$projv_{j-1}\ f\ = projv_j\ f\ + \sum_{n\in Z} Wf(j,n) \tag{2.19}$$

The odelet associated with this multi-resolution analysis is determined by the existence of a sequence $(g_n)_{n\in Z} \in L^2(\mathrm{R})$ such that

$$\psi(t) = \sqrt{2}\sum_{k=-\infty}^{+\infty} g(n)\phi(2t-n) \tag{2.20}$$

The orthogonality condition on the spaces $V_j$ and $W_j$ allows to establish the relationship between the two sequences $h$(n) and $g$(n) [33].

$$g(n) = \ -1\ ^n h(-n+1) \tag{2.21}$$

The biorthogonal multiresolution analysis is defined using the two spaces $\{V_k\}_{k\in Z}$ , $\{V_k\}_{k\in Z}$. The associated scaling functions $\{\phi_{j,n}\}_{n\in Z}$ and $\{\phi_{j,n}\}_{n\in Z}$ constitute the two non-orthogonal bases of the approximation spaces $V_k$ and $V_k$, the biorthogonal wavelet bases $\{\psi_{j,n}\}_{n\in Z}$ and $\{\psi_{j,n}\}\}_{n\in Z}$ constitute the bases of the complementary spaces of $V_k$ and $V_k$in $V_{k-1}$ and $V_{k-1}$ i.e. the spaces $W_k$ and $\widetilde{W}_k$

$$V_{k-1} = V_k \oplus W_k \qquad\qquad \widetilde{V}_{k-1} = \widetilde{V}_k \oplus \widetilde{W}_k \tag{2.22}$$

The spaces are not orthogonal to their complement, but to the complement of the dual space. $V_k \perp W_k$ And $V_k \perp \widetilde{W}_k$ the biorthogonal conditions translate:

$$\langle \phi_{j,k}, \widetilde{\phi}_{j,k'} \rangle = \delta_{k-k'} \qquad \forall j,k,k' \in Z \tag{2.23}$$

$$\langle \psi_{j,k}, \widetilde{\psi}_{j,k'} \rangle = \delta_{k-k'} \qquad \forall j,k,k' \in Z \tag{2.24}$$

### 2.3.2 One-dimensional fast wavelet transform

In practice, the signals that are encountered are discrete. Consequently, it is necessary to implement a discrete version of the wavelet transform. This discrete version can be derived

from the scale equations. To illustrate, consider the scale function $\phi(t)$. For a discrete signal $C = \{c_k | \ k = \cdots, -2\ , -1, 0\ , 1, 2\ , \ldots\}$, on associate C with a function $f(t)$ in $V_0$: $f\ t\ = \sum_{k=-\infty}^{+\infty} c_k \phi(t-k)$

Mallat created the fast wavelet transform algorithm, which uses the wavelet function $\psi(t)$ to express the signal $f(t)$. The following is the definition of the algorithm:

$$c_{j,k} = \sum_{m=-\infty}^{+\infty} h_{m-2k} c_{j-1,m} \tag{2.25}$$

$$d_{j,k} = \sum_{m=-\infty}^{+\infty} g_{m-2k} c_{j-1,m} \tag{2.26}$$

Based on the two earlier equations:

$$\boldsymbol{C_j} = \{c_{j,k} \backslash k = \cdots\ , -2, -1, 0, 1, 2, \ldots\ \} \tag{2.27}$$

$$\boldsymbol{D_j} = \{d_{j,k} \backslash k = \cdots\ , -2, -1, 0, 1, 2, \ldots\ \} \tag{2.28}$$

are respectively the result of the convolution of $C_{j-1}$ with the filters $H^* = \{h_{-k} | \ k = \cdots, -2\ , -1, 0, 1, 2, \ldots\}$ and $G^* = \{g_{-k} | \ k = \cdots, -2\ , -1, 0\ , 1, 2\ , \ldots\}$ followed by down-sampling of factor 2. The previous two equations can be written more compactly as follows:

$$C_j = H * C_{j-1} \tag{2.29}$$

$$D_j = G * C_{j-1} \tag{2.30}$$

The process of decomposition starts from $C_0 = C$ up to J levels of decomposition. Subsequent to level J levels, undergoes a transformation, resulting in a sequence of novel signals: $\{\ \mathbf{C_j}; \mathbf{D_j}; \mathbf{D_{j-1}}; \ldots;\ \mathbf{D_1}\}$.

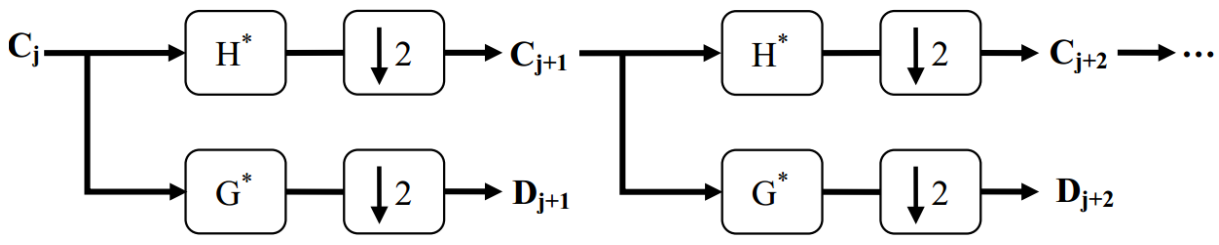



**Figure 2.1:** Discrete signal decomposition algorithm.

### 2.3.3 Principle of decomposition and reconstruction in the case of 2-D transform

The principle of decomposition and reconstruction for the 2-D transform is presented in a general manner in Figs. 2.2 and 2.3 [33].

The wavelet decomposition of an image (N×N) therefore proceeds as follows:

- First, each column of the 2-D image is decomposed using vertically the 1-D filters. This produces two images (N×N/2), one corresponding to the low vertical frequencies, the other to the high vertical frequencies.
- Secondly, the same 1-D filters are applied to the lines of these two images. Each of them is decomposed into two new images (N/2×N/2), one corresponding to the low horizontal frequencies, the other to the high horizontal frequencies.

In total, four images (LL, HL, LH, HH) are therefore generated at each level of decomposition. The following decomposition performs the same process on LL, corresponding to the low horizontal and vertical frequencies.

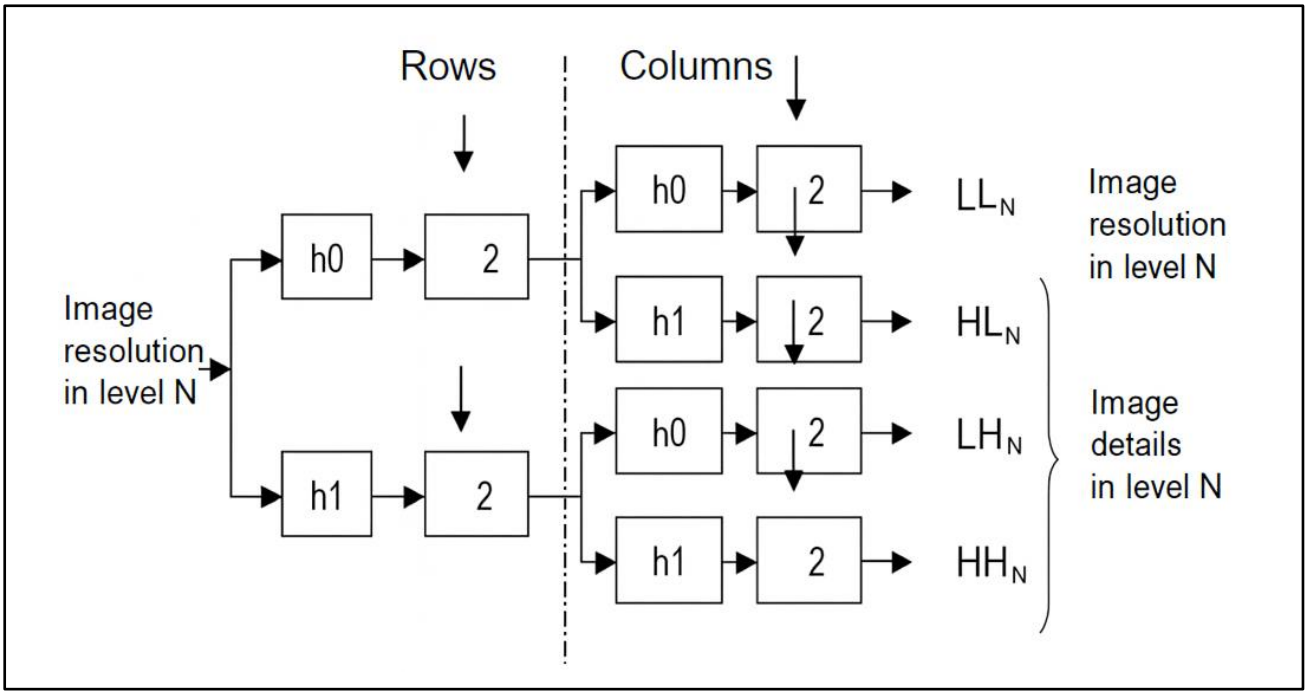



**Figure 2.2:** Filter Bank analysis for the wavelet transform

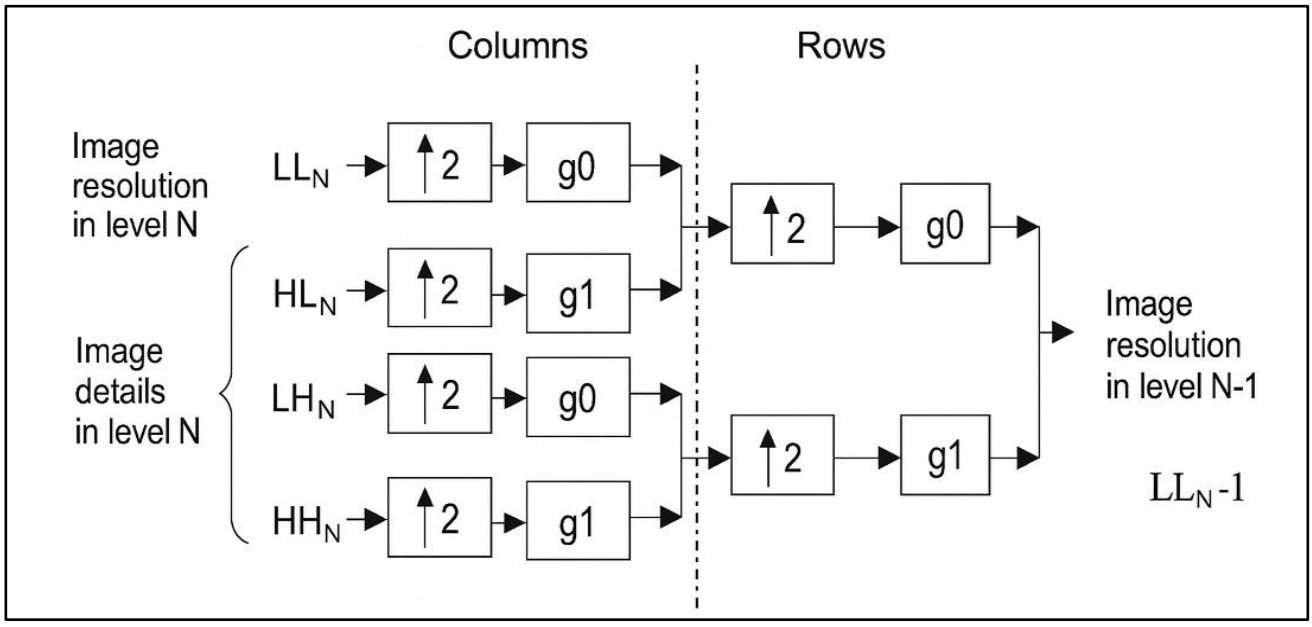



**Figure 2.3:** Inverse Wavelet Transform

### 2.4 Lifting Scheme Theory

In order to create and apply wavelet transforms, the lifting scheme is offered as an effective tool. This leads to the creation of so-called "second generation" wavelets. Consequently, the lifting scheme will be described as a general and flexible technique.

Originally, the aim of the lifting scheme, introduced by Sweldens [91], was to propose a method for constructing biorthogonal wavelets whose moments cancel each other for increasingly higher orders: "a lift" towards high zero moments. It is an interesting alternative to the classic convolutional filtering scheme of the transform, because it is much less complex. Indeed, the number of operations is divided by a ratio of up to two compared to a classic scheme. Among the advantages of lifting, the following may be noted:

- The possibility of measuring wavelet transforms in points without going through the Fourier domain
- Obtaining a lower computational complexity and a simpler and more efficient implementation.
- The possibility of implementing the inverse transform, on the other hand in the case of classical wavelets it is not obvious that the synthesis filters are exactly the inverse of the analysis filters.

The lifting scheme [92][93][94][95][91] is carried out in three steps presented in Figs. 2.4, 2.5:

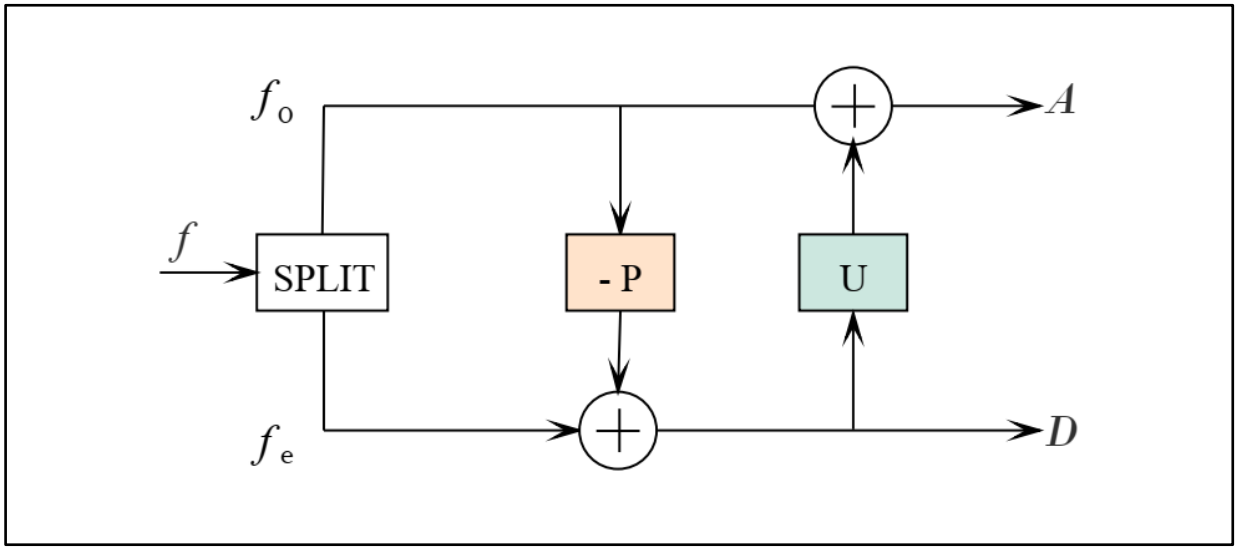



**Figure 2.4:** Block diagram of 1D lifting scheme steps [96]

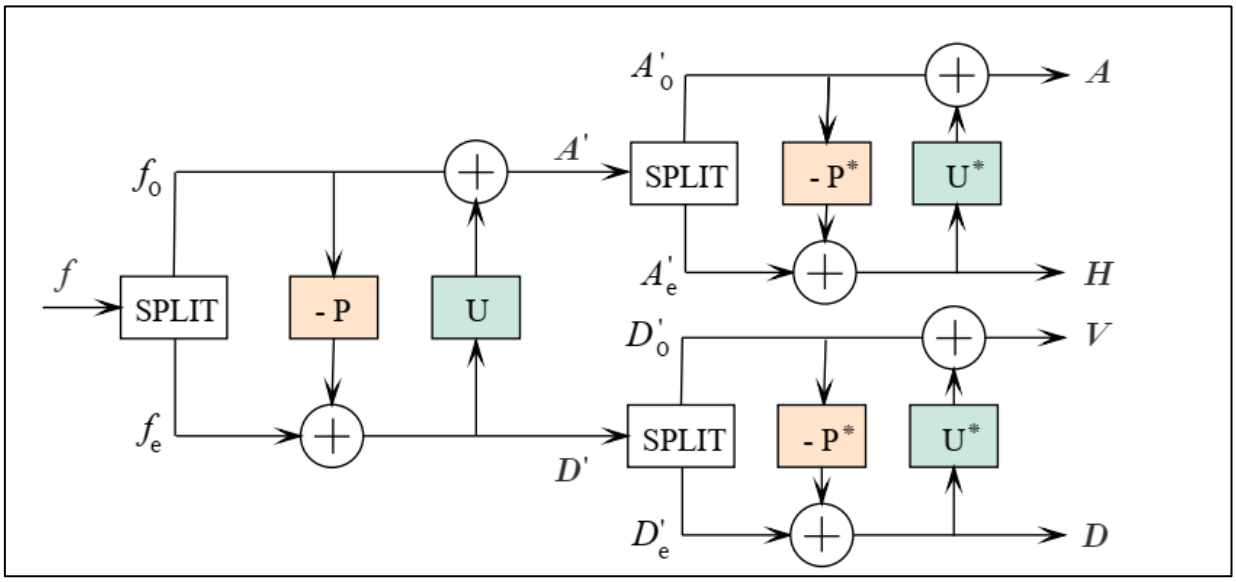



**Figure 2.5:** Block diagram of 2D separable lifting steps [96]

**1- Polyphase transform:** is a process that can be described in relatively simple terms. Given a signal as input, the transform separates it into two components, the even and odd components. To be more precise, the application of the polyphase transform to the original signal $f$ results in the partitioning of $f$ into two disjoint subsets: $f_e$, which contains the samples belonging to $f$ with an even index, and $f_0$, which contains the samples belonging to $f$ with an odd index. This process is referred to as the "Lazy Wavelet Transform".

Polyphase transform: $f \rightarrow (f_e, f_0)$

**2- Prediction operation:** is based on the finding that most imaging signals exhibit a local correlation structure in both the frequency and spatial domains. In other words, samples with highly correlated frequencies in the same neighborhood will show a high correlation rate, whereas samples with highly correlated frequencies outside of the same neighborhood will only show a low correlation rate. The two sub-signals, $f_e$ and $f_0$, will be highly correlated if the signal $f$ displays such a structure, making it possible to predict $f_0$ from $f_e$. To predict $f_0$, the operator P is applied to the subset $f_e$. The wavelet coefficients, or detail of the signal, are the difference $d$ between $f_0$ and its prediction.

$$d = f_0 - P(f_e)$$

The calculation of the prediction and storage of the details are performed through a dual lifting step. Therefore, the second step of the lifting scheme is expressed as follows:

$$\text{Dual lifting step: } (f_e, f_0) \rightarrow (f_e, d)$$

3- **Update operation:** is the final step in the process, and it is essential for acquiring a more precise low-frequency representation of the signal while preserving its fundamental global properties. Two subsets are identified: $f_e$ and de. The subset de is derived through a process analogous to high-pass filtering, followed by sub-sampling. The subset $f_e$ is obtained by straightforward sub-sampling, resulting in the signal's distribution across the entire frequency band of the original signal. It is important to note that Shannon's conditions are not met in this scenario, necessitating the implementation of a third step to ensure an adequate representation of the low-frequency signal. This third step involves the application of an update operator on the detailed signal [87].

$$s = f_e + U(d)$$

The second step is referred to as a lifting step, which involves the transformation of $(f_e, d)$ to $(s, d)$.

$$\text{Lifting step: } (f_e, d) \rightarrow (s, d).$$

### 2.4.1 Properties of the Lifting Scheme

**Faster calculation:** Lifting enables the execution of calculations within the same memory location as the input signal, thereby facilitating significant reductions in memory requirements.

**Efficiency:** In the majority of cases, the number of operations is reduced in comparison to the independent filtering of the two sub-bands. Indeed, lifting calculates the low and high frequency bands simultaneously.

**Reversible:** The implementation of the inverse lifting scheme is elementary. It is obtained by reversing the order of the operations and undoing them.

**General:** The transform is implemented without any reference to the Fourier domain. Furthermore, the extension of the scheme to other application frameworks is extremely easy. [90].

### 2.5 Conclusion

The wavelet transform is a tool that can provide a one-dimensional signal a multiresolution representation. Data compression and transmission, image and signal processing, and numerical solutions of differential and integral equations are just a few of the application areas in which it has seen significant success. The reason the wavelet transform works so well is because, for a certain class of functions, the majority of the function's information is concentrated in a limited number of wavelet coefficients.

In the context of multidimensional signals, such as images, the construction of wavelet bases that are separable by tensor product is a viable option. However, it should be noted that these bases possess an undeformable square support and are isotropic. Consequently, they are not optimally suited for the representation of regions in images that exhibit contours or local singularities. To address this limitation, numerous anisotropic wavelet bases, including curvelets, contourlets, bandelets, and oriented wavelets, have been proposed to facilitate a more efficient representation of images.

Meanwhile, the discovery of the lifting structure has made it easier to construct simple multiresolution transforms, which are invertible by nature and allow for the use of non-linear operators that can capture the singularities present in a signal. By minimizing the necessary arithmetic operations, the lifting scheme also improves the wavelet transform's efficiency. Using the lifting scheme, the direct and inverse transforms are implemented without having to go through the Fourier domain.

CHAPTER 3

# LIFTING WAVELET TRANSFORM-BASED PHOTOMETRIC VISUAL SERVOING

**Contents**

# CHAPTER 3. LIFTING WAVELET TRANSFORM-BASED PHOTOMETRIC VISUAL SERVOING

## 3.1 Introduction

In this chapter, an introduction is provided to the central focus of this thesis: *Lifting Wavelet Transform-based Photometric Visual Servoing* (LWT-PVS). A camera image provides a wealth of information about the world perceived by the robot, encoded in varying intensity levels, commonly referred to as luminance. The core idea behind LWT-PVS is to capture this rich information, encoded as pixel intensities, and transform it into useful features for visual servoing tasks. By leveraging the Lifting Wavelet Transform (LWT), multi-resolution features are extracted with the aim of guiding the robot's movements with high precision.

One of the key contributions of this chapter is the development of an interaction matrix for LWT-PVS, which is formulated in analytical form. This interaction matrix establishes the relationship between the changes in the visual features and the spatial velocity of the vision sensor, providing the necessary framework for the visual servoing task. It is demonstrated that the proposed method can perform visual servoing without relying on traditional image processing techniques, such as image matching and visual tracking. This makes the approach more efficient and less computationally intensive, while maintaining robustness and accuracy in various operational scenarios.

## 3.2 Interaction Matrix for Lifting Scheme Coefficient

The interaction matrix, denoted as $L$, is constructed based on the chosen visual features and is expressed within the camera reference frame. This matrix plays a fundamental role in establishing the relationship between the changes in visual features and the spatial velocity of the vision sensor, represented by $\boldsymbol{v_c}$. Essentially, it serves as the mathematical framework that links the sensor's motions to the observed feature variations in the camera's field of view. This relationship can be expressed in the following manner, as described in [2]:

$$s = L_{\mathbf{s}} v_c \tag{3.1}$$

For the purpose of the visual servoing task, the proposed visual features, denoted as $\boldsymbol{W}$, are derived by incorporating both the stacked approximation A and the detail components **H**, **V**, and $\boldsymbol{D}$. These components are obtained through the application of two-dimensional separable lifting operations performed on the image I($x$, $y$). In this context, ($x$, $y$) represent the normalized pixel coordinates within the image plane, which correspond to the three-dimensional spatial coordinates $\boldsymbol{X}$=*(X, Y, Z)* in the camera reference frame. The visual features effectively capture multi-resolution characteristics of the image and are mathematically expressed as follows:

$$x = \frac{X}{Z=\frac{(u-c_u)}{f\alpha}} \quad , \quad y = \frac{Y}{Z=\frac{(v-c_v)}{f}} \tag{3.2}$$

In this context, the image point coordinates $\boldsymbol{m} = (u; v)$ are expressed in pixel units and are influenced by the intrinsic parameters of the camera, collectively represented as $\boldsymbol{a} = (c_u, c_v, f, \alpha)$. Here, $c_u$ and $c_v$ denote the pixel coordinates of the camera's principal point, which is the point where the optical axis intersects the image plane. The focal length of the camera is given by $f$, while $\alpha$ represents the aspect ratio, which captures the relationship between the pixel dimensions in the horizontal and vertical directions. These intrinsic parameters form a key component of the camera model, enabling the mapping of three-dimensional scene points to their corresponding two-dimensional image points.

As shown in Fig. 2.5, and with assuming the function $f$ is the captured image $I$, **A**, **H**, **V** and **D** are expressed as:

$$H = A'_e - P^*(A'_o) \tag{3.3}$$

$$A = A'_o + U^*(H) \tag{3.4}$$

$$D = D'_e - P^*(D'_o) \tag{3.5}$$

$$V = D'_o + U^*(D) \tag{3.6}$$

where

$$I_o = I(x_o, y) \tag{3.7}$$

$$I_e = I(x_e, y) \tag{3.8}$$

$$A'_o = A'(x_e, y_o) \tag{3.9}$$

$$A'_e = A'(x_e, y_e) \tag{3.10}$$

$$D'_o = D'(x_o, y_o) \tag{3.11}$$

$$D'_e = D'(x_o, y_e) \tag{3.12}$$

Such that

$$D' = I_e - P(I_o) \tag{3.13}$$

$$A' = I_o + U(D') \tag{3.14}$$

Coordinates with odd and even indexes, $x_o$, $y_o$ and $x_e$, $y_e$, are $x$ and $y$ such that $u$ and $v$ in Equation (3.2) are odd or even.

The primary objective of this section is to derive the interaction matrix that establishes the relationship between the features obtained via the Lifting Scheme and the spatial velocity of the camera. By doing so, this matrix acts as a bridge connecting the variations in the lifted image features to the motion dynamics of the vision sensor, providing a vital foundation for tasks such as control and motion estimation in visual servoing applications.

$$W = L_{\mathbf{W}} v_c \tag{3.15}$$

where:

$$W = \begin{pmatrix} H \\ A \\ D \\ V \end{pmatrix} \tag{3.16}$$

and:

$$L_W = \begin{pmatrix} L_H \\ L_A \\ L_D \\ L_V \end{pmatrix} \tag{3.17}$$

with:

$$H = L_H v_c \tag{3.18}$$

$$A = L_{\mathrm{A}} v_c \tag{3.19}$$

$$D = L_D v_c \tag{3.20}$$

$$V = L_V v_c \tag{3.21}$$

The predictor P and updater U operators, along with their Hermitian transposes $P^*$ and $U^*$, can be formulated using convolution filters denoted as $p$, $u$, $p^*$, and $u^*$. These convolutional representations simplify the mathematical expressions and allow the corresponding Equations, specifically (3.3), (3.4), (3.5), (3.6), (3.13), and (3.14), to be rewritten in an equivalent, more compact form as follows:

$$H = A'_e - \iint A'_o(x, y, t)p^*(x - \tau_1, y - \tau_2)dxdy \tag{3.22}$$

$$A = A'_o + \iint H(x, y, t)u^*(x - \tau_1, y - \tau_2)dxdy \tag{3.23}$$

$$D = D'_e - \iint D'_o(x, y, t)p^*(x - \tau_1, y - \tau_2)dxdy \tag{3.24}$$

$$V = D'_o + \iint D(x, y, t)u^*(x - \tau_1, y - \tau_2)dxdy \tag{3.25}$$

$$D' = I_e - \iint I_o(x, y, t)p^*(x - \tau_1, y - \tau_2)dxdy \tag{3.26}$$

$$A' = I_o + \iint D'(x, y, t)u^*(x - \tau_1, y - \tau_2)dxdy \tag{3.27}$$

The derivative of approximations and details can be written as:

$$H = \frac{dA'_e\ x,y,t}{dt} - \iint \frac{dA'_o\ x,y,t\ p^*(x-\tau_1,y-\tau_2)}{dt}dxdy = \frac{dA'_e\ x,y,t}{dt} - \iint \frac{dA'_o\ x,y,t}{dt}p^*(x - \tau_1, y - \tau_2)dxdy \tag{3.28}$$

$$A = \frac{dA'_o\ x,y,t}{dt} + \iint \frac{dH\ x,y,t\ u^*(x-\tau_1,y-\tau_2)}{dt}dxdy = \frac{dA'_o\ x,y,t}{dt} - \iint \frac{dH\ x,y,t}{dt}u^*(x - \tau_1, y - \tau_2)dxdy \tag{3.29}$$

$$D = \frac{dD'_e\ x,y,t}{dt} - \iint \frac{dD'_o\ x,y,t\ p^*(x-\tau_1,y-\tau_2)}{dt}dxdy = \frac{dD'_o\ x,y,t}{dt} - \iint \frac{dD'_o\ x,y,t}{dt}p^*(x - \tau_1, y - \tau_2)dxdy \tag{3.30}$$

$$V = \frac{dD'_o\ x,y,t}{dt} + \iint \frac{dD\ x,y,t\ u^*(x-\tau_1,y-\tau_2)}{dt} dxdy = \frac{dD'_o\ x,y,t}{dt} + \iint \frac{dD\ x,y,t}{dt} u^*(x-\tau_1, y-\tau_2)dxdy \tag{3.31}$$

$$D' = \frac{dI_e\ x,y,t}{dt} - \iint \frac{dI_o\ x,y,t\ p(x-\tau_1,y-\tau_2)}{dt} dxdy = \frac{dI_e\ x,y,t}{dt} - \iint \frac{dI_o\ x,y,t}{dt} p(x-\tau_1, y-\tau_2)dxdy \tag{3.32}$$

$$A' = \frac{dI_o\ x,y,t}{dt} + \iint \frac{dD'\ x,y,t\ u(x-\tau_1,y-\tau_2)}{dt} dxdy = \frac{dI_o\ x,y,t}{dt} + \iint \frac{dD'\ x,y,t}{dt} u(x-\tau_1, y-\tau_2)dxdy \tag{3.33}$$

Relying on the temporal luminance constancy hypothesis, it is assumed that the intensity (or luminance) of a specific point **x** remains unchanged over time. This assumption proves to be applicable across various components, including $I_e$, $I_o$, $\boldsymbol{A'_e}, \boldsymbol{A'_o}, \boldsymbol{D'_e}, \boldsymbol{D'_o}$, $\mathbf{H}\ and\ \boldsymbol{D}$, i.e. these entities collectively represent the different luminance attributes or transform components derived from the image, and the constancy hypothesis provides the foundation for modeling their temporal behavior, as indicated in the following formulation:

$$I(\mathbf{x} + d\mathbf{x}; t + dt) = I(\mathbf{x}; t) \tag{3.34}$$

Here, ***dx*** represents the displacement of the point x occurring during the time interval ***dt***. By performing a first-order Taylor series expansion of Equation (4.34) around the point $\mathbf{x}$, it is possible to derive the optical flow constraint equation. This equation establishes a linear approximation relating the temporal and spatial variations of intensity, laying the groundwork for computing the motion field.

$$\nabla I^T \mathbf{x} + I_t = 0 \tag{3.35}$$

where $\nabla I^T = \left[\frac{dI}{dx} \quad \frac{dI}{dy}\right] = \begin{bmatrix} I_x & I_y \end{bmatrix}$ is the spatial gradient of the image I(**x**,t) at point **x**.
Thus:

$$I = -\nabla I^T \mathbf{x} \tag{3.36}$$

Taking into account the interaction matrix $L_{\mathbf{x}}$ associated with ***x*** (i.e. $\dot{\mathbf{x}} = L_{\mathbf{x}} \boldsymbol{v_c}$):

$$L_{\mathbf{x}} = \begin{pmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -1+x^2 & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{pmatrix} \tag{3.37}$$

By assuming a planar scene, a relationship connecting the depth of scene points to the coordinates of their corresponding image points can be established as follows:

$$\frac{1}{Z} = Ax + By + C \tag{3.38}$$

Relation (3.35) gives:

$$I = -\nabla I^T L_{\mathbf{x}} v_c \tag{3.39}$$

As a result, the interaction matrix corresponding to image intensities can be expressed as:

$$L_I = -\nabla I^T L_{\mathbf{x}} \tag{3.40}$$

where $L_I$ is given by:

$$L_I^T = \begin{pmatrix} I_x(Ax+By+C) \\ I_y(Ax+By+C) \\ (-xI_x - yI_y)(Ax+By+C) \\ -xyI_x - (1+y^2)I_y \\ 1+x^2\ I_x + xyI_y \\ -yI_x + xI_y \end{pmatrix} \tag{3.41}$$

Thus (3.28), (3.29), (3.30) and (3.31) become:

$$\dot{H} = [L_{A'_e} - \iint L_{A'_o} p^*(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.42}$$

$$\dot{A} = [L_{A'_o} + \iint L_D u^*(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.43}$$

$$\dot{D} = [L_{D'_e} - \iint L_{D'_o} p^*(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.44}$$

$$\dot{V} = [L_{D'_o} + \iint L_D u^*(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.45}$$

with:

$$\dot{D}' = [L_{I_e} - \iint L_{I_o} p(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.46}$$

$$\dot{A}' = [L_{I_o} + \iint L_{D'} u(x-\tau_1, y-\tau_2)dxdy]\, v_c \tag{3.47}$$

So we get:

$$L_H = [L_{A'_e} - \iint [L_{A'_o} p^*(x-\tau_1, y-\tau_2)dxdy] \tag{3.48}$$

$$L_A = [L_{A'_o} + \iint L_D u^*(x-\tau_1, y-\tau_2)dxdy] \tag{3.49}$$

$$L_D = [L_{D'_e} - \iint L_{D'_o} p^*(x-\tau_1, y-\tau_2)dxdy] \tag{3.50}$$

$$L_V = [L_{D'_o} + \iint L_D u^*(x-\tau_1, y-\tau_2)dxdy] \tag{3.51}$$

After substitution of (3.9) into (3.11), the following result is obtained:

$$L_H = \begin{pmatrix} L_H^{v_x} & L_H^{v_y} & L_H^{v_z} & L_H^{w_x} & L_H^{w_y} & L_H^{w_z} \end{pmatrix} \tag{3.52}$$

$$L_A = \begin{pmatrix} L_A^{v_x} & L_A^{v_y} & L_A^{v_z} & L_A^{w_x} & L_A^{w_y} & L_A^{w_z} \end{pmatrix} \tag{3.53}$$

$$L_D = \begin{pmatrix} L_D^{v_x} & L_D^{v_y} & L_D^{v_z} & L_D^{w_x} & L_D^{w_y} & L_D^{w_z} \end{pmatrix} \tag{3.54}$$

$$L_V = \begin{pmatrix} L_V^{v_x} & L_V^{v_y} & L_V^{v_z} & L_V^{w_x} & L_V^{w_y} & L_V^{w_z} \end{pmatrix} \tag{3.55}$$

With setting $Ax + By + C = \alpha$ we get:

$$\begin{cases} L_H^{v_x} = \alpha I_{A'_{e_x}} - \iint \alpha I_{A'_{o_x}} p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_H^{v_y} = \alpha I_{A'_{e_y}} - \iint \alpha I_{A'_{o_y}} p^*\ x-\tau_1, y_o-\tau_2\ dxdy \\ L_H^{v_z} = \alpha\left(-xI_{A'_{e_x}} - yI_{A'_{e_y}}\right) - \iint \alpha\left(-xI_{A'_{o_x}} - yI_{A'_{o_y}}\right) p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_H^{w_x} = -xyI_{A'_{e_x}} - \left(1+y^2\right) I_{A'_{e_y}} - \iint \left[-xyI_{A'_{o_x}} - \left(1+y^2\right) I_{A'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_H^{w_y} = \left(1+x^2\right) I_{A'_{e_x}} + xyI_{A'_{e_y}} - \iint \left[\left(1+x^2\right) I_{A'_{o_x}} + xyI_{A'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_H^{w_z} = -yI_{A'_{e_x}} + xI_{A'_{e_y}} - \iint \left[-yI_{A'_{o_x}} + xI_{A'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \end{cases} \tag{3.56}$$

$$\begin{cases} L_A^{v_x} = \alpha I_{A'_{o_x}} + \iint \alpha I_{H_x} u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_A^{v_y} = \alpha I_{A'_{o_y}} + \iint \alpha I_{H_y} u^*\ x-\tau_1, y_o-\tau_2\ dxdy \\ L_A^{v_z} = \alpha\left(-xI_{A'_{o_x}} - yI_{A'_{o_y}}\right) + \iint \alpha\left(-xI_{H_x} - yI_{H_y}\right) u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_A^{w_x} = -xyI_{A'_{o_x}} - \left(1+y^2\right) I_{A'_{o_y}} + \iint \left[-xyI_{H_x} - \left(1+y^2\right) I_{H_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_A^{w_y} = \left(1+x^2\right) I_{A'_{o_x}} + xyI_{A'_{o_y}} + \iint \left[\left(1+x^2\right) I_{H_x} + xyI_{H_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_A^{w_z} = -yI_{A'_{o_x}} + xI_{A'_{o_y}} + \iint \left[-yI_{H_x} + xI_{H_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \end{cases} \tag{3.57}$$

$$\begin{cases} L_D^{v_x} = \alpha I_{D'_{e_x}} - \iint \alpha I_{D'_{o_x}} p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_D^{v_y} = \alpha I_{D'_{e_y}} - \iint \alpha I_{D'_{o_y}} p^*\ x-\tau_1, y_o-\tau_2\ dxdy \\ L_D^{v_z} = \alpha\left(-xI_{D'_{e_x}} - yI_{D'_{e_y}}\right) - \iint \alpha\left(-xI_{D'_{o_x}} - yI_{D'_{o_y}}\right) p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_D^{w_x} = -xyI_{D'_{e_x}} - \left(1+y^2\right) I_{D'_{e_y}} - \iint \left[-xyI_{D'_{o_x}} - \left(1+y^2\right) I_{D'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_D^{w_y} = \left(1+x^2\right) I_{D'_{e_x}} + xyI_{D'_{e_y}} - \iint \left[\left(1+x^2\right) I_{D'_{o_x}} + xyI_{D'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_D^{w_z} = -yI_{D'_{e_x}} + xI_{D'_{e_y}} - \iint \left[-yI_{D'_{o_x}} + xI_{D'_{o_y}}\right] p^*\ x-\tau_1, y-\tau_2\ dxdy \end{cases} \tag{3.58}$$

$$\begin{cases} L_V^{v_x} = \alpha I_{D'_{o_x}} + \iint \alpha I_{D_x} u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_V^{v_y} = \alpha I_{D'_{o_y}} + \iint \alpha I_{D_y} u^*\ x-\tau_1, y_o-\tau_2\ dxdy \\ L_V^{v_z} = \alpha\left(-xI_{D'_{o_x}} - yI_{D'_{o_y}}\right) + \iint \alpha\left(-xI_{D_x} - yI_{D_y}\right) u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_V^{w_x} = -xyI_{D'_{o_x}} - \ 1+y^2\ I_{D'_{o_y}} + \iint \left[-xyI_{D_x} - \ 1+y^2\ I_{D_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_V^{w_y} = \ 1+x^2\ I_{D'_{o_x}} + xyI_{D'_{o_y}} + \iint \left[\ 1+x^2\ I_{D_x} + xyI_{D_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \\ L_V^{w_z} = -yI_{D'_{o_x}} + xI_{D'_{o_y}} + \iint \left[-yI_{D_x} + xI_{D_y}\right] u^*\ x-\tau_1, y-\tau_2\ dxdy \end{cases} \tag{3.59}$$

### 3.3 Control Scheme

In this section, the methodological considerations underlying the selection of filters for the Lifting scheme are presented. The predict and update filters are accordingly selected to be Lazy wavelets, as detailed below:

$$u = [1/2 \quad 1/2], \qquad u^* = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} \tag{3.60}$$

$$p = [1/4 \quad 1/4], \qquad p^* = \begin{pmatrix} 1/4 \\ 1/4 \end{pmatrix} \tag{3.61}$$

The control strategy applied to the chosen features is the same as presented in [2]:

$$v_c = -\lambda L_W^+ (W - W^*) \tag{3.62}$$

Here, $\lambda$>0 represents the control gain, and $L_W^+$ denotes the Moore-Penrose pseudo inverse of the interaction matrix $L_W$.

Object and image planes are assumed to be parallel, which means $A = B = 0$ [2], and the depth $Z$ is held at the desired position $Z^*$. So (3.6) becomes:

$$C = \frac{1}{Z^*} \tag{3.63}$$

The computation of the visual servoing control law can be formulated as an optimization problem aimed at minimizing a visual error function in the image space. Several numerical optimization strategies have been employed in the visual servoing literature, each presenting specific advantages and limitations.

First-order gradient-based methods, such as steepest descent, are simple and computationally efficient but often suffer from slow convergence and sensitivity to step-size tuning, particularly in nonlinear visual servoing problems [2]. Gauss–Newton methods improve convergence speed by exploiting a local quadratic approximation of the cost function; however, their effectiveness strongly depends on the validity of the linearization and may degrade in the presence of large initial errors or poorly conditioned interaction matrices [74]. Full Newton-based methods, while theoretically attractive, are generally impractical for real-time visual servoing due to the computational burden associated with Hessian estimation [80].

To overcome these limitations, this work adopts the Levenberg–Marquardt (LM) approach, which combines the robustness of gradient descent with the fast local convergence of Gauss–Newton optimization [60]. By introducing an adaptive damping factor, the LM method ensures stable convergence when the system is far from the solution and progressively accelerates convergence as the error decreases. This property is particularly important in visual servoing, where nonlinearities and variations in the interaction matrix are common.

The Levenberg–Marquardt strategy therefore provides a suitable compromise between numerical stability, convergence speed, and computational efficiency, making it well adapted to the visual servoing framework considered in this thesis.

Consequently, Equation (3.13) is reformulated as follows:

$$v_c = -\lambda(H + \mu diag(H))^{-1} L_W^T (W - W^*) \tag{3.64}$$

Here, $H = L_G^T L_G$ serves as an approximation of the Hessian matrix, and μ represents a damping factor. To enhance the convergence characteristics of the control law, the parameters $\lambda$ and $\mu$ in the *Levenberg-Marquardt* optimization algorithm are made adaptive, following the approach described in [27]:

$$\lambda = 10^{log_{10}\ Ne\ -7.5} \tag{3.65}$$

$$\mu = 10^{2log_{10}\ Ne\ -21} \tag{3.66}$$

Here, $N_e$ represents the norm of the difference between the current image and the target image. The values 7.5 and 21 were determined empirically.

### 3.4 Conclusion

In conclusion, this chapter introduces a novel approach to visual servoing by incorporating multi-resolution wavelet features derived from the lifting scheme. The primary contribution is the formulation of an interaction matrix that connects the variations in image features with the spatial velocities of the camera. This matrix serves as the mathematical backbone for relating visual feature dynamics to sensor motion, enabling precise control in 6-dof positioning tasks. The visual features are derived from stacked approximations and detail components, which are obtained through two-dimensional separable lifting operations on the image. By utilizing this multi-scale representation, the image's rich characteristics are effectively captured, thereby enhancing the system's ability to perform visual servoing tasks.

The proposed method integrates Lazy/Haar wavelets as the prediction and update filters, and employs the Levenberg-Marquardt optimization technique to ensure robust convergence. This optimization approach is preferred over traditional methods, as it expands the convergence domain and improves the efficiency of the control law. The interaction matrix is further enhanced through the use of Moore-Penrose pseudoinversion, allowing for accurate estimation of the motion required to bring the visual features into alignment with the desired target.

The framework presented here eliminates the need for complex image processing techniques, such as image matching and visual tracking, which are typically computationally expensive and prone to errors under variable conditions. Instead, the use of lifted wavelet coefficients provides a more efficient and stable solution for visual servoing applications. The adaptive nature of the control parameters, $\lambda$ and $\mu$, ensures that the system can handle varying conditions in real-time, making the approach both effective and versatile.

CHAPTER 4

# SIMULATION RESULTS

**Contents**

## 4.1 Introduction

This chapter presents a comprehensive simulation-based evaluation of photometric visual servoing strategies, with particular emphasis on the robustness and stability of the control loop under varying illumination conditions and different initial camera configurations. The study focuses on the comparison between classical PVS and the proposed LWT-PVS.

Photometric visual servoing directly exploits image intensity information as visual features, thereby avoiding explicit geometric modeling and feature extraction. While this dense representation provides rich visual feedback, it is well known to be highly sensitive to photometric disturbances such as illumination variations, shadows, and global intensity changes. These limitations often restrict the practical applicability of classical PVS in real-world environments.

The approach proposed in this thesis aims to address these limitations by introducing a wavelet-domain representation within the photometric visual servoing framework. By decomposing image intensities using the lifting wavelet transform, the proposed method separates structural information from illumination-dependent components, thereby improving robustness while preserving the core advantages of photometric control.

The objective of this chapter is to quantitatively and qualitatively evaluate the impact of this representation on closed-loop behavior. The analysis is conducted through a systematic set of simulations organized into two distinct scenarios, corresponding to moderate and large initial camera displacements. For each scenario, both PVS and LWT-PVS are evaluated under nominal illumination conditions and under deliberately introduced illumination variations. This experimental design allows a clear assessment of convergence properties, stability margins, and robustness to photometric disturbances.

The results are analyzed in terms of photometric error evolution, camera velocity profiles, and overall convergence behavior. Particular attention is devoted to the smoothness and boundedness of the control signals, which are critical for safe and reliable robotic operation. The simulation results presented in this chapter are intended to validate the theoretical developments introduced in the previous chapters and to provide strong motivation for the experimental validation presented in the subsequent chapter.

The remainder of this chapter is organized as follows. Section 4.2 describes the simulation framework and implementation details. Sections 4.3.1 and 4.3.2 present and analyze the results obtained for the two scenarios under consideration. Finally, Section 4.4 provides a comparative discussion and concludes the chapter.

## 4.2 Implementation Details and Simulation Configuration

This section describes in detail the simulation environment, software framework, and implementation parameters used to evaluate classical Photometric Visual Servoing (PVS) and the proposed Lifting Wavelet Transform–based Photometric Visual Servoing (LWT-PVS). All simulations are performed under strictly identical conditions to ensure a fair and reproducible comparison.

### Simulation Environment and Software Framework

All simulations are conducted using the ViSP (Visual Servoing Platform) library [82], which provides a dedicated and well-established framework for the implementation and evaluation of visual servoing algorithms. ViSP is widely used in the visual servoing community and offers native support for photometric visual servoing, interaction matrix computation, and camera motion simulation.

The simulations follow a kinematic eye-in-hand configuration, where a monocular camera is rigidly attached to the robot end-effector. Robot motion is simulated kinematically using camera velocity commands, without considering joint-level dynamics, which is consistent with standard visual servoing studies.

The same ViSP-based simulation pipeline is used for both PVS and LWT-PVS to guarantee identical numerical conditions, control loop timing, and integration schemes.

### Camera Model and Image Acquisition

The camera is modeled using a pinhole projection model with known intrinsic parameters. Lens distortion is neglected. The camera frame is denoted by $\mathcal{F}_c$.

Images are acquired in grayscale format with a fixed resolution of $240 \times 320$ pixels. This image size is used consistently throughout all simulations and corresponds to the resolution employed in the implementation of photometric visual servoing in ViSP.

The reference image $\mathbf{I}^*$is captured at the desired camera pose and remains fixed during the servoing process. At each iteration $k$, the current image $\mathbf{I}(k)$is generated by the simulator according to the current camera pose.

### Visual Features and Photometric Error Definition

For classical Photometric Visual Servoing, the visual feature vector is defined as the stacked vector of pixel intensity values:

$$\mathbf{s}(k) = \mathbf{I}(k),$$

and the corresponding photometric error is defined as:

$$\mathbf{e}(k) = \mathbf{I}(k) - \mathbf{I}^*.$$

This formulation directly follows the photometric visual servoing framework introduced by Collewet and Marchand [12].

For the proposed LWT-PVS approach, both the current and reference images are decomposed using the lifting wavelet transform described in Chapter 3. The wavelet decomposition is applied to the entire image at a fixed number of decomposition levels. The resulting approximation and detail coefficients are concatenated to form the visual feature vector:

$$\mathbf{s}_{\mathrm{LWT}}(k) = \mathcal{W}(\mathbf{I}(k)),$$

and the photometric error is defined in the wavelet domain as:

$$\mathbf{e}_{\mathrm{LWT}}(k) = \mathcal{W}(\mathbf{I}(k)) - \mathcal{W}(\mathbf{I}^*).$$

This representation preserves the dense photometric nature of the control law while introducing multi-resolution information.

**Control Law and Motion Representation**

For both PVS and LWT-PVS, the camera velocity screw is computed using the same velocity-based control law:

$$\mathbf{v}_c = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = -\lambda \mathbf{L}^{+} \mathbf{e},$$

Where $\mathbf{v} = (v_x, v_y, v_z)^\top$ denotes translational velocities and $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^\top$ denotes rotational velocities.

The camera pose error is characterized for analysis purposes by translational displacement $\mathbf{t} = (t_x, t_y, t_z)$, and rotational displacement $\boldsymbol{\theta} = (\theta_x, \theta_y, \theta_z)$.

These quantities are not used in the control law but are monitored to evaluate convergence behavior.

**Discrete-Time Control Loop**

The visual servoing loop is implemented in discrete time with a fixed sampling period $T_s$. At each iteration:

1. Image acquisition from the ViSP simulator
2. Computation of visual features and photometric error
3. Evaluation of the interaction matrix
4. Computation of the camera velocity
5. Camera pose update using first-order integration

This loop is repeated until convergence. Both PVS and LWT-PVS share the same discrete-time control architecture implemented in the ViSP simulator. The control loop, illustrated in Fig. 4.1, differs only in the definition of the visual features and the associated interaction matrix.

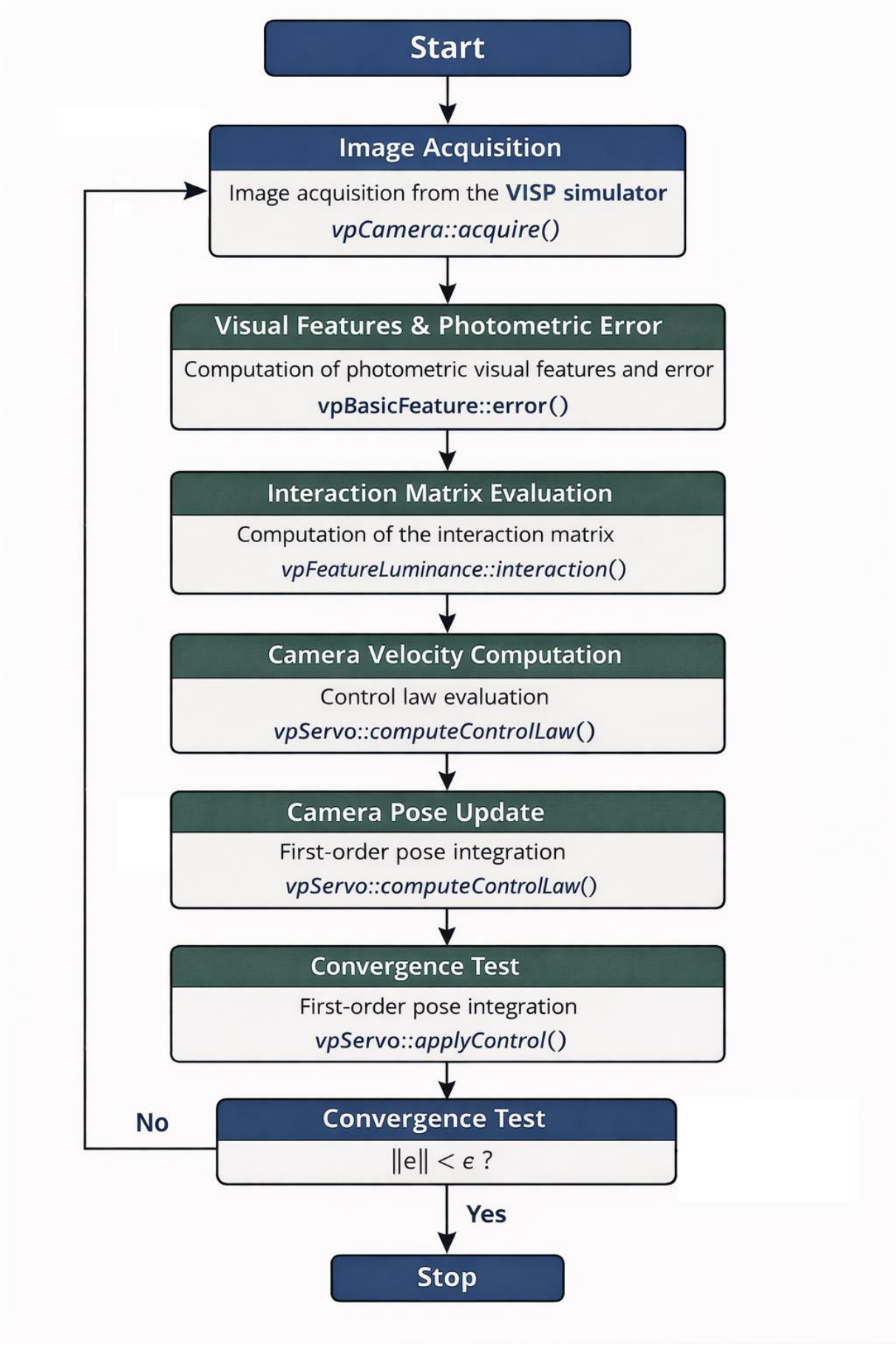



**Figure 4.1:** Discrete-time control loop for PVS and LWT-PVS implemented using the ViSP framework

**Illumination Variation Modeling**

Illumination variations are introduced by modifying the intensity distribution of the simulated images during the servoing process. Both global illumination changes and spatial intensity variations are applied, thereby violating the brightness constancy assumption.

The same illumination perturbations are applied to both PVS and LWT-PVS to ensure a fair robustness comparison.

**Stopping Criteria**

The servoing process is stopped when one of the following conditions is met:

- the norm of the photometric error falls below a predefined threshold, or
- a maximum number of iterations is reached.

The stopping criteria are identical for all experiments.

**Summary of the Simulation Protocol**

The implementation described above is used consistently for all experiments presented in this chapter. Only the initial camera pose—defined by translational and rotational displacements—and the illumination conditions are varied to define the two scenarios analyzed in Sections 4.3.1 and 4.3.2.

The resulting evolutions of the photometric error, camera velocity components, and convergence behavior are presented in Fig. 4.3 to Fig. 4.6 and Fig. 4.8 to Fig. 4.11, and analyzed in detail in the following sections.

## 4.3 Simulation Results

### 4.3.1 Scenario 1

This first scenario is designed to evaluate the behavior of photometric visual servoing schemes under moderate initial camera displacement with respect to the desired pose. The initial configuration is deliberately chosen to place the camera sufficiently far from the target pose to activate the nonlinear characteristics of the control law, while remaining within a realistic operational range for practical robotic applications. The initial pose error is defined by a translational displacement $\Delta\mathbf{T}_1 = [3\,\text{cm}, 3\,\text{cm}, 10\,\text{cm}]$ and a rotational displacement $\Delta\mathbf{R}_1 = [-30°, -25°, 20°]$.

Such a configuration ensures that all six degrees of freedom are simultaneously excited, allowing the evaluation of coupling effects between translational and rotational motions. At the same time, the magnitude of the displacement remains representative of typical industrial or laboratory conditions in which visual servoing is activated after a coarse positioning stage.

The objective of this scenario is twofold. First, it serves as a baseline configuration for assessing the nominal convergence properties of classical PVS and the proposed LWT-PVS approach under controlled conditions. Second, it enables a detailed analysis of the transient behavior of the control loop, including error oscillations, coupling effects between motion components, numerical conditioning of the interaction matrix, and the smoothness of the generated control signals.

By considering both normal illumination conditions and artificial illumination variations, this scenario highlights the intrinsic sensitivity of photometric control laws to photometric disturbances, even when geometric displacements remain moderate. The results obtained in this scenario therefore provide valuable insight into the robustness of each method and establish a meaningful reference point for the more challenging conditions investigated in Scenario 2.

The figures associated with Scenario 1 (Figs. 4.3(a–c) to 4.6(a–c)) illustrate the evolution of the photometric error, camera velocities, and convergence behavior over time. These results allow a direct and quantitative comparison between classical PVS and LWT-PVS in terms of stability margins, convergence speed, transient smoothness, and robustness to illumination changes.

#### 4.3.1.1 Normal Illumination

The results obtained for classical PVS under normal illumination conditions in Scenario 1 are presented in Fig. 4.3(a–c). The norm of the photometric error exhibits a global decreasing trend, confirming convergence toward the desired visual configuration. However, the transient phase is characterized by noticeable oscillations in the error evolution, particularly during the early iterations of the servoing process.

These oscillations can be attributed to the nonlinear nature of the photometric interaction matrix and to the sensitivity of raw intensity-based features to local intensity variations. In this operating regime, small local discrepancies in image intensity can generate corrective actions that propagate through the control law. The corresponding camera velocity profiles display irregular fluctuations, especially at the beginning of the servoing process, reflecting these corrective actions induced by local photometric inconsistencies.

Although convergence is ultimately achieved, the system operates close to its stability limits, and the lack of smoothness in the control signals indicates limited robustness, even under nominal illumination conditions. Such behavior may lead to undesirable mechanical stress or reduced positioning accuracy in real robotic systems.

The corresponding results obtained with LWT-PVS are shown in Fig. 4.4(a–c). In contrast to classical PVS, the pose errors converge more smoothly, with significantly reduced

oscillations throughout the servoing process. Both translational and rotational components exhibit a near-monotonic decay toward zero, indicating improved stability and damping characteristics.

Moreover, the camera velocity profiles are smoother and present lower peak magnitudes, reflecting improved numerical conditioning of the interaction matrix. The norm of the visual error decreases faster and more regularly, demonstrating enhanced convergence behavior. These observations indicate that, even under nominal illumination conditions, the wavelet-based representation provides greater robustness and improved transient behavior compared to classical photometric visual servoing.

#### 4.3.1.2 Illumination Variations

When illumination variations are introduced, the limitations of classical PVS become more pronounced, as illustrated in Fig. 4.5(a–c). The photometric error no longer decreases monotonically and may temporarily increase, indicating a violation of the brightness constancy assumption underlying the photometric control formulation. As a consequence, the camera velocity commands become irregular and may exhibit abrupt variations, reflecting instability in the control loop.

These behaviors confirm the strong sensitivity of classical photometric visual servoing to illumination disturbances, even in scenarios where geometric displacement remains moderate. In such conditions, the control law struggles to maintain consistent convergence, and the interaction matrix becomes poorly conditioned due to photometric inconsistencies.

The results obtained with LWT-PVS under the same illumination variations are presented in Fig. 4.6(a–c). Despite the degraded visual conditions, the pose errors maintain stable convergence trends. Although the convergence rate is slightly reduced compared to nominal illumination, no divergence or sustained oscillatory behavior is observed.

The velocity profiles remain smooth and bounded, and the norm of the visual error decreases steadily throughout the servoing process. This behavior confirms that the wavelet-based visual features effectively attenuate illumination-induced disturbances while preserving the structural information required for control.

Notably, the system achieves a high level of precision, with final translational errors as small as $[-0.069\,\text{mm},\ -0.112\,\text{mm},\ -0.186\,\text{mm}]$ and rotational errors reaching values as low as $[-1.119°,\ -0.274°,\ 0.011°]$.

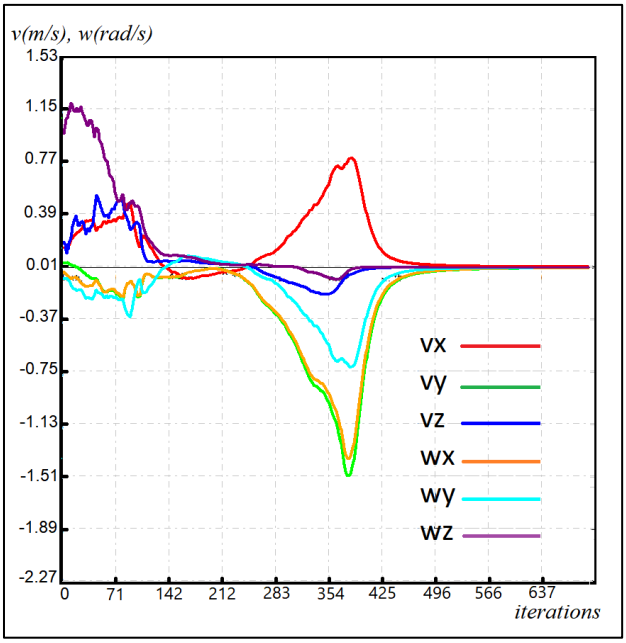
These results clearly demonstrate the ability of the proposed approach to maintain accurate alignment despite significant illumination perturbations.

(a) Initial image I

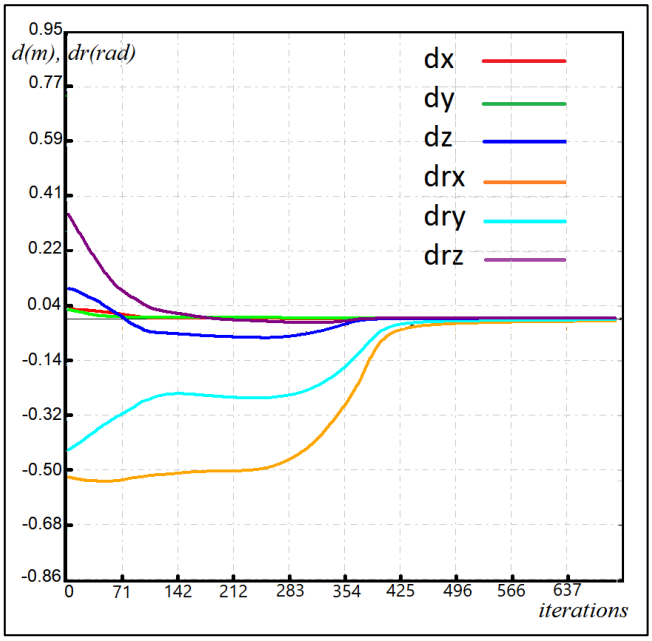
(b) Desired image $I^*$

(c) Error between desired images in normal conditions and with illumination variations $(I - I^* + 255)/2$ (used for visualization)

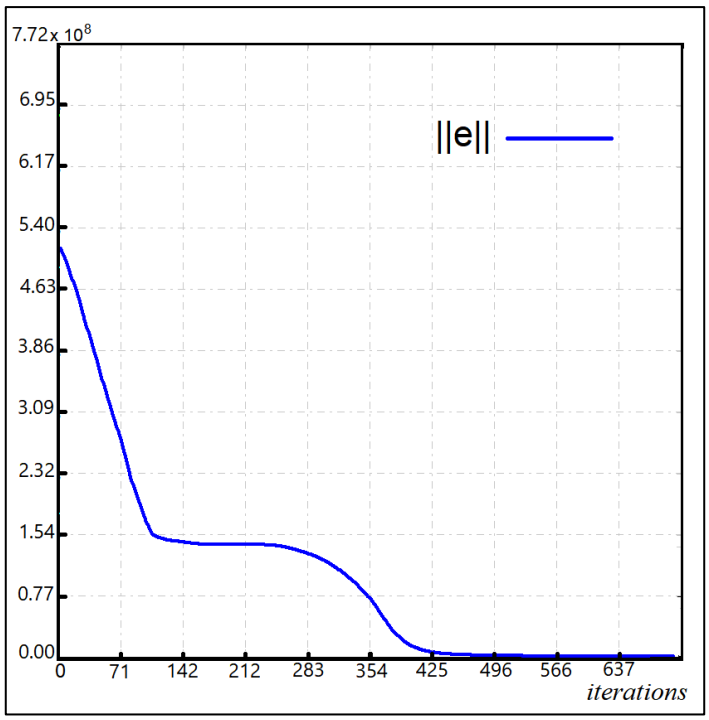
**Figure 4.2:** Simulation Images for scenario 1




(a) Robot velocities applied

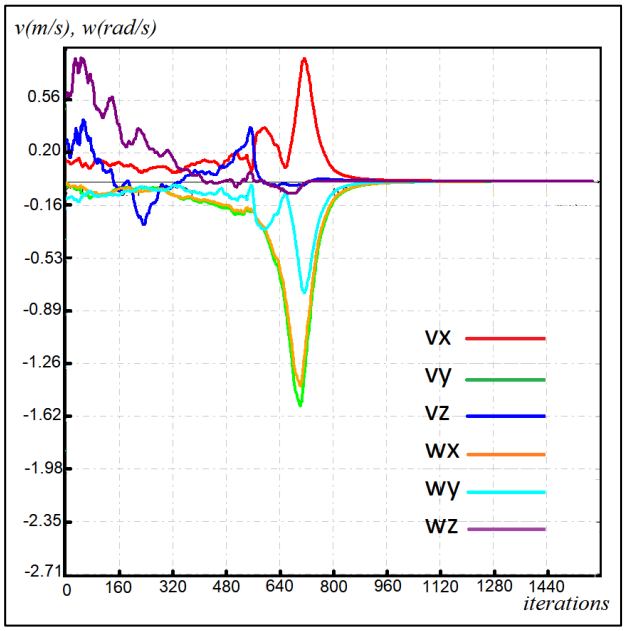



(b) Errors in positioning (in m and rad)

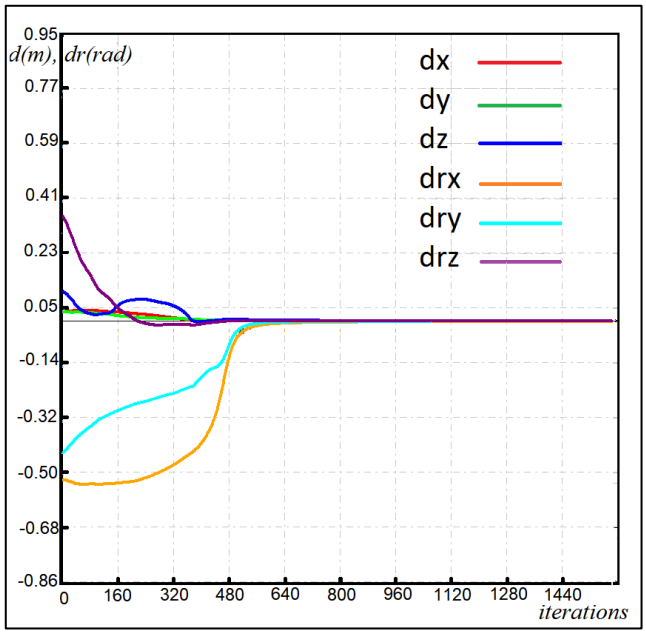



(c) $\|s - s^*\|$

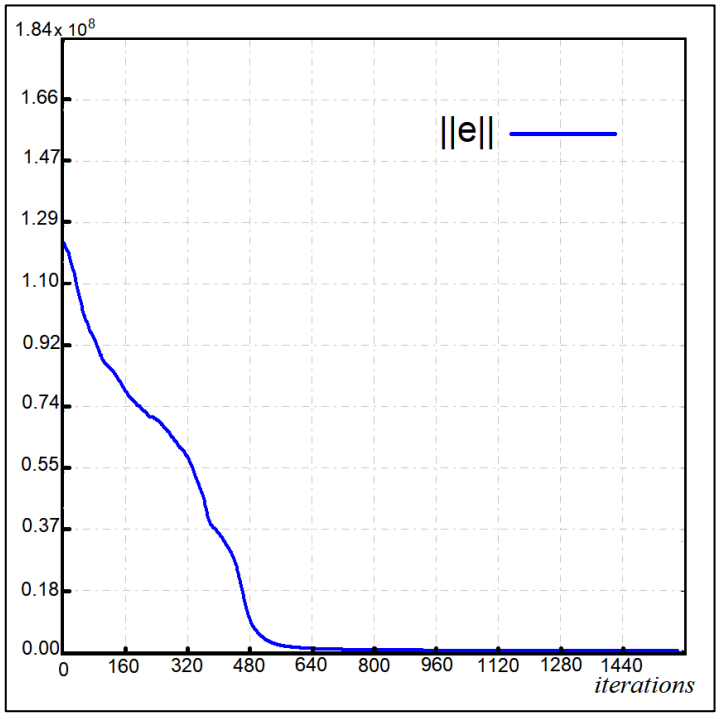
**Figure 4.3:** Simulation Results of **PVS** for scenario 1 in Normal Illumination




(a) Robot velocities applied

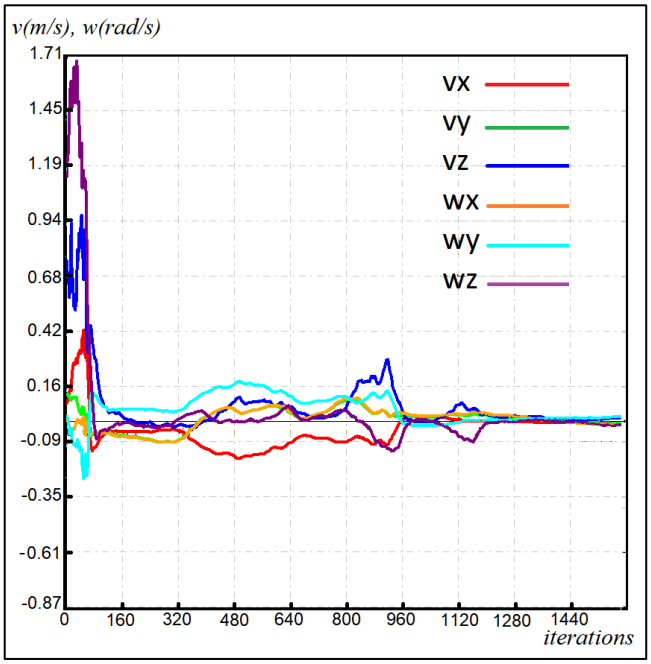



(b) Errors in positioning (in m and rad)

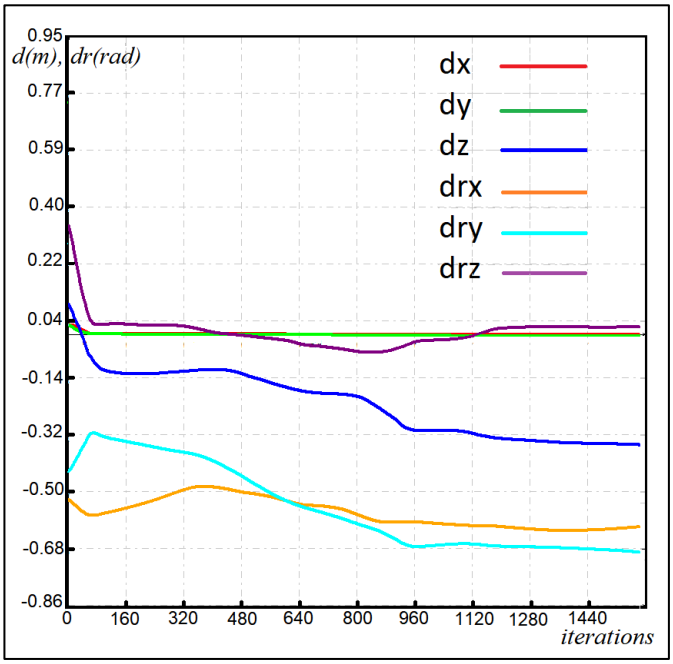



(c) $\|s - s^*\|$

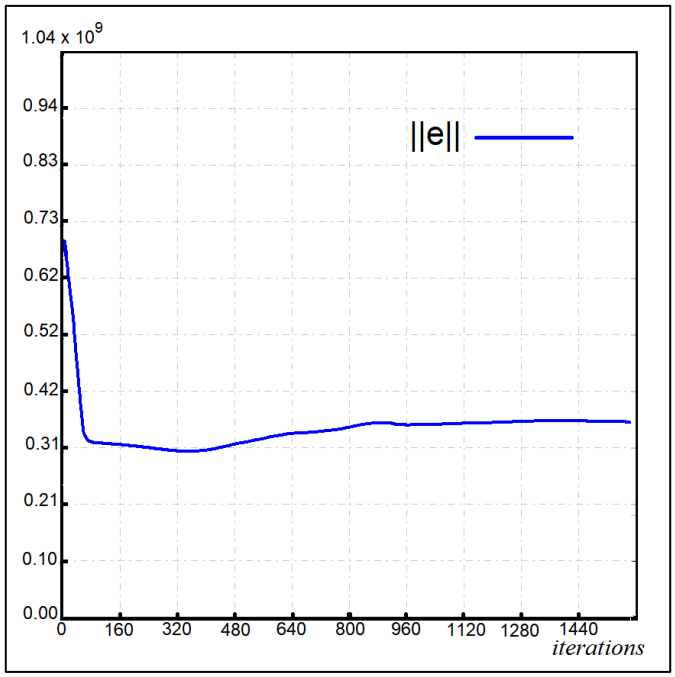
**Figure 4.4:** Simulation Results of **LWT-PVS** for scenario 1 in Normal Illumination




(a) Robot velocities applied

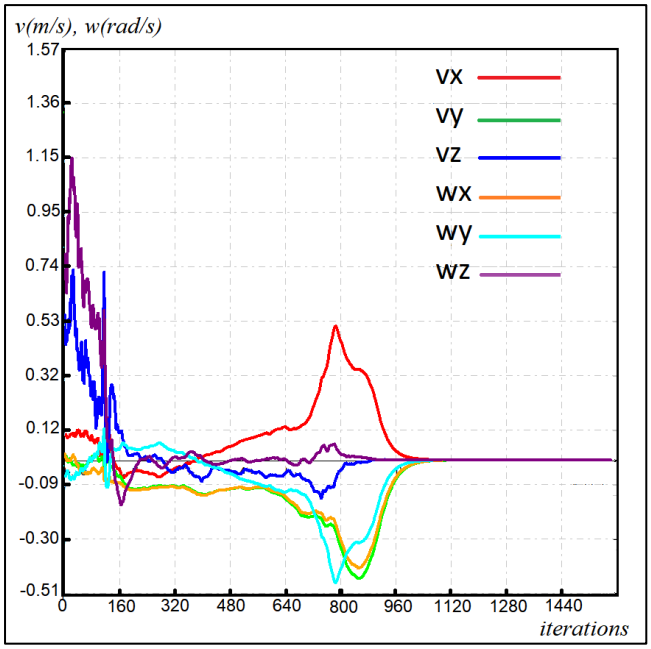



(b) Errors in positioning (in m and rad)

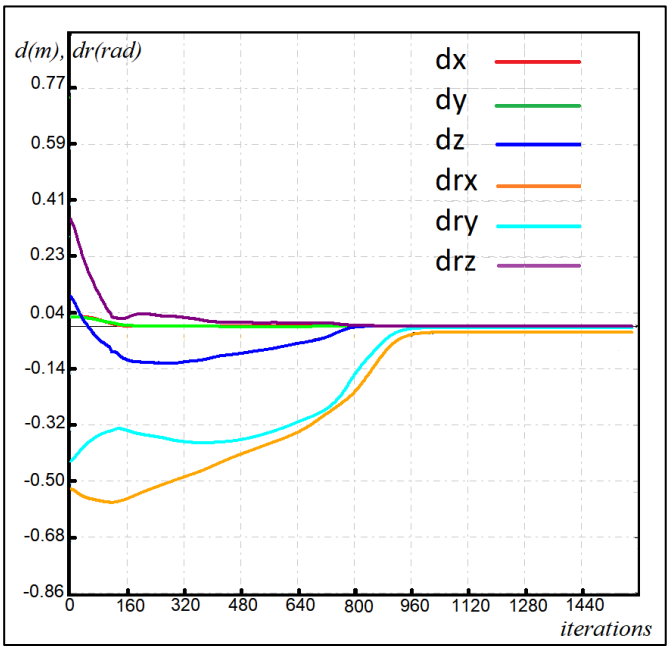



(c) $\|s - s^*\|$

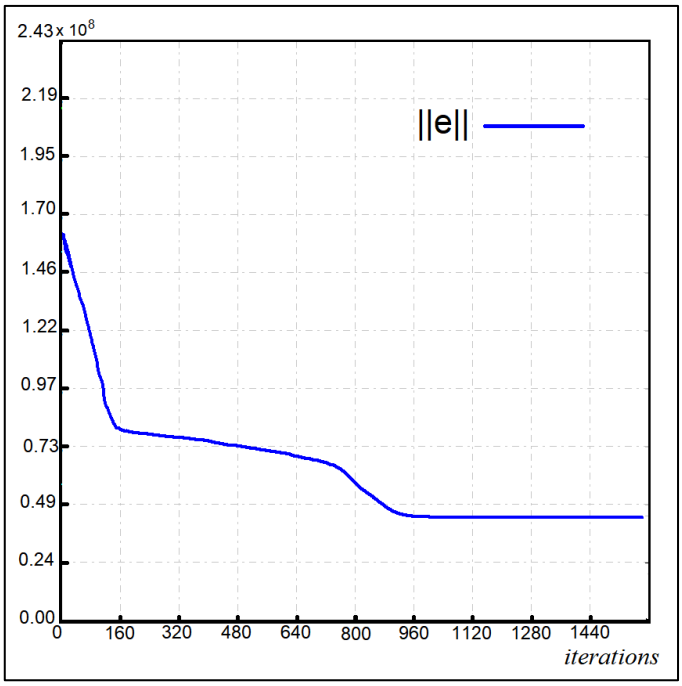
**Figure 4.5:** Simulation Results of **PVS** for scenario 1 in Illumination Variations




(a) Robot velocities applied




(b) Errors in positioning (in m and rad)




(c) $\|s - s^*\|$

**Figure 4.6:** Simulation Results of **LWT-PVS** for scenario 1 in Illumination Variations

### 4.3.2 Scenario 2

The second scenario addresses a significantly more challenging configuration, characterized by a large initial camera displacement involving substantial translational and rotational offsets relative to the desired pose. This scenario is intentionally designed to stress the photometric visual servoing framework and to evaluate the limits of stability, convergence, and robustness of both classical PVS and the proposed LWT-PVS approach.

Large initial displacements are known to exacerbate nonlinearities in the interaction matrix and to amplify sensitivity to photometric inconsistencies. As a result, this scenario constitutes a stringent test of the domain of attraction and the robustness margins of the control schemes under study. The initial pose error is defined by a rotational displacement $\Delta\mathbf{R}_2 = [-40°, 35°, 30°]$ and a translational displacement $\Delta\mathbf{T}_2 = [5\text{ cm}, -5\text{ cm}, 30\text{ cm}]$.

As in Scenario 1, both normal illumination conditions and illumination variations are considered. However, in this scenario, the combination of large motion amplitudes and photometric disturbances represents a particularly adverse configuration for classical photometric visual servoing. The objective is to assess whether the proposed wavelet-based representation can preserve stability and convergence under such extreme conditions.

The results corresponding to Scenario 2, illustrated in Figs. 4.8(a–c) to 4.11(a–c), allow a direct comparison between PVS and LWT-PVS in terms of convergence behavior, smoothness of the control signals, and robustness to illumination changes. This scenario plays a critical role in demonstrating the extended convergence domain and enhanced robustness of the proposed LWT-PVS framework.

#### 4.3.2.1 Normal Illumination

For Scenario 2, the initial camera displacement is significantly larger than in Scenario 1, resulting in reduced visual overlap and increased nonlinearity in the control dynamics. The results obtained with classical PVS under normal illumination conditions are presented in Fig. 4.8(a–c).

The photometric error decreases slowly and exhibits **pronounced oscillations** throughout the transient phase. These oscillations reflect the difficulty of accurately modeling the interaction matrix far from the desired pose and the strong coupling between motion components. The corresponding camera velocity profiles display large fluctuations, indicating strong nonlinear effects and limited convergence margins.

Although convergence may eventually occur, the transient phase is long and unstable, and the lack of smoothness in the control signals highlights the reduced robustness of classical PVS

under large initial displacements. Such behavior may compromise both precision and mechanical safety in practical robotic systems.

The results obtained with LWT-PVS under nominal illumination are shown in Fig. 4.9(a–c). In contrast to classical PVS, the proposed approach ensures **smooth and stable convergence** despite the large initial displacement. Both translational and rotational pose errors decrease steadily with minimal oscillations, demonstrating improved damping and numerical conditioning of the interaction matrix.

Moreover, the camera velocity profiles remain continuous and bounded, with significantly reduced peak magnitudes compared to classical PVS. The norm of the visual error decreases at a faster and more regular rate, indicating an **enlarged domain of attraction** and improved convergence efficiency under challenging geometric conditions.

#### 4.3.2.2 Illumination Variations

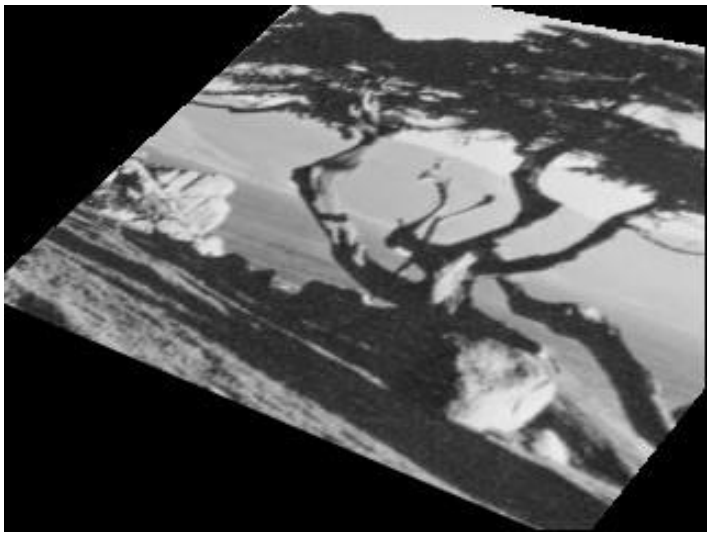
When illumination variations are combined with large initial displacements, classical PVS exhibits **severe instability**, as illustrated in Fig. 4.10(a–c). The photometric error oscillates significantly and may increase over time, indicating a breakdown of the brightness constancy assumption underlying the photometric control formulation. The resulting velocity commands become erratic, reflecting a loss of closed-loop stability.

These results clearly demonstrate the limitations of classical photometric visual servoing in challenging environments where strong nonlinearities and photometric disturbances coexist. Under such conditions, the control law fails to maintain reliable convergence and becomes highly sensitive to illumination-induced inconsistencies.

The corresponding results obtained with LWT-PVS are presented in Fig. 4.11(a–c). Despite the combined challenges of large initial displacement and illumination variations, the system maintains **stable and consistent convergence**. Although the transient phase is longer than under nominal illumination, all pose error components eventually converge toward zero without divergence or sustained oscillatory behavior.

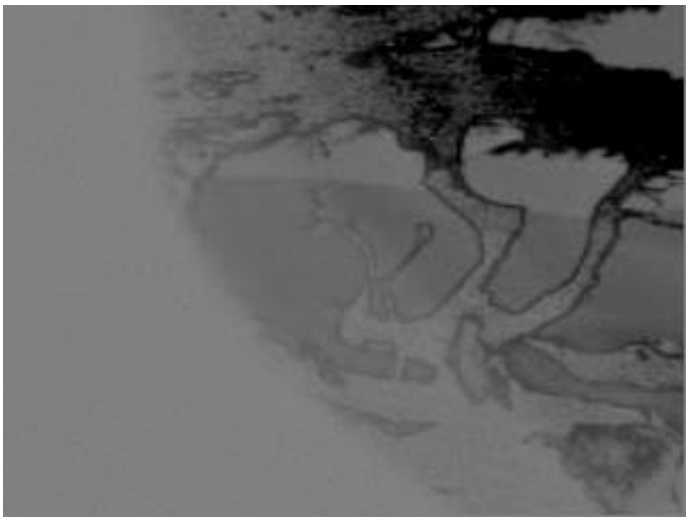
The velocity profiles remain smooth and bounded, and the norm of the visual error decreases monotonically. Notably, the system achieves high positioning accuracy, with final translational errors of $[0.027\,\text{mm},\ -0.038\,\text{mm},\ -0.574\,\text{mm}]$ and rotational errors of $[0.408°,\ -0.548°,\ 0.006°]$.

These results confirm the strong robustness of the proposed wavelet-based approach and demonstrate its ability to preserve stability and precision under extreme visual and geometric perturbations.



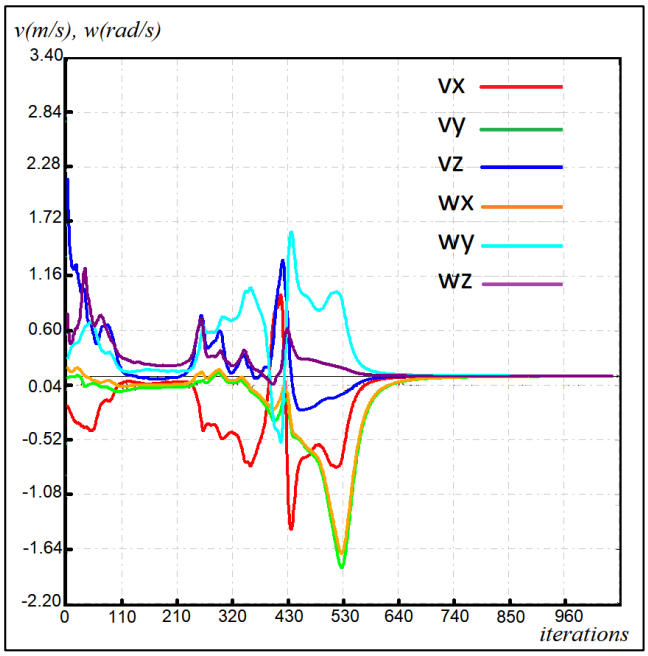
(a) Initial image I

(b) Desired image $I^*$

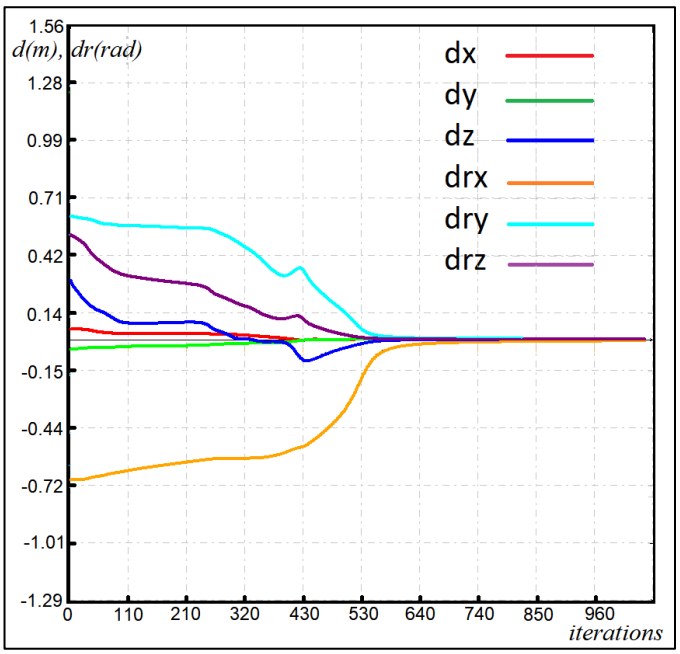


(c) Error between desired images in normal conditions and with illumination variations (I - $I^*$ + 255)/2 (used for visualization)

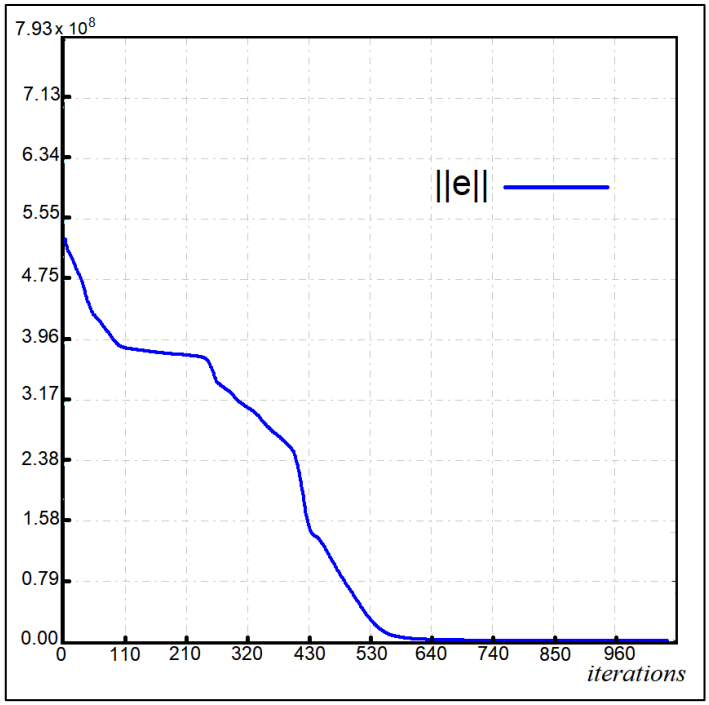
**Figure 4.7:** Simulation Images for scenario 2




(a) Robot velocities applied

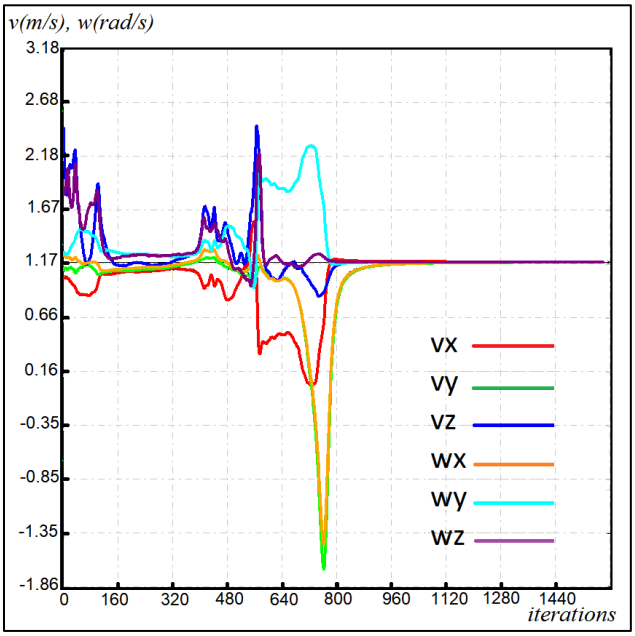



(b) Errors in positioning (in m and rad)

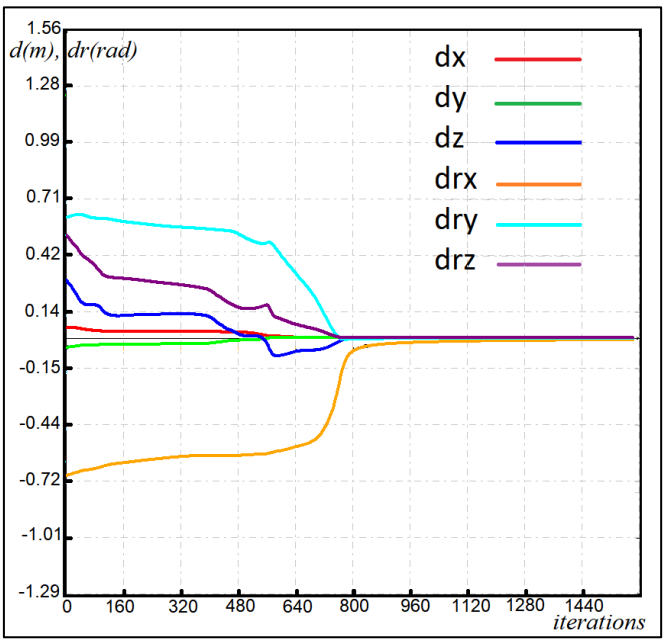



(c) $\|s - s^*\|$

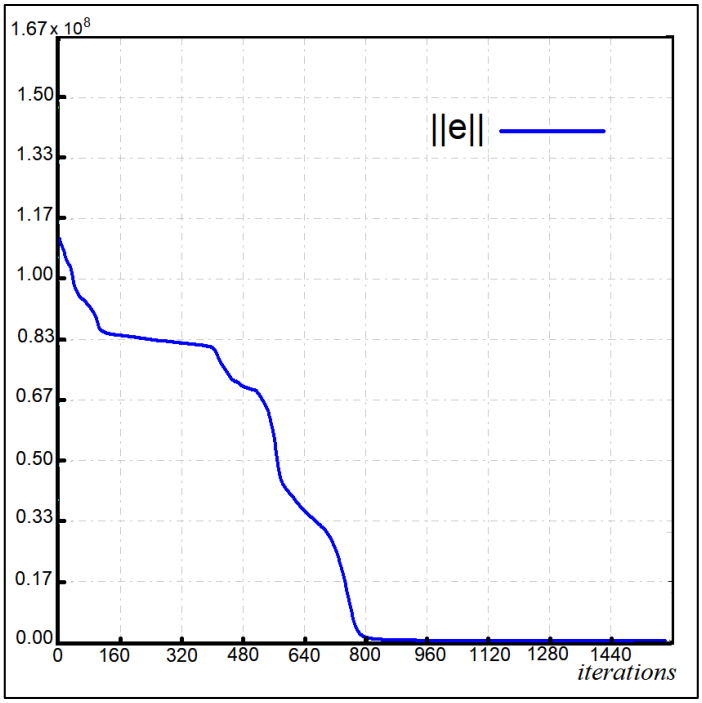
**Figure 4.8:** Simulation Results of **PVS** for scenario 2 in Normal Illumination




(a) Robot velocities applied

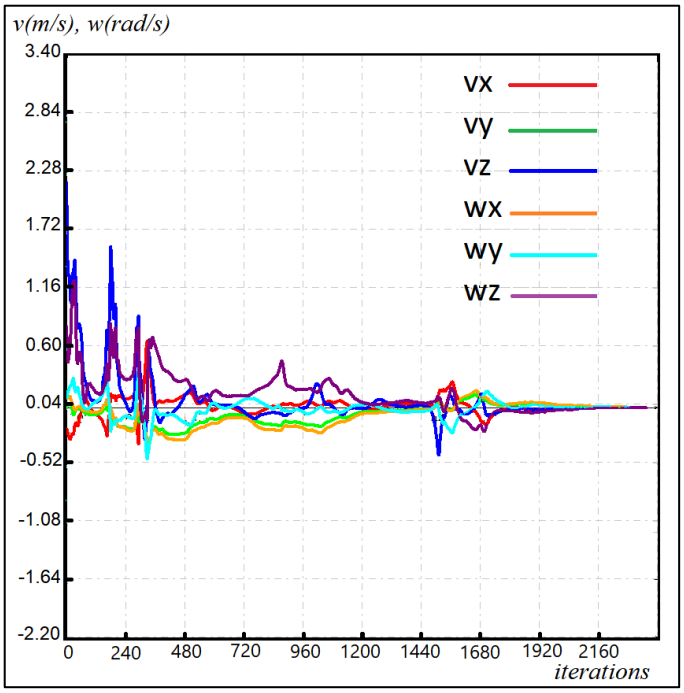



(b) Errors in positioning (in m and rad)

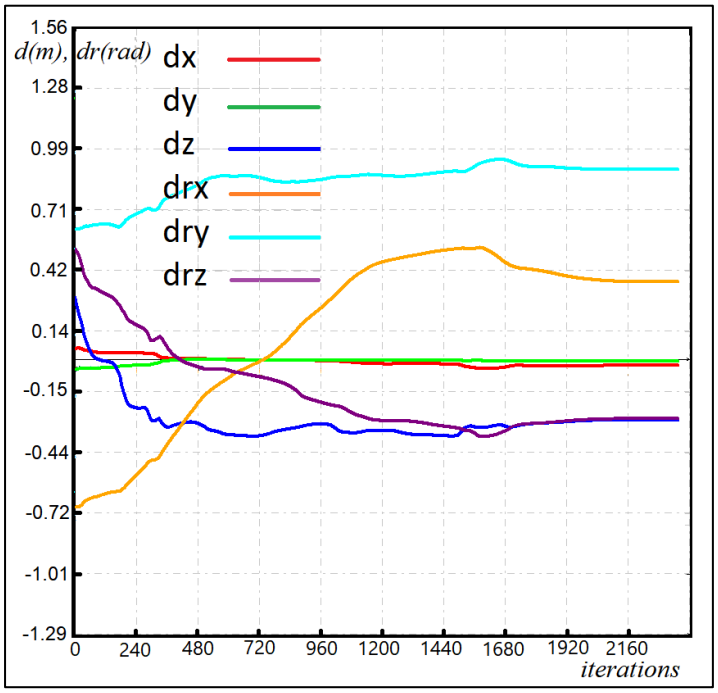



(c) $\|s - s^*\|$

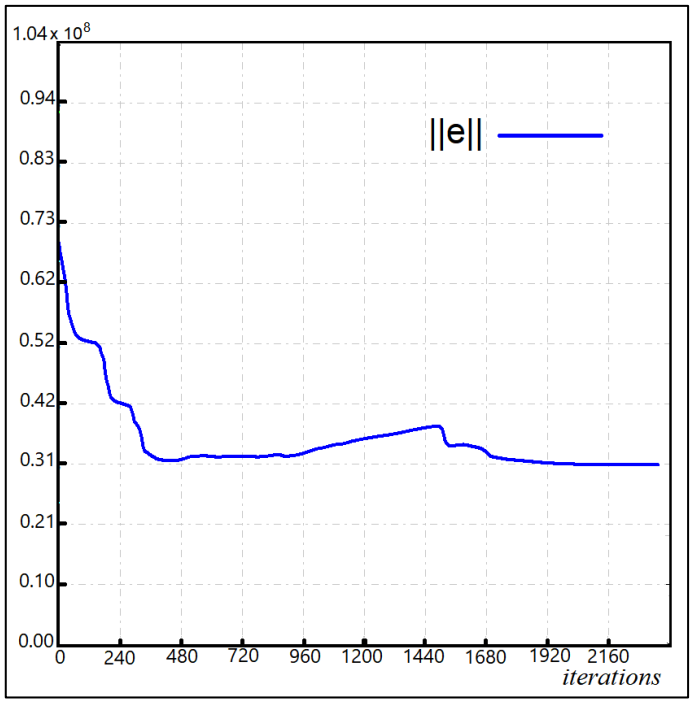
**Figure 4.9:** Simulation Results of **LWT-PVS** for scenario 2 in Normal Illumination




(a) Robot velocities applied

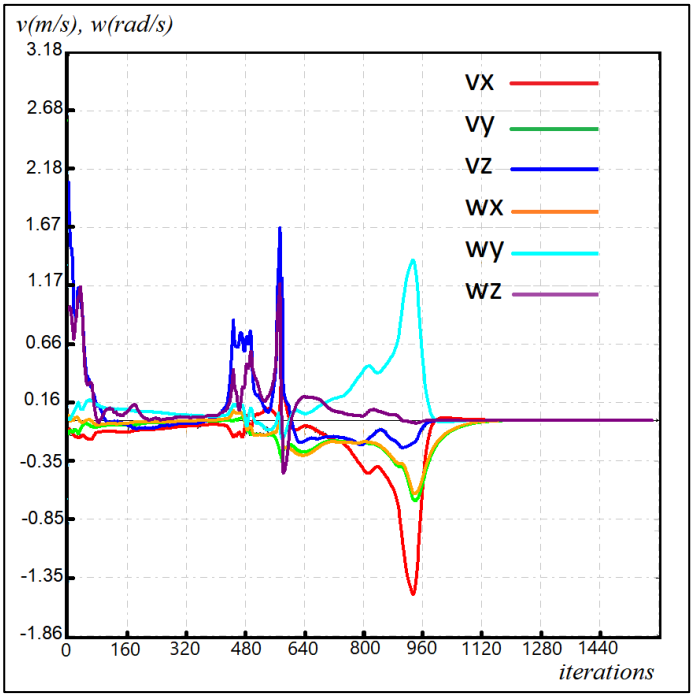



(b) Errors in positioning (in m and rad)

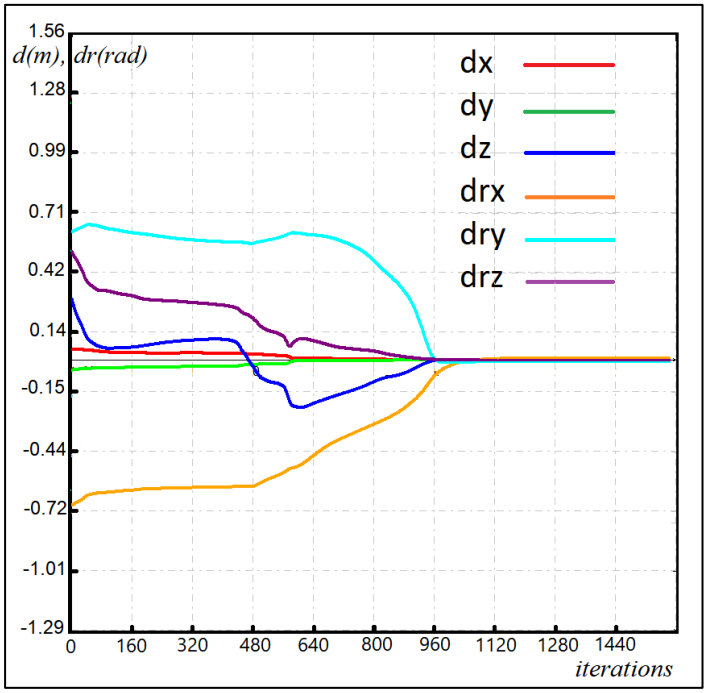



(c) $\|s - s^*\|$

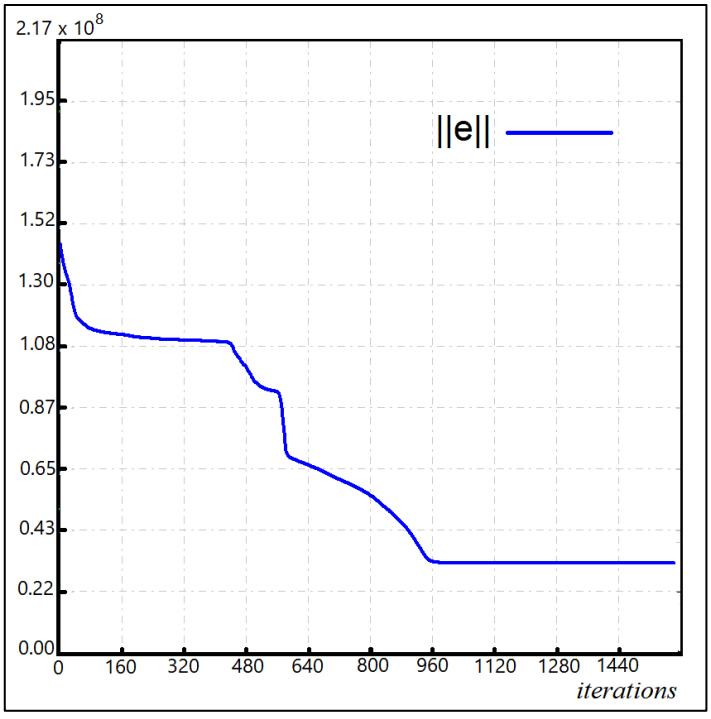
**Figure 4.10:** Simulation Results of **PVS** for scenario 2 in Illumination Variations




(a) Robot velocities applied




(b) Errors in positioning (in m and rad)




(c) $\|s - s^*\|$

**Figure 4.11:** Simulation Results of **LWT-PVS** for scenario 2 in Illumination Variations

### 4.4 Conclusion

In conclusion, the simulation results presented in this chapter demonstrate the effectiveness and robustness of the Lifting Wavelet Transform-based photometric features for 6-dof positioning tasks, even in the presence of significant illumination variations. The proposed method outperforms traditional photometric visual servoing (PVS) techniques, successfully converging to the desired pose in both simulation scenarios. This is particularly noteworthy given the dynamic nature of the visual conditions, where portions of the scene appear and disappear as the camera moves, highlighting the challenges posed by changing lighting conditions.

Across both scenarios and illumination conditions, LWT-PVS consistently outperforms classical PVS. The key advantages of the proposed method can be summarized as follows:

- Faster and more regular convergence.
- Reduced oscillatory behavior.
- Improved numerical conditioning of the interaction matrix.
- Enlarged convergence domain.
- Strong robustness to illumination variations

The results demonstrate that robustness is achieved not by modifying the control law, but by redefining the visual representation. The lifting wavelet transform separates illumination-dependent components from structural information, enabling the control system to focus on invariant features.

From a practical standpoint, the smoother velocity profiles produced by LWT-PVS reduce the risk of actuator saturation and mechanical stress, making the approach suitable for real robotic systems.

Despite the slight nonzero convergence of the error norm, primarily due to discrepancies in pixel intensities between the initial and desired images, the overall performance of the proposed method is exemplary. The controlled lighting variations introduced in both scenarios were effectively handled by the system, confirming the robustness of the Lifting Scheme-derived features in overcoming the challenges associated with illumination changes. These findings not only validate the proposed modeling approach but also demonstrate its potential for real-world applications, where lighting conditions can be unpredictable and dynamic.

Although the simulation results are highly encouraging, certain limitations must be acknowledged. The computational overhead associated with wavelet decomposition is higher

than that of classical PVS, although the lifting scheme significantly mitigates this issue. Additionally, extreme lighting conditions or severe occlusions may still affect performance.

These limitations motivate further investigation through experimental validation and real-time optimization.

Overall, the results confirm the superiority of the proposed method in terms of precision, robustness, and reliability in comparison to traditional visual servoing techniques, making it a promising approach for a wide range of positioning and tracking tasks in robotics and other dynamic systems.

# GENERAL CONCLUSION AND PERSPECTIVES

**Contents**

## 1 General Conclusion

In this thesis, the introduction of a novel method for visual servoing, referred to as **Lifting Wavelet Transform-based Photometric Visual Servoing (LWT-PVS)**, is presented. This method leverages wavelet coefficients computed using the lifting scheme as the primary features for visual servoing tasks, providing an innovative approach to overcoming challenges associated with illumination variations. A key contribution of this work is the derivation of a closed-form interaction matrix that relates the temporal variations in wavelet coefficients to the spatial velocities of the vision sensor.

Extensive simulations have been conducted to validate the proposed method against classical photometric visual servoing techniques, which typically employ raw image intensities as features. The results demonstrated the effectiveness of LWT-PVS in achieving accurate and robust 6-dof positioning, even in the presence of significant illumination changes. The proposed method showed superior performance in dynamic lighting conditions, outperforming traditional photometric approaches in maintaining stability and precision. However, the system was found to experience performance degradation for larger translational and rotational displacements, and it required more computation time for convergence compared to methods that rely on geometric features.

Despite these challenges, the LWT-PVS method offers a promising framework for visual servoing applications, particularly in environments where lighting conditions are unpredictable and where traditional feature extraction methods struggle. The use of multi-resolution wavelet features provides a richer and more robust representation of the image, which enhances the system's ability to perform under variable conditions, offering a solid foundation for future developments in this area.

## 2 Future works

While the proposed LWT-PVS method has shown significant promise, several avenues for improvement and further research remain:

1. **Global Descriptors for Large Displacements**: One limitation identified in this work is the deterioration of performance for large translational and rotational displacements. Future research will focus on incorporating global descriptors, such as moments of wavelet coefficients, to mitigate this issue. By capturing more comprehensive information from the image, these global descriptors could help maintain robustness and accuracy even for large displacements.
2. **Real-Time Visual Servoing**: Another area for future work is the real-time implementation of LWT-PVS in dynamic environments. While this study focused on

simulations, real-world applications demand the integration of real-time algorithms capable of handling environmental variations. Efforts will be made to develop real-time visual servoing algorithms based on wavelet lifting schemes, optimizing computational efficiency while maintaining performance.

3. **Optimization of Convergence Time**: The method's convergence time, particularly for large displacements, could be further optimized. Investigating more efficient optimization techniques or adaptive control schemes that adjust parameters dynamically based on the system's state may improve convergence rates and reduce the computational burden.
4. **Integration with Geometric Features**: To balance the advantages of photometric and geometric methods, a hybrid approach could be developed. This would combine wavelet-based photometric features with traditional geometric features, ensuring that the system can handle a wide range of tasks with varying complexity and accuracy requirements.
5. **Handling Occlusions and Partial Visibility**: The impact of occlusions or partial visibility of the target object remains a challenge in visual servoing. Future work could explore methods for handling partial occlusions, perhaps by incorporating machine learning techniques for feature selection and adaptation under these conditions.
6. **Expansion to 3D Visual Servoing**: While the current work focuses on 2D visual features, extending the approach to 3D visual servoing tasks is a logical next step. This would involve integrating depth information and handling 3D object interactions, allowing the system to operate in more complex, three-dimensional environments.
7. **Exploration of Other Wavelet Variants**: Finally, it would be valuable to explore the use of different wavelet transforms beyond the lifting scheme. By experimenting with other variants, such as curvelets or shearlets, the system's sensitivity to higher frequencies could potentially be improved, and the representation of complex image structures could be enhanced.

In conclusion, while the proposed LWT-PVS method represents a significant advancement in visual servoing, there is still considerable room for improvement. By addressing the aforementioned challenges and exploring new techniques, future research could further enhance the robustness, efficiency, and versatility of visual servoing systems, expanding their applicability to a wider range of real-world tasks.

.

# BIBLIOGRAPHY


1. Hutchinson, Seth, Gregory D. Hager, and Peter I. Corke. “A Tutorial on Visual Servo Control.” *IEEE Transactions on Robotics and Automation* 12, no. 5 (1996): 651–670.
2. Chaumette, François, and Seth Hutchinson. “Visual Servo Control, Part I: Basic Approaches.” *IEEE Robotics & Automation Magazine* 13, no. 4 (2006): 82–90.
3. Lippiello, Vincenzo, Bruno Siciliano, and Luigi Villani. “Position-Based Visual Servoing in Industrial Multirobot Cells Using a Hybrid Camera Configuration.” *IEEE Transactions on Robotics* 23, no. 1 (2007): 73–86.
4. Krupa, Aude, Jérôme Gangloff, Christian Doignon, Michel de Mathelin, Guillaume Morel, Jacques Leroy, Luc Soler, and Jacques Marescaux. “Autonomous 3-D Positioning of Surgical Instruments in Robotized Laparoscopic Surgery Using Visual Servoing.” *IEEE Transactions on Robotics and Automation* 19, no. 5 (2003): 842–853.
5. Barajas, M., J. P. Davalos-Viveros, S. Garcia-Lumbreras, and J. L. Gordillo. “Visual Servoing of UAV Using Cuboid Model with Simultaneous Tracking of Multiple Planar Faces.” In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 596–601, 2013.
6. Wilson, William J., C. C. Williams Hulls, and G. S. Bell. “Relative End-Effector Control Using Cartesian Position-Based Visual Servoing.” *IEEE Transactions on Robotics and Automation* 12, no. 5 (1996): 684–696.
7. Andreff, Nicolas, Bernard Espiau, and Radu Horaud. “Visual Servoing from Lines.” In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, vol. 3, 2000.
8. Tahri, Oussama, Achille Yene Tamtsia, Youcef Mezouar, and Christian Demonceaux. “Visual Servoing Based on Shifted Moments.” *IEEE Transactions on Robotics* 31, no. 3 (2015): 798–804.
9. Liu, T., H. Liu, B. Yang, and Z. Zhang. “LDCNet: Limb Direction Cues-Aware Network for Flexible Human Pose Estimation in Industrial Behavioral Biometrics Systems.” *IEEE Transactions on Industrial Informatics*, 2023.
10. Liu, C., J. Xu, and F. Wang. “A Review of Keypoints’ Detection and Feature Description in Image Registration.” *Scientific Programming*, 2021.
11. Shang, H., C. Liu, and R. Wang. “Measurement Methods of 3D Shape of Large-Scale Complex Surfaces Based on Computer Vision: A Review.” *Measurement*, 2022.
12. Collewet, Christophe, and Éric Marchand. “Photometric Visual Servoing.” *IEEE Transactions on Robotics* 27, no. 4 (2011): 828–834.
13. Dame, Amaury, and Eric Marchand. “Mutual Information-Based Visual Servoing.” *IEEE Transactions on Robotics* 27, no. 5 (2011): 958–969.
14. Richa, R., R. Sznitman, R. Taylor, and G. Hager. “Visual Tracking Using the Sum of Conditional Variance.” In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2953–2958, 2011.
15. Crombez, Nicolas, Guillaume Caron, and É. M. Mouaddib. “Photometric Gaussian Mixtures Based Visual Servoing.” In *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5486–5491.
16. 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.
17. Al-Shanoon, A., and H. Lang. “Robotic Manipulation Based on 3-D Visual Servoing and Deep Neural Networks.” *Robotics and Autonomous Systems*, 2022.
18. Huang, M., L. Ye, J. Li, and F. Yi. “High Precision Control of a Vehicle Based on Photometric Gaussian Direct Visual Servo Without Global Positioning.” In *Proceedings of the Robotics Conference*, 2023.

19. D'Avella, S., C. A. Avizzano, and P. Tripicchio. "ROS-Industrial Based Robotic Cell for Industry 4.0." *Robotics and Computer-Integrated Manufacturing*, 2023.
20. Jiang, H., A. Luo, H. Fan, S. Han, and S. Liu. "Low-Light Image Enhancement with Wavelet-Based Diffusion Models." *ACM Transactions on Graphics* 42, no. 4 (2023).
21. Bai, Y., W. Cheng, W. Wen, and Y. Liu. "Application of Time-Frequency Analysis in Rotating Machinery Fault Diagnosis." *Shock and Vibration*, 2023.
22. Zhang, Y., and K. Lin. "End-to-End Optimized Image Compression with the Frequency-Oriented Transform." *Machine Vision and Applications*, 2024.
23. Guo, T., T. Zhang, E. Lim, M. Lopez-Benitez, and F. Ma. "A Review of Wavelet Analysis and Its Applications: Challenges and Opportunities." *IEEE Transactions*, 2022.
24. Silik, A., M. Noori, W. A. Altabey, and R. Ghiasi. "Comparative Analysis of Wavelet Transform for Time-Frequency Analysis and Transient Localization in Structural Health Monitoring." *Structural Durability and Health Monitoring*, 2021.
25. Zhang, Z., D. Guo, S. Zhou, J. Zhang, et al. "Flight Trajectory Prediction Enabled by Time–Frequency Wavelet Transform." *Nature Communications*, 2023.
26. Harrou, F., A. Zeroual, F. Kadri, and Y. Sun. "Enhancing Road Traffic Flow Prediction with Improved Deep Learning Using Wavelet Transforms." *Results in Engineering*, 2024.
27. Duflot, L.-A., R. Reisenhofer, B. Tamadazte, N. Andreff, and A. Krupa. "Wavelet and Shearlet-Based Image Representations for Visual Servoing." *The International Journal of Robotics Research* 38, no. 4 (2019): 422–450.
28. Tonetti, V., J. C. Pena, M. D. A. Scarpelli, et al. "Landscape Heterogeneity: Concepts, Quantification, Challenges and Future Perspectives." *Environmental* (Cambridge University Press), 2023.
29. Iraci, G. *Visual Servoing for 3D Acquisition*. 2021.
30. Enisoglu, R. "Leveraging AI with Wavelet-Transformed Trend Features for Low-Latency Internet Traffic Identification." 2025.
31. Sweldens, Wim. "The Lifting Scheme: A Construction of Second Generation Wavelets." *SIAM Journal on Mathematical Analysis* 29, no. 2 (1995): 511–546.
32. Gao, W., D. Liu, Q. Wang, and Y. Zhao. "FBLPF-ABOW: An Effective Method for Blink Artifact Removal in Single-Channel EEG Signal." *IEEE Journal of Biomedical and Health Informatics*, 2023.
33. Daubechies, Ingrid. *Ten Lectures on Wavelets*. SIAM, 1992.
34. Aldroubi, Akram, and Michael Unser. *Wavelets in Medicine and Biology*. CRC Press, 1996.
35. Unser, Michael. "Texture Classification and Segmentation Using Wavelet Frames." *IEEE Transactions on Image Processing* 9, no. 4 (2000): 827–836.
36. Alaeiyan, H., M. R. Mosavi, and A. Ayatollahi. "Hybrid Noise Removal to Improve the Accuracy of Inertial Sensors Using Lifting Wavelet Transform Optimized by Genetic Algorithm." *Alexandria Engineering Journal*, 2023.
37. Mohamed, A. F., A. S. Samra, B. Yousif, and A. T. Khalil. "Enhanced Brain Image Security Using a Hybrid of Lifting Wavelet Transform and Support Vector Machine." *Scientific Reports*, 2025.
38. Shi, H., R. Li, X. Bai, Y. Zhang, and L. Min. "A Review for Control Theory and Condition Monitoring on Construction Robots." *Journal of Field Robotics*, 2023.
39. Amon, A. M., K. Fenech, P. Kovács, and T. Dózsa. "Rational Gaussian Wavelets and Corresponding Model-Driven Neural Networks." *arXiv preprint* arXiv:2502.01282, 2025.
40. Shalby, E. M., A. Y. Abdelaziz, and E. S. Ahmed. "A Comprehensive Guide to Selecting Suitable Wavelet Decomposition Levels and Functions in Discrete Wavelet Transform for Fault Detection in Distribution Networks." *Scientific Reports*, 2025.

41. Chavez, D., et al. “Reversible Transformations and Their Applications.” *IEEE Transactions on Signal Processing* 48, no. 5 (2000): 1395–1403.
42. Mendoza, D., and G. Petit. “Wavelet-Based Image Features for Visual Servoing.” *International Journal of Robotics Research* 24, no. 10 (2005): 815–830.
43. Ourak, M., B. Tamadazte, O. Lehmann, and N. Andreff. “Direct Visual Servoing Using Wavelet Coefficients.” *IEEE/ASME Transactions on Mechatronics* 24, no. 3 (2019): 1129–1140.
44. Sundararajan, D. *Discrete Wavelet Transform: A Signal Processing Approach*. Wiley, 2016.
45. Espiau, Bernard, François Chaumette, and Patrick Rives. “A New Approach to Visual Servoing in Robotics.” *IEEE Transactions on Robotics and Automation* 8, no. 3 (1992): 313–326.
46. Weiss, Lee E., Arthur C. Sanderson, and Charles P. Neuman. “Dynamic Sensor-Based Control of Robots with Visual Feedback.” *IEEE Journal on Robotics and Automation* 3, no. 5 (1987): 404–417.
47. Bakthavatchalam, Manikandan. *Utilisation de Moments Photometriques en Asservissement visuel*. PhD diss., Université de Rennes 1, 2015.
48. Allen, P. K., A. Timcenko, B. Yoshimi, and P. Michelman. “Automated Tracking and Grasping of a Moving Object with a Robotic Hand–Eye System.” *IEEE Transactions on Robotics and Automation* 9, no. 2 (1993): 152–165.
49. Flandin, G., François Chaumette, and Éric Marchand. “Eye-in-Hand/Eye-to-Hand Cooperation for Visual Servoing.” In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2741–2746, 2000.
50. Tsai, Roger Y., and R. K. Lenz. “A New Technique for Fully Autonomous and Efficient 3-D Robotics Hand–Eye Calibration.” *IEEE Transactions on Robotics and Automation* 5, no. 3 (1989): 345–358.
51. Kragic, Danica, and Henrik I. Christensen. *Survey on Visual Servoing for Manipulation*. Research Report, KTH Stockholm, 2002.
52. Chaumette, François, and Seth Hutchinson. “Visual Servo Control, Part II: Advanced Approaches.” *IEEE Robotics & Automation Magazine* 14, no. 1 (2007): 109–118.
53. Malis, Ezio, François Chaumette, and S. Boudet. “2-1/2D Visual Servoing.” *IEEE Transactions on Robotics and Automation* 15, no. 2 (1999): 238–250.
54. Chaumette, François. “Potential Problems of Stability and Convergence in Image-Based and Position-Based Visual Servoing.” In *The Confluence of Vision and Control*, 66–78. Springer, 1998.
55. Feddema, John T., and Osmer R. Mitchell. “Vision-Guided Servoing with Feature-Based Trajectory Generation.” *IEEE Transactions on Robotics and Automation* 5, no. 5 (1989): 691–700.
56. Papanikolopoulos, N. P., and P. K. Khosla. “Selection of Features and Evaluation of Visual Measurements for 3-D Robotic Visual Tracking.” In *Proceedings of the International Symposium on Intelligent Control*, 320–325, 1993.
57. Deng, Lingfeng, Farrokh Janabi-Sharifi, and William J. Wilson. “Stability and Robustness of Visual Servoing Methods.” In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 2002.
58. Malis, Ezio, and Patrick Rives. “Robustness of Image-Based Visual Servoing with Respect to Depth Distribution Errors.” In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 2003.
59. Malis, Ezio. “Improving Vision-Based Control Using Efficient Second-Order Minimization Techniques.” In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1843–1848, 2004.

60. Marey, M., and François Chaumette. “Analysis of Classical and New Visual Servoing Control Laws.” In *Proceedings of the IEEE International Conference on Robotics and Automation*, 3244–3249, 2008.
61. Iwatsuki, M., and N. Okiyama. “A New Formulation of Visual Servoing Based on Cylindrical Coordinate System.” *IEEE Transactions on Robotics* 21, no. 2 (2005): 266–273.
62. Chaumette, François. *La Relation Vision-Commande: Théorie et Application à des Tâches Robotiques*. PhD diss., Université Rennes I, 1990.
63. Chaumette, François. “Image Moments: A General and Useful Set of Features for Visual Servoing.” *IEEE Transactions on Robotics* 20, no. 4 (2004): 713–723.
64. Mukundan, Ramakrishnan, and K. R. Ramakrishnan. *Moment Functions in Image Analysis: Theory and Applications*. World Scientific, 1998.
65. Tahri, Oussama, and François Chaumette. “Point-Based and Region-Based Image Moments for Visual Servoing of Planar Objects.” *IEEE Transactions on Robotics* 21, no. 6 (2005): 1116–1127.
66. Abdul Hafez, A. H., Supreeth Achar, and C. V. Jawahar. “Visual Servoing Based on Gaussian Mixture Models.” In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, 3225–3230.
67. Martinet, Philippe. “Comparison of Visual Servoing Techniques: Experimental Results.” In *Proceedings of the European Control Conference*, Karlsruhe, 1999.
68. Comport, Andrew I., Éric Marchand, and François Chaumette. “Robust Model-Based Tracking for Robot Vision.” In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 692–697, 2004.
69. Petit, A., Éric Marchand, and K. Kanani. “A Robust Model-Based Tracker for Space Applications.” In *Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3719–3724.
70. Marchand, Éric, and François Chaumette. “Virtual Visual Servoing: A Framework for Real-Time Augmented Reality.” *Computer Graphics Forum* 21 (2002): 289–297.
71. Doignon, Christophe. “An Introduction to Model-Based Pose Estimation and 3-D Tracking Techniques.” INTECH Open, 2007.
72. Janabi-Sharifi, Farrokh, Lingfeng Deng, and William J. Wilson. “Comparison of Basic Visual Servoing Methods.” *IEEE/ASME Transactions on Mechatronics* 16, no. 5 (2011): 967–983.
73. Malis, Ezio. *Contributions à la Modélisation et à la Commande en Asservissement Visuel*. PhD diss., Université de Rennes 1, 1998.
74. Malis, Ezio, and François Chaumette. “Theoretical Improvements in the Stability Analysis of a New Class of Model-Free Visual Servoing Methods.” *IEEE Transactions on Robotics and Automation* 18, no. 2 (2002): 176–186.
75. Sonka, Milan, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Cengage Learning, 2014.
76. Baker, Simon, and Iain Matthews. “Lucas–Kanade 20 Years On: A Unifying Framework.” *International Journal of Computer Vision* 56, no. 3 (2004): 221–255.
77. Horn, Berthold K. P., and Brian G. Schunck. “Determining Optical Flow.” *Artificial Intelligence* 17 (1981): 185–203.
78. Phong, Bui Tuong. “Illumination for Computer-Generated Pictures.” *Communications of the ACM* 18, no. 6 (1975): 311–317.
79. Collewet, Christophe, and Eric Marchand. *Photometric Visual Servoing*. INRIA Research Report RR-6631, 2008.
80. Ruszczynski, Andrzej P. *Nonlinear Optimization*. Princeton University Press, 2006.
81. Collewet, C., E. Marchand, and F. Chaumette. “Visual Servoing Set Free from Image Processing.” In *Proceedings of the IEEE International Conference on Robotics and Automation*, 81–86, 2008.

82. Marchand, Éric, F. Spindler, and François Chaumette. “ViSP for Visual Servoing: A Generic Software Platform with a Wide Class of Robot Control Skills.” *IEEE Robotics & Automation Magazine* 12, no. 4 (2005): 40–52.
83. Grossmann, A., and J. Morlet. “Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape.” *SIAM Journal on Mathematical Analysis* 15, no. 4 (1974): 723–736.
84. Mallat, Stéphane. “Multiresolution Representation and Wavelets.” PhD diss., University of Pennsylvania, 1988.
85. Mallat, Stéphane. *Une Exploration des Signaux en Ondelettes*. Éditions de l’École Polytechnique, 2000.
86. Kim, Y. H., and J. M. Modestino. “Adaptive Entropy-Coded Subband Coding of Images.” *IEEE Transactions on Image Processing* 1 (1992): 31–48.
87. Taubman, David. “High Performance Scalable Image Compression with EBCOT.” *IEEE Transactions on Image Processing* 9, no. 7 (2000): 1158–1170.
88. Elkefi, A., and M. Antonini. “Compression de maillages 3D multirésolution, transformée en ondelettes 2e génération.” ISRN I3S/RR-2003-30-FR, November 2003.
89. Baudrier, E. *Comparaison d’Images Binaires Reposant sur une Mesure Locale des Dissimilarités: Application à la Classification*. PhD diss.
90. Gouze, A. “Schéma Lifting Quinconce pour la Compression d’Images.” PhD diss., Université de Nice–Sophia Antipolis, 2002.
91. Solé, J. *Lifting Schemes for Image Compression*. PhD diss., Universitat Politècnica de Catalunya, 2005.
92. Barlaud, M., and C. Labit. *Compression et Codage des Images et des Vidéos*. Hermes Science Publications, 2002.
93. Brangoulo, S. *Codage d’Images Fixes et de Vidéos par Ondelettes de Seconde Génération*. PhD diss., Université de Rennes, 2005.
94. Grosbois, R. *Image Security and Processing in the JPEG2000 Compressed Domain*. PhD diss., Université Paris, 2003.
95. Skodras, A., C. Christopoulos, and T. Ebrahimi. “The JPEG2000 Still Image Compression Standard.” *IEEE Signal Processing Magazine* (September 2001): 36–58.
96. Rouag, Fadi Elislam, Nadjiba Terki, Madina Hamiane, Habiba Dahmani, and Mohammed Bourennane. "Lifting wavelet transform-based photometric visual servoing." *Signal, Image and Video Processing* 18, no. 8 (2024): 5867-5878.