國立臺灣科技大學
NATIONAL TAIWAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Universitat de Girona

DOCTORAL THESIS

RECOGNITION AND 6D POSE ESTIMATION OF PARTIALLY OCCLUDED 3D OBJECTS IN CLUTTERED SCENES

雜亂場景中部分遮擋三維物體的辨識和六維姿態估計

Joel Vidal Verdaguer

2019

Doctoral Thesis

Recognition and 6D Pose Estimation of Partially Occluded 3D Objects in Cluttered Scenes

雜亂場景中部分遮擋三維物體的辨識
和六維姿態估計

A thesis presented for the degree of
Doctor of Philosophy
in the Department of Mechanical Engineering at the
National Taiwan University of Science and Technology
and in the Doctoral Program in Technology at the
University of Girona

Joel Vidal Verdaguer

Supervised by:
Prof. Chyi-Yeu Lin
Dr. Robert Martí Marly

2019

摘要

物件辨識和姿態估計對場景理解和高效，靈活和可靠的自主系統而言是一項關鍵任務。傳統上，物件辨識的大多數研究工作都集中對 2D 影像中物件，使用包括雜波、遮擋和不同的照明場景的內容進行檢測和分類。使用機器學習方法進行物件辨識，儘管具有很高的強健性，但這些方法都是從 2D 視角來面對問題，而不是在 3D 空間中考慮物件的精確旋轉方位和位置。在此背景下，根據 3D 場景數據的方法作爲第一種可針對不同的複雜場景穩健地解決 6D 姿態估算問題的解答方案，顯示出迄今爲止最可期待的性能水平和最佳的結果。然而，問題尚未解決，高度混亂的場景和遮擋仍然是這最先進方法的具體挑戰。本文提出並分析了基於表現最佳的點對特徵投票 (point pairs features voting) 方法的創新解決方案，以定義一種創新的以特徵爲基的方法，用於在雜亂場景下對部分遮擋物件進行強健辨識和 6D 姿態估計。

此研究考慮了當前方法的缺點後，定義出新的判別預處理方法、改進的匹配方法、更強健的群聚和一些跟視角相關的後處理步驟。針對具有挑戰性的物件阻擋場景，本研究還提出了一種基於自上而下的視覺注意力和色彩提示的創新解決方法，以提高在物件僅部分可被看見案例之性能。本論文所提出的方法的性能是和 14 種目前最先進的方法，在一個最全面且公開可得，具雜亂和遮擋的眞實場景之基準下一起進行評估。結果顯示，本文提出的方法在所有資料庫上的性能均明顯優於所有一起測試的最先進解決方案。本方法在不同類型的物件和場景皆展示了有效性，特別是在相對低的物件可見情況下提高了性能，擴展了當前 6D 姿勢估計方法的能量。最後，通過建構和測試一種用於智慧製造的創新自動離線編程解決方案，證明了本研究的實用價值。具體地說，建構了一套自動機器手臂整合系統，用以充分發掘物件辨識和姿態估計技術的強健性和進步性。物件辨識方法先將工件姿態資訊提供給靈活的離線編程平台，以全自主方式有效地解決在製造場景中機器手臂整合的關鍵問題。本系統在眞實場景的一系列實驗中進行測試，並與不同的現有解決方案進行比較，顯示出本方法的穩健性和優勢。總體而言，該整合系統顯示了本文提出之最尖端物體辨識方法的價值和潛力，定義出針對高度先進全自主系統的創新智慧解決方案。

Abstract

Object recognition and pose estimation is a crucial task towards scene understanding and highly efficient, flexible and reliable autonomous systems. Traditionally, most research efforts in object recognition have been focused on the detection and classification of objects in two-dimensional images, including clutter, occlusion and different illumination scenarios. Despite reaching a high level of robustness, specially using machine learning approaches, these methods face the problem from a 2D point of view rather than providing the precise rotation and position of the objects in the 3D space. In this context, methods based on three-dimensional scene data appeared as the first solutions to robustly solve this 6D pose estimation problem for different complex scenarios, showing a promising level of performance with the best results so far. However, the problem has not yet been solved, with highly cluttered scenes and occlusions still remaining challenging cases for state-of-the-art methods. This thesis proposes and analyses novel solutions based on the top performing Point Pair Features voting approach to define a novel feature-based method for robust recognition and 6D pose estimation of partially occluded objects in cluttered scenarios.

The research considers the drawbacks of current approaches to define a novel discriminative preprocessing solution, an improved matching method, a more robust clustering and several view-dependent postprocessing steps. Focusing on the challenging occluded cases, the research also proposes a innovative solution based on top-down visual attention and color cues to boost performance in partially visible cases. The performance of the proposed method is evaluated against 14 state-of-the-art solutions on a comprehensive publicly available benchmark with real-world scenarios under clutter and occlusion. The results shows an outstanding improvement for all datasets outperforming all tested state-of-the-art solutions. The validity of the proposed approach is shown for different types of objects and scenarios, specially boosting performance for relatively low visible cases extending the capacities of current 6D pose estimation methods. Finally, the practical value of the research is demonstrated by defining and testing a novel automatic offline

programming solution for intelligent manufacturing. Specifically, an automatic robot integration system that exploits the robustness and benefits of the recognition and pose estimation is proposed. The recognition method provides the workpiece pose information to a flexible offline programming platform, efficiently solving, in an autonomous way, a critical problem for robot integration in manufacturing scenarios. The system is tested on a series of experiments on real-world scenarios and compared against different existing solutions, showing the robustness and benefits of the method. Overall, the system shows the value and potential of a cutting edge object recognition method to define innovative intelligent solutions towards highly advanced autonomous systems.

Acknowledgements

First of all I would like to express my deepest gratitude to Prof. Jerry Lin for his continued support, supervision and guidance without which this thesis would not have been possible. I really appreciate his invaluable help, patience and kindness during all these years and his efforts to make foreigner students feel at home. I would also like to thank very much Dr. Robert Martí for his help and his invaluable and exemplary supervision work. Additionally, I would like to thank all the lab members and friends I made during this journey, who have helped me in so many ways. Finally, I would like to thank my family for their continued support, endless patience and love.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Since the origins of visual image processing, object recognition has been considered as an essential part of visually-guided intelligent systems, representing one of the main motivations and research directions in the computer vision field [7]. In this direction, most research efforts in computer vision have been focused on the detection and classification of objects in two-dimensional images, including clutter, occlusion and different illumination scenarios. Although these methods have reached an astonish level of robustness [39, 92], their capabilities have been focused on understanding the scene in terms of object classification and recognition with binary outputs and regions of interest (ROI), facing the problem from a 2D point of view, rather than inferring the precise rotation and position of specific objects in the 3D space. This spatial problem, commonly known as 3D, 6DOF, 6D pose estimation or pose recovering, is intrinsically related with the complexity and variability of the 3D nature of the space [105] and still remains a long challenging task in the computer vision field. In general, although several types of pose invariant recognition methods for monocular images were presented, few literature focused on explicitly solving the 6D pose estimation problem, e.g. [24, 68, 69], which mainly relied on polyhedral objects and gradients extracted from highly textured cases. In this line, only recently, methods based on template matching [43, 45] and machine learning [57] have shown promising results solving the problem for different types of objects in challenging scenarios. In another direction, the introduction of range data applied to object recognition in the late 70's [80] opened a new research path by providing additional depth information and data sources robust to illumination changes [95]. These methods, based on three-dimensional scene data,

were the first solutions to robustly solve the 6D pose estimation problem for different complex scenarios and still remains the best approaches to the problem. Before the 90's, most solutions focused on the detection of objects within a specific narrow domain of simple solids, polygonal or polynomial shapes [16]. Since then, the increasing computational power and the introduction of more accessible sensor technologies have motivated new branches and research directions, continuously enlarging the object domain and the complexity of the scenes. Recently, different methods based on feature-based approaches [37, 103], template matching [43, 48] and machine learning [21, 58] have obtained promising results on several challenging datasets, including more different types of objects and more complex scenarios. However, these methods are still severely effected by highly cluttered scenes and, in special, occluded cases. On the top of that, the real applicability and performance of the methods remains unclear with only few methods evaluated on practical applications, which mostly focus on simple pick-and-place tasks and constrained environments.

Therefore, object recognition is still a challenging problem for which novel solutions more robust to clutter and occlusion are required. These solutions should be evaluated on public and more comprehensive scenarios under a common criteria providing a clear picture of their comparative performance. Ideally, these solutions should be also evaluated and analyzed for practical cases, showing their applied value in real-world scenarios.

1.2 Visual Object Recognition

Visual object recognition is a fundamental part of scene understanding and has always been a main research direction in computer vision [7]. As a natural capability of the human visual system, this high-level vision process represents a key part of advanced autonomous systems that can perform highly complex tasks, opening the door to a wide range of potential applications from household to industrial environments up to space exploration. Some examples include industrial vision [73], medical imaging [20] or autonomous navigation [35]. The difficulties of the problem, intrinsic in the complexity of the human visual perception, can be seen reflected on the wide range of existing literature, diverging in multiple approaches and research directions, including the dimensionality of the data (i.e 2D or 3D), object domain (e.g. polygonal, polynomial, free-form, rigid or non-rigid), scene complexity (e.g. single or multiple instances, single or multiple objects, with or without clutter and occlusion) and system task (e.g. detection, categorization or localization), coexisting different problems definitions and nomenclatures [7].

The problem of recognizing a set of objects in a given scene and estimating their location has been a matter of intense study from the origins of computer vision in early 60's [89]. Based on the problem definition presented by Besl and Jain [16], the autonomous single-arbitray-view 3D object recognition problem can be simplified and summarized in 3 parts:

1. The targeted objects are examined and respective models are created.
2. Given a single scene view, the following problems are solved for each modeled object:
   (a) Determine the existence of the object in the scene.
   (b) If the object exists, determine the number of instances in the scene.
   (c) For each instance, determine the position and rotation w.r.t a given coordinate system.
3. An optional learning step of new object models from the unknown parts of the scene can be applied. This capacity is commonly referred to as plasticity.

This definition reflects that object detection and position and rotation estimation in the 3D space, i.e special euclidean group SE(3), from now on referred simply as "6D pose" or "pose", are basic tasks of the complete three-dimensional object recognition problem.

In this context, pose estimation of three-dimensional objects in two-dimensional images faces difficulties regarding viewing position, photometric effects, object setting and, for non-rigid cases, changing shape [105]. As an alternative solution, the introduction of range data in object recognition in late 70's [80] opens a new research branch with the inclusion of depth information, increasing robustness against some of the aforementioned difficulties. Nevertheless, before the 90's, most of the proposed methods for three-dimensional object recognition on range data focused on a very restrictive object domain and simple scenarios. The reader can refer to [16] for a survey of the existing methods before 1985. Since then, the introduction of more affordable and multimodal sensors [95] and growing computation power allow a continuous emergence of novel and more robust solutions, increasing the object domain and complexity of the scenes. The reader can refer to [8, 55, 76, 97] for additional surveys of methods and techniques that appeared before 2005. Nowadays, the state-of-the-art of free-form object recognition using range or multimodal data, can be divided in three main categories: Feature-based, template matching and machine learning methods.

The object recognition problem and its variants can be tackled, at its basis, as a matching problem between a system's internal object representation, i.e. object model, and the scene data [16]. For an effective matching, the representations must be somehow corresponding to the real represented data inherent features, such as color or surface characteristics [97]. These object models are usually generated by sensor data or CAD systems. Commonly, the range of techniques involving a prior knowledge provided through an object model follows the same paradigm, known as model-based vision system [8, 14]. Using range data, the object pose estimation problem relying on surface information can be seen as an specific case of the surface matching problem [15]. In this context, the matching between the object and the scene can be treated as the correspondence problem between simple point representations, as a rigid point set registration problem, or using more complex representations, such as meshes, parametric surfaces or solids. A straight forward solution to this matching problem was proposed by minimizing distance error with point set surface fitting approaches, like the Iterative Closes Point (ICP) with its variants [17, 29, 91]. These methods show high precision and efficiency but require a coarse estimation of the object pose, as they may converge to a local minimum. For a global matching solution, where the searching space is prohibitively large, the correspondences commonly rely on the similarity of quantitative values from symbolic descriptors, representing a group or region of data forming a distinguishable element, referred to as *features* [16]. Indeed, this description defines a higher level representation of the object model. Notice that the term *model* or *object model* will be used in this thesis to indistinctly refer to all levels of object representation.

In the literature, we can differentiate two main categories of feature-based approaches with their distinctive pipelines; local and global. Local feature-based methods are based on matching descriptors of local surface characteristics, usually extracted around selected keypoints for efficiency reasons. Among their principal attributes, there is the implicit robustness against occlusion and clutter resulting from the local nature of the description. In turn, these local properties make them sensitive to noise and to the relative size of the surface features. The reader can refer to the work presented by Guo. et al. [42] for a comprehensive state-of-the-art survey of these methods. Global features, on the other hand, follow a different pipeline for which the whole object surface is described by a single or small set of descriptors. For most approaches, each global feature describes each of the views of the object, named view-dependent descriptors. The global nature of these descriptors implies the separation of the described surface from their surroundings, which introduces a segmentation step on the common global feature recognition pipeline. These properties make global feature approaches more

robust to noise and aware of the object structural information, i.e. spatial arrangement and delimitation of surface characteristics, but also make them difficult to use on cluttered and occluded scenes. Some examples of these features are the Extended Gaussian Images(EGI) [52], the Viewpoint Feature Histogram(VFH) [93], the Ensemble of Shape Function(ESF) [111] and the Clustered Viewpoint Feature Histogram(CVFH) [5]. As a particular case, Drost et al. [37] presents in 2010 one of the most successful and powerful feature-based methods combining a global modeling and local matching stages. The method joins benefits of a global object definition and local matching pipeline by efficiently matching and grouping pairs of points using feature quantization and two-dimensional Hough transform-like corresponding grouping. The approach showed an outstanding performances with a promising trade-off between recognition rates and speed. Later, the solution was improved by several authors [18, 30, 36, 47, 60] and its features were studied in detail by [59].

In another direction, template matching techniques, extended from two-dimensional computer vision, have also been proposed for RGB-D data. Basically, these techniques rely on finding the matching of an image's part to a pre-defined template, usually following a blind-search approach. These methods compute a single similarity measure between the template and scene data for each step of a sliding window on the scene data, selecting the highest obtained value above a given threshold as positive matching case. Most research efforts on these methods have been focused on improved similarity measures, specially more robust to illumination, clutter and occlusion, and faster searching strategies, which can efficiently reduce the search space. For two-dimensional images, approaches relying on image gradients [45, 99] have provided relatively good results under occlusion and illumination changes. Based on these methods, Hinterstoisser et al. [44] proposed a template matching technique extended to RGB-D data using quantized surface normals as a depth cue. In a similar fashion, recently, Hodan et al. [50] applied the concept of multimodal matching of [44] on an efficient cascade-style evaluation strategy.

Techniques based on supervised machine learning have been also used for object recognition and pose estimation on RGB-D data. Diverging from traditional approaches, these methods do not match model and data representations through a fixed strategy but *learn* an integrated mapping from the data to the recognition output using training data. Decision trees, Artificial Neural Network (ANN) and Support Vector Machines (SVM) are among some of the common branches of techniques used in supervised learning [19, 62]. Following the same trend as in other fields, new promising methods based on machine learning have raised within the 6D pose estima-

tion research in recent years. Brachmann et al. [22] introduced a method for object pose estimation using a random forest to classify the pixels of a RGB-D image. Tejani et al. [101] adapted the multimodal template of [44] as a scale-invariant patch representation integrated into a random forest. Finally, Kehl et al. [58] presented a method based on Convolutional Neural Network (CNN) using RGB-D patches. Overall, these method are relatively fast during matching while its performance heavily relies on the quality and relevance of the training data.

1.3 Objectives and Scope of Study

The main objectives of this research are to study, define, implement and analyze novel and better real-world applicable solutions for recognition and pose estimation of objects. These solutions should provide higher robustness for different types of scenarios, focusing on the yet-to-be-solved cases for partially occluded objects in clutter scenes. Overall, the research should propose and develop practical solutions that are robust, fast and generic.

In particular, the thesis focused on the following objectives:

- Understanding the recognition and pose estimation problem, its challenges, different solutions and best existing approaches.
- Proposing novel and practical solutions for solving current drawbacks, improving, outperforming and extending the capabilities of existing approaches.
- Analyzing the performance of the proposed solutions on publicity available datasets against different state-of-the-art methods. Solutions should be evaluated in different types of scenarios following a common and standard criteria.
- Defining and developing practical systems, showing their performance on real-world scenarios by solving existing problems with visually-guided autonomous solutions.

Based on the main research direction and research objectives, the scope of this research is limited only to the recognition of rigid (or solid) 3D objects with texture, size and position within the used sensor technology, resolution and working range capacities. The solutions are studied and developed for depth and multimodal data using standard point cloud and mesh representations. The target recognition range includes from common household to industrial 3D objects on different types of challenging scenes with illumination changes, clutter and occlusion.

1.4 Structure of the Thesis

The thesis is divided in 7 chapters as follows. Chapter 1, introduces the thesis background and motivation, the visual object recognition problem from a 3D perspective, the objectives and scope of study, the structure of the thesis and publications. Chapter 2 introduces in detail the Point Pair Feature voting approach and overviews its related methods. Chapter 3 defines a novel feature-based method, analyzing and introducing several new solutions based on the Point Pair Feature voting approach. Chapter 4 faces the occlusion problem proposing an innovative solution based on visual attention and color cues. Chapter 5 analyses the proposed approached on a comprehensive and publicly available benchmark, evaluating the proposed solution against 14 other state-of-the-art methods under a common and fixed evaluation criteria. Chapter 6 shows the practical value of the proposed method by presenting a case study where the method is integrated on an automated offline programming platform for intelligent manufacturing. Finally, Chapter 7 provides the conclusions of the thesis.

1.5 Publications

Parts of this thesis contains material previously published in the following publications:

Title: 6D pose estimation using an improved method based on point pair features
Authors: Joel Vidal, Chyi-Yeu Lin and Robert Martí
Published in: 2018 4th International Conference on Control, Automation and Robotics (ICCAR)
DOI: 10.1109/ICCAR.2018.8384709

Title: A Method for 6D Pose Estimation of Free-Form Rigid Objects Using Point Pair Features on Range Data
Authors: Joel Vidal, Chyi-Yeu Lin, Xavier Lladó and Robert Martí
Published in: Sensors 2018, 18(8), 2678
DOI: 10.3390/s18082678

Title: Automatic robot path integration using three-dimensional vision and offline programming
Authors: Amit Kumar Bedaka, Joel Vidal, Chyi-Yeu Lin
Published in: The International Journal of Advanced Manufacturing Technology (On press)
Note: This article is the result of a joint work with Amit Kumar Bedaka. Both, Amit and Joel, contribute equally to the research.

The author has also collaborated in the following publication:

Title: BOP: Benchmark for 6D Object Pose Estimation
Authors: Tomáš Hodaň, Frank Michel, Eric Brachmann, Wadim Kehl,Anders Glent Buch,Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiří Matas, Carsten Rother
Published in: Computer Vision —ECCV 2018. Lecture Notes in Computer Science, vol 11214. Springer
DOI: 10.1007/978-3-030-01249-6_2

Chapter 2

The Point Pair Features Voting Approach

2.1 Introduction

First introduced by Drost et al. [37], the Point Pair Features voting approach is a feature-based solution combining a global modeling and a local matching stage within a local pipeline using sparse features. The approach represents a compromised solution between the local and global approaches, showing a promising trade-off between recognition rates and speed. The approach unique characteristics, outstanding performance and potential defines the basis of the proposed solutions. This chapter introduces the basics of the approach and overviews the most important related methods.

2.2 The Basics

Using point cloud representations of oriented points (i.e., points with normals), the method relies on four-dimensional features extracted from pairs of points (from now on "point pairs" or simply "pairs") to globally describe the whole object from each surface point in a way that later the object can be locally matched with the scene. This four-dimensional feature, called Point Pair Feature or PPF, defines an asymmetric description between two oriented points by encoding their relative distance and normal information, as shown in Figure 2.1. In detail, having a set of points in the 3D space $\boldsymbol{M} \subset \mathbb{R}^3$ representing the model object, for a given 3D point $m_r \in \boldsymbol{M}$, called reference, and a given 3D point $m_s \in \boldsymbol{M}$, named second, such that $m_r \neq m_s$, with their respective unit normal vectors $\hat{n}_{m_r}$ and $\hat{n}_{m_s}$, a model four-dimensional

feature $f^m \in (\boldsymbol{F^m} \subset \mathbb{R}^4)$ is defined by Equation (2.1),

$$F_{rs}(m_r, m_s, \hat{n}_{m_r}, \hat{n}_{m_s}) = [\, ||\vec{d}||, \angle(\hat{n}_{m_r}, \vec{d}), \angle(\hat{n}_{m_s}, \vec{d}), \angle(\hat{n}_{m_r}, \hat{n}_{m_s}) \,]^T, \quad (2.1)$$

where $\vec{d} = (m_{s_x} - m_{r_x}, m_{s_y} - m_{r_y}, m_{s_z} - m_{r_z})$ and $\angle(\vec{a}, \vec{b})$ is the angle between the vector $\vec{a}$ and $\vec{b}$. In the same way, having a set of point $\boldsymbol{S} \subset \mathbb{R}^3$ representing the scene data, the function F_{rs} can be applied to compute a scene PPF using a pair of scene points $s_r, s_s \in \boldsymbol{S}$ such that $s_r \neq s_s$, with their respective unit normal vectors $\hat{n}_{s_r}$ and $\hat{n}_{s_s}$. Notice that, if the object model has $|\boldsymbol{M}|$ points, the total number of features is defined by $|\boldsymbol{F^m}| = |\boldsymbol{M}|^2 - |\boldsymbol{M}|$. In order to reduce the effect of this square relation on the method performance, the input data of both model and scene are downsampled with respect to the model size, effectively decreasing the complexity of the system.

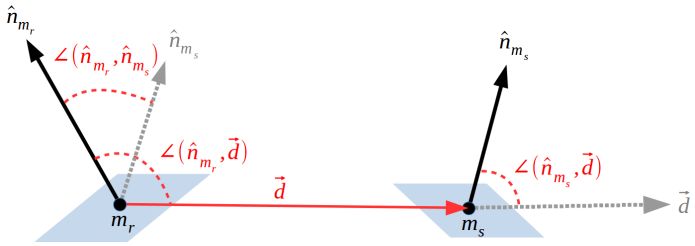


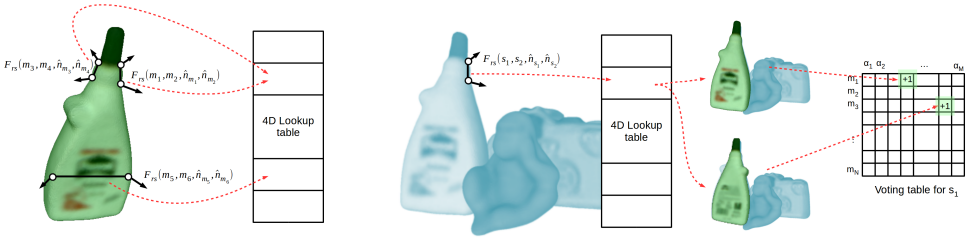
Figure 2.1: Point Pair Feature definition for a model's point pair (m_r, m_s).

The method can be divided into two main stages: modeling and matching. On modeling, the global model descriptor is created by computing and saving all the possible model pairs with their related PPF. During the matching stage, the model pose in the scene is estimated by matching the scene pairs with the stored model pairs using the PPF. This matching process consists of two distinctive parts: (1) find the correspondence between the pairs using the four-dimensional features and (2) group the correspondences generating hypotheses' poses.

The correspondence problem between similar point pairs is efficiently solved by grouping the pairs with the same quantized PPF on a hash table or, alternatively, a four-dimensional lookup table. Quantizing the feature space defines a mapping from each four-dimensional space element to the set of all point pairs that generate this specific feature. In particular, for the object model, this mapping from quantized features to sets of model pairs defines the object model description expressed by the function $L : \mathbb{Z}^4 \rightarrow \mathcal{P}(\boldsymbol{M_{pp}})$, where $\boldsymbol{M_{pp}} = \{(m_r, m_s) \mid m_r, m_s \in \boldsymbol{M}, m_r \neq m_s\}$ and $\mathcal{P}(\boldsymbol{X})$ represents the power set of $\boldsymbol{X}$. In other words, point pairs that generate the same quantized PPF are grouped together on the same table position pointed by their common quantized index, effectively grouping pairs with similar features. This

process of model construction is done during the modeling stage, as shown in Figure 2.2a for three sample point pairs. Using this model description, given one scene pair, similar model pairs can be retrieved by accessing a table position pointed by the PPF quantized index. The quantization index is obtained by a quantization function $Q : \mathbb{R}^4 \rightarrow \mathbb{Z}^4$ using the step size Δ_{dist} for the first dimension and Δ_{angle} for the remaining three dimensions. The quantization step size will bound the similarity level, i.e., correspondence distance, between matching features, and hence point pairs. Defining a function $N : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that computes a normal from a point, the correspondence matching subset of model pairs $\boldsymbol{A} \subseteq \boldsymbol{M_{pp}}$ for a given scene pair (s_r, s_s) and its related quantized feature $\bar{f}^s = Q(F_{rs}(s_r, s_s, \hat{n_{s_r}}, \hat{n_{s_s}}))$ is defined by Equation (2.2):

$$L(\bar{f}^s) = \{(m_r, m_s) \in \boldsymbol{M_{pp}} \mid Q(F_{rs}(m_r, m_s, N(m_r), N(m_s))) = \bar{f}^s\}. \quad (2.2)$$



(a) (b)

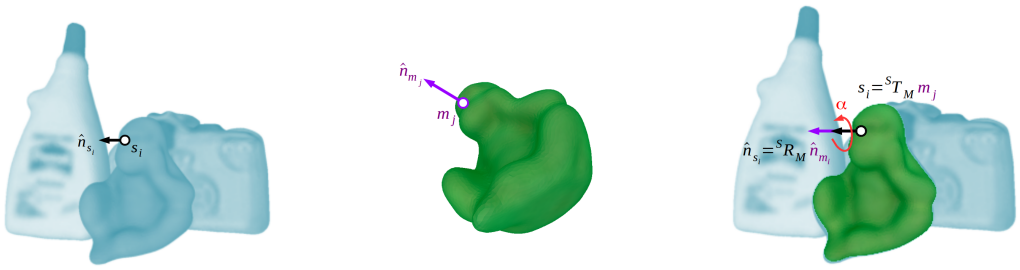
Figure 2.2: Representation of the modeling and matching steps of the Point Pair Features voting approach. (**a**) modeling example for three point pairs from the model; (**b**) matching example for one point pair from the scene.

From each scene-model point pair correspondence, a 6D pose transformation, or hypothesis, can be generated. Specifically, for a corresponding point pair $(m_r, m_s) \in \boldsymbol{A}$, the matched reference points (s_r, m_r) and their normals $(\hat{n}_{s_r}, \hat{n}_{m_r})$ constrain five degrees of freedom, aligning both oriented points, and the second points (s_s, m_s), as long as they are non-collinear, constrain the remaining degree of freedom, which is a rotation around the aligned normals. However, the discriminative capability of a single four-dimensional feature from two sparse oriented points is clearly not enough to uniquely encode any surface characteristic, producing wrong correspondences. Therefore, the method requires a group of consistent correspondences to support the same hypothesis. Actually, the more correspondences support a single pose, the more likely this will be. In this regard, grouping consistent point pair correspondences, or, alternatively, 6D poses obtained from corresponding pairs have a high dimension complexity. In order to effectively solve

this problem, a local coordinate, which we will refer to as LC, is used to efficiently group the poses within a two-dimensional space. As with two corresponding pairs, for a given scene point $s_i \in \boldsymbol{S}$ that belongs to the object model, a 6D pose can be defined by only using one corresponding model point $m_j \in \boldsymbol{M}$ and a rotation angle α around their two aligned normals, i.e., $\hat{n}_{s_i}$ and $\hat{n}_{m_j}$. In this way, for the scene point s_i, a 6D pose transformation candidate ${}^{S}T_M \in SE(3)$ can defined by the LC represented by the parameters (m_j, α), as shown in Figure 2.3. To solve this transformation, both points and normals are aligned respectively with the origin and x-axis of a common world coordinate system $\{W\}$. Taking the scene point, this alignment can be expressed by the transformation ${}^{W}T_S = (R, t) \in SE(3)$. The rotation that aligns the normal vector $\hat{n}_{s_i}$ to the x-axis $\hat{e}_x$ is defined by the axis-angle representation $\theta\hat{v}$, where $\theta = \angle(\hat{n}_{s_i}, \hat{e}_x)$ and $\hat{v} = \frac{\hat{n}_{s_i} \times \hat{e}_x}{||\hat{n}_{s_i} \times \hat{e}_x||}$. Therefore, the rotation matrix $R \in SO(3)$ can be efficiently found using the Rodrigues' rotation formula [33]. In turn, the translation $t \in \mathbb{R}^3$ is defined by $t = -Rs_i$. Exactly in the same way, the transformation ${}^{W}T_M \in SE(3)$ is found for the model point m_j and its normal $\hat{n}_{m_j}$. Using these two transformations and the rotation angle, the 6D pose for a given object instance is defined by Equation (2.3):

$$ {}^{S}T_M = ({}^{W}T_S)^{-1} R_x(\alpha) {}^{W}T_M, \tag{2.3} $$

where $R_x(\beta) \in SO(3)$ represents a rotation of β angle around the x-axis. Using the LC, the correspondence grouping problem can be individually tackled for any scene pair created from s_i by grouping the corresponding model pairs in a two-dimensional space using the parameters (m_j, α).



(a) (b) (c)

Figure 2.3: Representation of the local coordinate (LC) system used by the Point Pair Features voting approach; (**a**) scene oriented point; (**b**) corresponding object model oriented point; (**c**) alignment of the model with the scene by using the two oriented points and the α angle.

During grouping and hypothesis generation, for every reference scene point s_i, the method intends to find the LC, i.e., (m_j, α), which defines the best fitting model pose on the scene data or, in other words, that maximizes the number of pairs correspondences that support it. This correspondence grouping problem is solved by defining a two-dimensional voting table or *accumulator*, in a Generalized Hough Transform manner, representing the parameter space of the LC, where one dimension represents the corresponding model point m_j and the other the quantized rotation angle α. In particular, for each possible scene pair generated from s_i, i.e., $(s_r, s_s) \in \{(s_k, s_l) \mid s_k, s_l \in \boldsymbol{S}, s_k \neq s_l, s_k = s_i\}$, a LC will be defined by a corresponding pair (m_r, m_s) reference point, i.e., $m_j = m_r$, and the rotation angle α defined by the two second points (s_s, m_s). The corresponding model pairs are retrieved from the lookup table using the quantized PPF and, for each obtained LC, a vote is cast on the table, as represented by Figure 2.2b for a single pair. After all pairs are checked, the peak of the table represents the most supported LC, and hence the most likely pose, for this specific s_i point. This process is applied to all or, alternatively, a fraction of the scene points, obtaining a set of plausible hypotheses.

To increase the efficiency of the voting part, which requires to compute the α angle for each pair correspondence, it is possible to split the rotation angle α in two parts; one part related to the model point, α_m, and one part related to the scene point, α_s. In detail, taking into account that in the intermediate world coordinate system the α angle is defined around the x-axis, the rotation on the two-dimensional yz-plane can be divided with respect to the positive y-axis. In this case, the α_m and α_s will be defined as the rotation angles between the positive y-axis vector $\hat{e}_y$ and the yz-plane projection of the vectors obtained by the world transformed second points of the model pair $({}^W T_M \, m_s)$ and scene pairs $({}^W T_S \, s_s)$. As shown in Figure 2.4, these angles can be defined as $\alpha_s = a\tan2(a_z, a_y)$ and $\alpha_m = a\tan2(b_z, b_y)$, where $a = {}^W T_S \, s_s$, $b = {}^W T_M \, m_s$ and $a\tan2(\beta, \gamma)$ represents the multi-valued inverse tangent. With this solution, the model angle can be precomputed during the modeling stage and saved alongside the reference point in the lookup table (m_r, α_m). Later, during the matching stage, for each scene pair, the α angle is computed by adding the two angles. Considering that α is defined from the model to the scene, the total angle can be computed as Eq. (3.1),

$$\alpha = \alpha_s - \alpha_m. \tag{2.4}$$

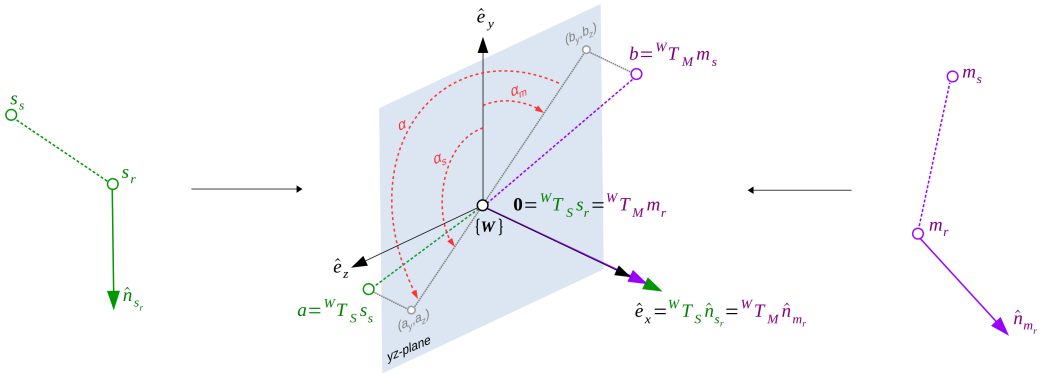


Figure 2.4: Representation of the LC α angle definition from two corresponding pairs (s_r, s_s) and (m_r, m_s).

Finally, in order to join similar candidate poses generated from different scene reference points, the method is completed with a clustering approach that groups similar poses that do not vary in rotation and translation more than a threshold.

2.3 Related Methods

Since 2010, several new methods have been proposed based on Drost's original idea. First, in 2011, Kim and Medioni [60] proposed a variation of the original method by including visible context information, differentiating visible points, points on the surface and invisible points. In addition, they used two novel verification steps to check for surface alignment and surface separability. Drost and Ilic [36] introduced in 2012 a multimodal extension of the method including edge information extracted from RGB data. Moreover, their approach included non-maximum suppression of the clustered poses and a pose refinement step based on Iterative Closest Point (ICP). In a different direction, Choi et al. introduce in 2012 [30] a multimodal approach by extending the point pair features to 10 dimensions by including the HSV color information of the point pairs. Following the same direction, the authors proposed an extended solution in 2016 [31], where the color quantization parameters were automatically estimated for different objects. Birdal and Ilic [18], in 2015, analyzed some drawbacks of the method and proposed a novel framework to overcome some of the issues regarding the high dimensionality of the search space, sensitivity of the correspondence and the effect of outliers and low density surfaces. Their novel solution included a coarse-to-fine segmentation step, a weighted Hough voting and a fast ranking and verification postprocessing steps. Hinterstoisser at al. [47] published in 2016 a

major revision of the method presenting a novel approach with higher performance and improved robustness against occlusion and clutter. Among their contributions, they proposed to use normal information during preprocessing and mitigated the discretization problems of the data by an exhaustive neighbor checking. Their method used two different size voting zones and an additional data structure to avoid multiple voting of similar pairs. In addition, they proposed an improved bottom-up clustering strategy and several additional verification steps. Recently, Kiforenko et al. [59] presented a complete performance evaluation of the Point Pair Features including a detailed comparison with the most popular local features.

Chapter 3

A Novel Approach Based on Point Pair Features

In this chapter, a new method based on the Point Pair Features voting approach [37] for robust 6D pose estimation of free-form objects under clutter and occlusions on range data is defined. In detail, the original ideas presented in [37] are improved and a complete method within a local feature-based pipeline is defined.

3.1 Method's Overview

The proposed method pipeline, shown in Figure 3.1, can be divided in an *Offline* modeling and an *Online* matching stages with six basic steps: *Preprocessing*, *Feature Extraction*, *Matching*, *Hypothesis Generation*, *Clustering* and *Postprocessing*. Due to the method's particular correspondence grouping step, using a voting table for each scene point, a basic straightforward implementation will require to create a voting table for each of the scene points during the hypothesis generation step, with large memory requirements. From a practical point of view, a more efficient solution is to iteratively generate a hypothesis for each scene point using a single voting table. In this regard, the green fine dotted box in Figure 3.1 represents the iterative implementation of the steps *Feature Extraction*, *Matching* and *Hypothesis generation* for each scene point. The method is considered to work with mesh data for modeling and organized point cloud for matching, as standardized data types.

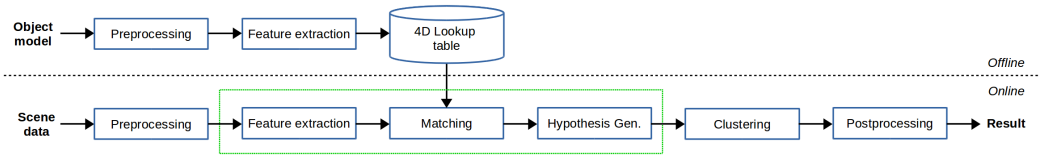


Figure 3.1: Proposed method pipeline. The green fine dotted box represents the iterative implementation of the steps *Feature Extraction*, *Matching* and *Hypothesis generation* for each scene point.

3.2 Preprocessing

The Point Pair Feature voting method strongly relies on the discriminative effect of the PPF and their sparse nature to allow an efficient, structural aware local matching. The performance of the original four-dimensional PPF and its variants has been deeply studied by Kiforenko et al. [59]. Their work concludes that a set of PPF globally defining a model point has stronger discriminative capability than most local features. On the other hand, they also showed that, despite its robustness, the PPF are significantly affected by noise. In fact, individually, each feature relies on the quality and relevance of the normal and distance information extracted from the sparse surface characteristics provided by the pairs of the sampled data. In this sense, low quality or non-discriminative features will reduce speed and decrease the recognition performance of the method. Therefore, the overall global description performance, in terms of time and recognition, depends on the number of features and the relevance and quality of each individual feature. This relation makes the performance of the approach to rely significantly on the preprocessing steps. In turn, the sampling and normal estimation in preprocessing are mainly affected by the sensor noise and the relative size of the underlying surface characteristics. Taking these considerations into account, we propose a combination of two normal estimation approaches and a novel downsampling methodology that mitigates sensor noise, accounts for surface variability and maximizes the discriminative effect of the features.

3.2.1 Normal Estimation

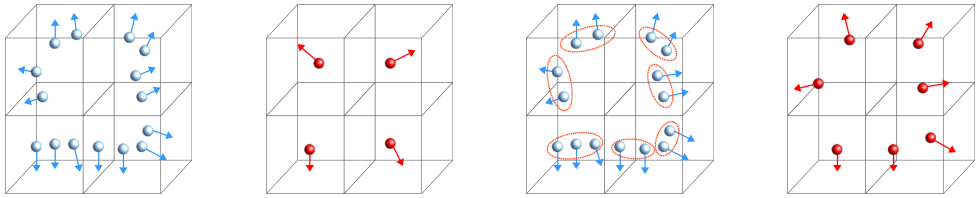
For the normal estimation problem, we propose using two different variants regarding the input data representation of each stage. For the *Offline* stage, using reconstructed or CAD mesh data, the normals are estimated by averaging the normal planes of each vertex' s surrounding triangles. In this case, noise and resolution limitations regarding surface reconstruction techniques

are considered out of the scope of this manuscript, and thus not considered. For the *online* stage, using the organized point cloud data, we use the method proposed in [44], based on the first order Taylor expansion, including a bilateral filter inspired solution for cases where the surface depth difference is above a given threshold. These two approaches provide a normal estimation relative to the data source resolution and, additionally, the *online* method provides an efficient and robust estimation against sensor noise [44]. Notice that noisy and spiky surface data will affect the quality of the normal estimation step and, in turn, the downsampling step, decreasing the method efficiency and performance. In this regard, a normal estimation robust to noise is a basic part of the method, with a high impact on the matching results [59].

3.2.2 Downsampling

Traditional downsampling methods, also called subsampling or decimation, based on voxel-grid or Poisson-disk sampling, have a fixed size structure that do not consider local information and tend to either average or ignore parts of the data, removing and distorting important characteristics of the underlying surface. If these characteristics want to be somehow preserved, these methods require increasing the sampling rate, i.e., decrease voxel size, which in turn dramatically decreases the algorithm performance adding superfluous data. As an alternative to these problems, we propose a novel approach that accounts for the variability of the surface data without increasing non-discriminative pairs.

The proposed method is based on a novel voxel-grid downsampling approach using surface information and an additional non-discriminative pairs' averaging step. The method starts by computing a voxel-grid structure for the point cloud data. For each voxel cell, a greedy clustering approach is used to group those points with similar normal information, i.e., the angle between normals is smaller than a threshold. Then, for each clustered group, we average the oriented points, effectively merging the similar points while keeping discriminative data. Figure 3.2 shows a simplified comparison between the common voxel-grid average method and the proposed normal clustering approach. Notice that, especially due to the PPF quantization space, for close points, distance is not relevant and normals encode the most discriminative information about underlying surface characteristics. As in the original method, the voxel size is set to Δ_{dist}, defining a value relative to the model size. However, in our method, the parameter effect on the algorithm performance is significantly reduced, moving towards a more robust parameter-independent method.



(a) (b) (c) (d)

Figure 3.2: Representation of the normal clustering voxel-grid downsampling. (**a**) original cloud with a voxel-grid structure; (**b**) common downsampling by average; (**c**) novel proposed clustering approach; (**d**) result of the proposed clustering approach.

Despite its local efficiency, this downsampling method does not account for the cases where the non-relevant surface characteristics are bigger than the voxel size. To mitigate these cases, when neighboring downsampled voxels contain similar data, we propose an additional step to average those points that do not provide additional surface information. This process is done by defining a new voxel-grid structure, with a much bigger voxel size (e.g., two or three times bigger), and averaging all points that do not have relevant normal data compared with all their neighbors' voxels points. This step will reduce the points on planar surfaces, decreasing the number of total votes supporting the hypothesis. However, as the process is applied equally to the scene and the object, this will mainly decrease the votes of the non-discriminative parts, effectively increasing the value of the rest of the surface data.

3.3 Feature Extraction

As mentioned before, Kiforenko et al. [59] published an exhaustive study and comparison of different types of PPF. Their results show that, despite the multimodal variants, the original four-dimensional feature [37] provides the best performance for range data. In light of this result, we propose to keep using the original PPF introduced in Chapter 2, represented in Figure 2.1 and Equation (2.1).

During the *Offline* stage, the model bounding box is obtained and the model diameter $d_m \in \mathbb{R}$ is estimated as the diagonal length of the box. For a given PPF, a four-dimensional index is obtained using the quantization function defined in Equation (3.1):

$$Q(\boldsymbol{x}) = [\lfloor \frac{x_1}{\Delta_{dist}} \rfloor, \lfloor \frac{x_2}{\Delta_{angle}} \rfloor, \lfloor \frac{x_3}{\Delta_{angle}} \rfloor, \lfloor \frac{x_4}{\Delta_{angle}} \rfloor]^T, \tag{3.1}$$

where the quantization step Δ_{dist} is set to 0.05 d_m and Δ_{angle} is fixed to $\frac{\pi}{15}$. These values have been set as a trade-off between recognition rates and speed. In this way, the lookup table is defined with a size of $\lceil \frac{d_m}{\Delta_{dist}} \rceil \times \lceil \frac{\pi}{\Delta_{angle}} \rceil \times \lceil \frac{\pi}{\Delta_{angle}} \rceil \times \lceil \frac{\pi}{\Delta_{angle}} \rceil$. After preprocessing, for each model pair, the quantized PPF index is obtained and the reference point and the computed α_m angle are saved into the pointed table cell. In this case, all points of the model are used.

During the online stage, for each reference point, all possible point pairs will be computed and, using the four-dimensional lookup table, matched with the object model. Following the solution proposed by [37], only one of every five points (in input order) will be used as a reference point, while all points will be used as second points. To improve the efficiency of the matching part, in order to avoid considering pairs further away than the model diameter d_m, for each scene reference point, we propose to use an efficient Kd-tree structure to obtain only the second points within the model diameter.

3.4 Matching

As explained in Chapter 2, the Point Pair Feature voting approach solves the matching problem by quantizing the feature space, grouping all similar pairs under the same four-dimensional index. As a result, any point pair is matched with all the other pairs that generate the same quantized features in a constant time. Despite its efficiency, this approach has two main drawbacks.

The first drawback is regarding the noise effect on the quantized nature of the point pairs matching, as the quantization function Q can output different indices for very similar real values. In these cases, similar pairs generate different quantized index, which points to different cells of the lookup table, missing correct correspondences during the online stage. Figure 3.3a shows a one-dimensional representation of the problem. A straightforward solution was proposed by [47]. Their approach *spreads* the PPF quantized index to all its neighbors, effectively retrieving from the lookup table all the corresponding pairs pointed by the index alongside the pairs stored in its 80 neighboring cells, i.e., $3^4 - 1$ cells for a four-dimensional table. The main drawback with this method is the increased number of access to the lookup table, which is done for each matching PPF, decreasing significantly the time performance of the method. In addition, another problem can arise regarding the corresponding distance between features. If the quantization size Δ is kept, see Figure 3.3b, the correspondence distance increases, dramatically augmenting the number of corresponding pairs and introducing matching pairs with lower similarity level to the voting scheme. An alternative approach is to decrease

the quantization size $\frac{\Delta}{3}$, see Figure 3.3c, accounting for the neighboring cells, using a bigger data structure.

We propose a more efficient solution by only checking a maximum of 16 neighbors keeping the size of the quantization step, as shown in Figure 3.3d. Considering that the difference between similar pairs are mainly generated by sensor noise, it is reasonable to assume that this noise follows a normal distribution characterized by a relatively small standard deviation σ, i.e., smaller than half of the quantization step $\sigma < \frac{\Delta}{2}$. Based on this assumption, we propose to check the quantization error $e_q = (\frac{x}{\Delta} - \lfloor \frac{x}{\Delta} \rfloor) \in [0, 1)$ to determine which neighbors are more likely to be affected by the noise. This process is defined for each dimension by the piecewise function represented in Equation (3.2):

$$N(e_q) = \begin{cases} -1, & e_q < \frac{\sigma}{\Delta}, \\ 1, & e_q > (1 - \frac{\sigma}{\Delta}), \\ 0, & \text{otherwise}, \end{cases} \tag{3.2}$$

where the result is interpreted as follows: -1 indicates that left neighbor could be affected, 1 indicates that right neighbor could be affected and 0 indicates that no neighbor is likely to be affected.

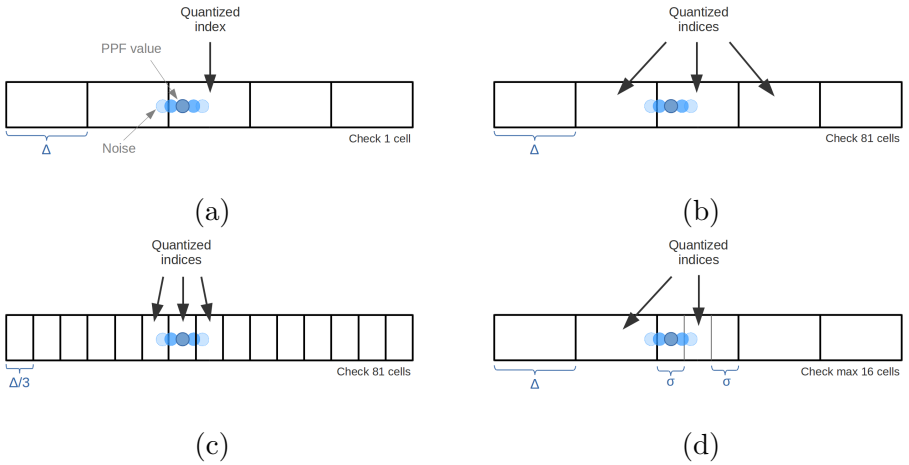


Figure 3.3: One-dimensional example of the noise effect on the lookup table during matching with four different strategies. (**a**) original approach; (**b**) approach of [47] using Δ; (**c**) approach of [47] using $\frac{\Delta}{3}$; (**d**) our approach.

During matching, for each dimension, those pairs from neighbors that are likely to be affected by noise are retrieved. In practice, for generalization, we set the standard deviation value to three times the quantization step $\sigma = \frac{\Delta}{3}$;

however, other values could be used regarding any specific noise model. This method have a best case scenario of accessing to a single table cell and worst case of accessing 16 cells, i.e., 2^4. As we keep the same quantization step as the original method, a relatively lower similarity level correspondence may be retrieved during matching, yet with smaller number and negligible impact on performance.

The second drawback is related with multiple voting and over-representation of similar scene features. This problem is generated when during a scene reference point matching, several different pairs obtain the same combination of model correspondence and quantized α rotation angle. In detail, this happens when similar scene pairs obtain the same model correspondence and they have a similar scene angle value α_s, generating the same quantized α index. Moreover, this situation is worsened by the neighboring checking method. This problem, especially found on planar surfaces, generates multiple superfluous votes for the same LC on the voting table that may produce a deviation in the results. Following the solution of [47], we avoid matching two model pairs with the same combination of quantized PPF index and scene angle α_s. This process is efficiently done by creating an additional 32 bits variable for every PPF quantization index, where each bit represents a quantized value of the scene angle. In this way, when matching a point pair using a PPF, the bit value related to the scene angle is checked. Only if the bit is 0 is the matching allowed and the bit is set to 1, avoiding any new matching with the same exact combination. Notice that the first drawback could be more efficiently solved during training by duplicating the pairs on the neighboring cell. However, in this case, the second drawback will be more difficult to avoid, as keeping track of the same pairs on different cells will request a more complex checking strategy.

3.5 Hypothesis Generation

As explained before, for each scene reference point, all the possible pairs are matched with the model. Then, during hypothesis generation, all consistent correspondence are grouped together generating a candidate pose. In detail, for each obtained scene–model pair correspondence, an LC combination is voted in the two-dimensional voting table. In this way, each position of the table represents an LC, which defines a model pose candidate in the scene, and its value represents the number of supports, which indicates how likely the pose is. The LC α angle is quantized by Δ_{angle} defining a voting table with a total size of $|M| \times \lceil \frac{2\pi}{\Delta_{angle}} \rceil$. After all votes have been cast, the highest value of the table indicates the most likely LC, defining a candidate pose for

this scene reference point. At this step, an important problem arises from the assumption that a local coordinate always exists and, therefore, each piece of scene data has a corresponding model point. In reality, most scenes will have a majority of points that do not belong to the object. In order to avoid generating false positive poses, which can induce bias to the following clustering step, we propose defining a threshold to only consider LC with a minimum number of supports, e.g., three or five votes. Therefore, if the peak of the table is below this number, the pose will be discarded; otherwise, a candidate pose with an associated score is generated.

3.6 Clustering

The matching result of different scene reference points yields multiple candidate poses which may be defining the same model hypothesis pose. In order to joint similar poses together, we propose using a hierarchical complete-linkage clustering method. This clustering approach enforces that all combinations of elements of each cluster follow the same conditions based on two main thresholds, distance and rotation. In practice, we sort the candidate poses by their vote support and create a cluster for each individual pose. Then, all clusters are checked in order and two clusters are joined together when for all combinations of their elements the conditions hold. In this way, the most likely clusters will be merged first, reducing the effect of mutual exclusive combinations. In detail, for two defined thresholds θ and ω, two clusters $\boldsymbol{C_i}, \boldsymbol{C_j} \subset SE(3)$ will be joined if they satisfy the condition:

$$\begin{aligned} max\{dist(P_k, P_l) \mid P_k \in \boldsymbol{C_i}, P_l \in \boldsymbol{C_j}\} < \theta \wedge \\ max\{rot(P_k, P_l) \mid P_k \in \boldsymbol{C_i}, P_l \in \boldsymbol{C_j}\} < \omega, \end{aligned} \tag{3.3}$$

where the binary function $dist : SE(3) \times SE(3) \to \mathbb{R}$ represents the Euclidean distance and the binary function $rot : SE(3) \times SE(3) \to [0, \pi]$ represents the rotation difference between two poses defined by the double arccosine of the inner product of unit quaternions [53]. Finally, for each cluster, all elements are merged and individual scores are summed up to define a new candidate pose.

3.7 Postprocessing

At this point, the method provides a list of candidate poses sorted by the clustered summed score. In order to obtained the best hypothesis pose, these

candidate poses are rescored, refined and filtered through a series of highly efficient postprocessing processes.

3.7.1 Rescoring and Refining

The score of each pose is just an approximation obtained from the sum of each clustered pose number of matching pairs. Due to the nature of the hypothesis generation and clustering steps, joining poses obtained from each table peak, the clustered pose score may not properly represent how well the pose fits the object model to the scene. In this regard, we propose computing a more reliable value through an additional re-scoring processes. This new score will be computed by adding the total number of model points that fit the scene, where a fitting point is a model point closer to a scene point than a threshold. In particular, for a given pose $P \in SE(3)$, the fittings score is computed as shown in Equation (3.4):

$$S_{fitting}(P) = \sum_{m \in \boldsymbol{M}} [min\{||Pm - s|| \mid s \in \boldsymbol{S}\} < th], \tag{3.4}$$

where [] represents the Iverson bracket and th represents the maximum distance threshold. Taking into account the preprocessing of the data, this threshold is set to half of the voxel size. Notice that this re-scoring procedure can be efficiently solved by a Kd-tree structure.

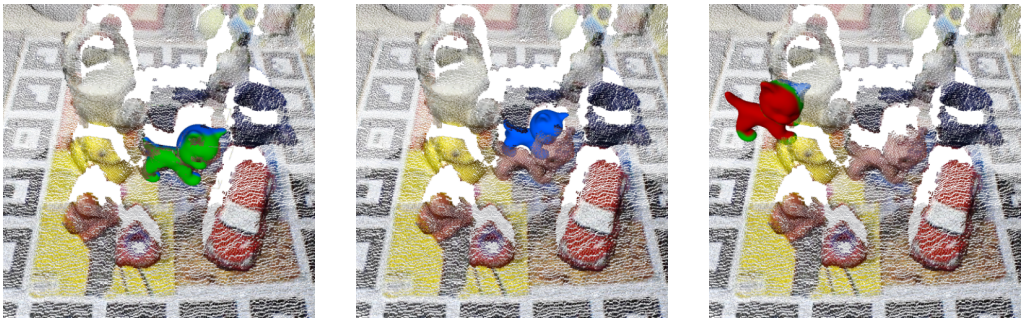
Even though this process provides a better fitting value approximation, there are two important issues that can still reduce the accuracy of the score. First, the deviation produced by model points that are self-occluded in the scene by the camera view, and, second, the possible aligning error of the object model respect to the scene. In order to mitigate these problems, we propose to use an efficient variant of the ICP algorithm alongside a perspective rendering of the object model for each hypothesis pose. For every clustered pose, the model object will be rendered using a virtual camera representing the scene acquisition system. At this point, the rendered data will be downsampled in the same way than the scene data. After that, an efficient point-to-plane ICP algorithm, based on Linear Least-Squares Optimization [67], using projective correspondence [91] will be applied. In this way, the poses are refined and a better fitting score is computed. Despite the efficiency of this process, the large number of hypotheses obtained from the previous steps could significantly affect the whole method performance. A compromise solution is to apply this re-scoring and ICP steps only to the subset of the clustered poses with the higher scores, which represent the more likely fitting poses.

3.7.2 Hypothesis Verification

Based on the ideas proposed by [18, 47, 60], after the re-scoring process, two verification steps are applied to filter false positive cases. In detail, these steps are used to discard well fitting model poses that do not consistently represent the underlying scene data.

Visibility Context Verification

The first verification step checks the model-scene data consistency and discards cases which do not properly match the visibility context of the scene data. From the virtual camera point of view, each point of the rendered view of the model can be classified in three types, regarding its position with respect to the scene data: inlier, occluded and non-consistent. Inlier, shown in Figure 3.4a, is a model point that is near a scene point within a threshold distance and it is considered to match and explain the underlying scene surface. Occluded, shown in Figure 3.4b, is a point that is further away from the scene than a surface inlier; therefore, it is below the scene surface and can not be considered right or wrong. Non-consistent, shown in Figure 3.4c, is a point that is closer to the camera than a surface inlier, which means that it is not explained by the scene data and it is considered wrong. Hypotheses with a big percentage of occluded points or relatively small percentage of non-consistent points are likely to be false positive cases, hence discarded. In order to deal with challenging cases and certain degree of sensor noise, a maximum percentage of 15% of non-consistence points and 90% of occlusion is used.



(a) (b) (c)

Figure 3.4: Model surface points classification regarding its distance to the scene data from the camera view. Green, blue and red colors label inlier, occluded and non-consistent points, respectively on the cat model surface. (**a**) matching pose; (**b**) occluded pose; (**c**) non-consistent pose.

Edge Verification

The second verification step accounts for well fitting poses with non-matching surface boundaries. This checking procedure is especially useful to discard cases relying on planar or homogeneous surfaces without relevant surface characteristics, which can easily be incorrectly fitted to other similar scene surfaces if no boundary considerations are applied. For each hypothesis pose, this step extracts the silhouette of the object model from the camera view, as shown in Figure 3.5a, and compares it with scene extracted edges, Figure 3.5b. The scene edges are extracted by identifying depth and normal variations. The comparison is performed by averaging the distance from each silhouette point to the scene edges. Therefore, having a set of pixels defining the scene edges $\boldsymbol{E_s} \subset \mathbb{Z}^2$, for each different model pose view silhouette, defined by a set of pixels $\boldsymbol{E_m} \subset \mathbb{Z}^2$, the average edge score can be computed as:

$$S_{edge}(\boldsymbol{E_m}, \boldsymbol{E_s}) = \frac{1}{|\boldsymbol{E_m}|} \sum_{e_m \in \boldsymbol{E_m}} min\{||e_m - e_s|| \mid e_s \in \boldsymbol{E_s}\}. \tag{3.5}$$

Poses where the final score is higher than a threshold are discarded. In practice, a threshold of 5 pixels is used as an average distance error.

(a) (b)

Figure 3.5: Examples of the extraction of the object model silhouette and the scene edges. (**a**) object model silhouette; (**b**) scene edges.

Notice that both steps may wrongly discard true positive cases under high occlusion. In this sense, both verification steps represent a trade-off between false positive pruning and occlusion acceptance rate. Hence, a request for high scene consistency, in terms of visibility context and contour matching, will reduce the capability of the system to handle occluded cases in benefit of higher reliability for normal cases.

Chapter 4

Facing Occlusion with Visual Attention and Color Cues

4.1 The Occlusion Problem

Although clutter and occlusion are two of the main challenges faced by object recognition, the robustness of most methods against these cases is unclear. In this direction, most solutions have been tested on datasets with scenes combining different levels of clutter and occlusion, providing only a general picture of the robustness. Even though recent methods show robustness against clutter on highly complex scenes [21, 47], occlusion cases are less clear and seem to be far more challenging. In this line, older results presented by [37, 75, 83] are some of the few published works facing occlusion in detail. In order to obtain a more updated and clear picture of the status of the pose estimation problem, Hodan et al. [49] presented an extensive benckmark for 6D pose estimation where different challenging existing and new datasets where collected, refined and evaluated under a common and fixed procedure. The benchmark were tested for 15 differerent state-of-the-art methods. In this regard, a preliminary work of the method proposed in Chapter 3 was also included in the evaluation. However, despite that most recent methods are robust to illumination changes and clutter, the results shows that 6D pose estimation on occlusion scenarios still remains a challenging case for all methods. In particular, results obtained on the Linemod Occlusion (LM-O) dataset show signs of the weakness of state-of-the-art methods against occluded cases, decreasing the overall recognition results from 88% to 59% when compared to the non-occluded version Linemod (LM) dataset. In more detail, visibility results presented in [49] show that recent method performance obtains less than 10% recognition rates when occlusion levels reach

50% of the object.

In this chapter, we propose to incorporate color information and visual attention principles to boost the performance of a pose estimation method for highly occluded scenarios, such as the ones faces on the LM-O dataset. In detail, we propose to improve the method presented in Chapter 3 by using color information to guide the attention of the method to potential scene zones and improve the surface matching of the method.

4.2 Visual Attention

Visual attention is an important biological mechanism that bases on selecting subsets of the world information to perform a faster and more efficient scene understanding. Inspired by the understanding of the human visual system (HVS) and the development of more efficient intelligent applications, visual attention has been an important research topic in both neuroscience and computer vision fields. Based on the bottom-up and top-down architectures [102], different computer vision methods for visual attention have been presented behind the ideas of salient maps [54], object-based attention [100] and saliency feature vectors [87]. Recently, Potapova et al. [86] presented a survey of visual attention from a 3D point of view, analyzing 3D visual attention for both human and robot vision. Their work reviews most important presented attention computational models, from the widely used contrast-based saliency models [54] to recently proposed Convolutional Neural Networks (CNN) learning approaches [65]. On this line, most research done on visual attention has focused on biological inspired bottom-up attentional mechanisms, where the generalized idea of salient features identification is applied to optimize the application of the limited computational resources to the most attractive elements, regardless of final task or prior knowledge. This pathway, however, does not completely match the requirements of occluded scenarios, where target objects may not necessary be prominent or highly distinguishable attention elements in the scene. Hence top-down mechanisms, where previously known features are identified as salient scene points for potential targets, are considered more suitable. Therefore, following this direction, we propose to integrate an attention mechanism to the method presented in Chapter 3 by using color cues as prior knowledge of the object.

4.3 Color Cues to Improve Matching

Although studies suggest that color contributes to biological object recognition [23], traditionally, color information has been scarcely applied to computer vision recognition approaches. While most of methods relies on shape and texture information as intensity edges [69] and gradients [68, 99], only few cases have considered color information as a prominent feature. Although this situation has been abruptly reversed with the rising of artificial neural networks approaches, for which color information is usually considered, only few deterministic solutions have explicitly used color information for object detection and recognition, such as color SIFT features [107] for 2D vision or CSHOT [103] and VCSH [110] for 3D vision. For the PPF voting approaches, Drost et al. [36] proposed a multimodal variant of the original method [37], defining pairs of oriented 3D points and 2D gradient edges. The proposed method showed a noticeable improvement on performance while showing robustness to light, having the main drawback of a big impact on the runtime performance [49]. In a different direction, Choi et al. [30, 31] proposed to extend the PPF to 10 dimensions, including color information from both points underneath surface. Although showing positive results on some datasets, recent results presented in [59] suggests that the inclusion of the color on the PPF may provide for some cases higher precision results but lower recognition rates. The deterioration of the recognition rates can be attributed to the subjugation of the geometric information to the color information, disregarding valid geometrical matches for non-matching color cases. This characteristic can effect recognition performance for cases where color information is distorted by illumination, modeling artifacts or scene and sensor characteristics, resulting in non-reliable color information. Therefore, following a different approach, we propose to include the color information as a weighting factor for feature matching, increasing the value of a voting pair if the underlying matching points color information is consistent with the geometrical data. In addition, we propose a new rescoring step for the method in order to take in consideration the weighting factors on the fitting score.

4.4 A Novel Solution for Occlusion

In this section, an attention-based approach and color cue weighting solution are integrated into the method presented in Chapter 3. In detail, color information is used to identify a set of salient points that will guide the attention of the pose estimation algorithm, decreasing the complexity of the

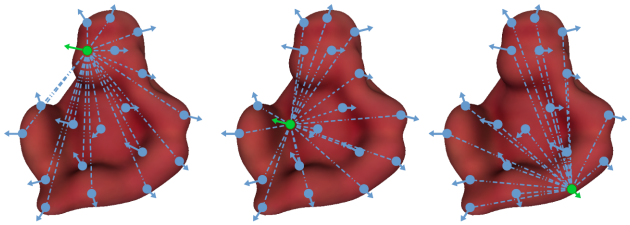


Figure 4.1: The Point Pair Features voting approach globally defines and locally matches the object model as a set of oriented point pairs from each reference point.

global matching problem while increasing the chances of obtaining a positive result. In addition, the color information is used as a weighting factor for the matching of point pairs and re-scoring step to increase the relevance of the color consistence geometrical data.

4.4.1 Attention-Based Matching Using Color Cues

The PPF voting approach is characterized for describing the whole object model as a set of oriented point pairs from each of its points, as show in Fig. 4.1. As explained in Chapter 2, the matching process relies on finding for each scene reference point the best LC, i.e. corresponding model point and rotation angle, that better explains a point pair model definition locally in the scene, i.e. most voted LC in the accumulator. In this sense, only scene reference points that belong to the object model surface will have a valid corresponding model point, and thus a LC. Therefore, other scene points only add superfluous cases which increase processing time and the likelihood of mismatching. From this point of view, the right selection of these references points is an important part of the method performance, which has been underestimated so far. In fact, up to now, most available approaches propose to use a blind-search approach, using all scene points [31, 47] or a fixed fraction of them, usually 1/5th [37, 108].

If we consider a more intuitive human approach, an object can be more efficiently found by focusing attention on zones of the scene that contains elements or features which resemble the ones of the object and can potentially be part of it. Following this reasoning, based on the nature of the method to define and match an object as a set of point pairs from single reference points, we propose to center the attention of those reference points on scene points with similar colors than the object model. In addition, the method needs to consider cases for which those potential zones may not be properly identified and the whole scene should be searched. Hence, we propose two

different strategies: (1) to focus the matching attention on parts of the scene with color information similar to the object; and (2) to search the whole scene at constant space intervals. These two solutions follow a more similar human-like approach for which an object is searched in a scene: initially looking for a similar color and then, in case the color is not a valid feature, searching the full scene at regular space intervals.

In order to identify points of the scene that have similar color than the object and can potentially belong to object surface, we propose to check the color similarity between each of the scene points and the object model. As a single object can have multiple colors on its surface and in different quantities, we only consider those scene points for which their color is found multiple times in the model surface. In this direction, we propose to search, for each scene point, all similar color model points by using a color metric and count the total number of similar points. Then, only those points with a minimum number of matching model color points, which are more likely part of the object, will be considered. In detail, for a given scene point $s \in S$, the set of similar color model points is defined by Eq. 4.1,

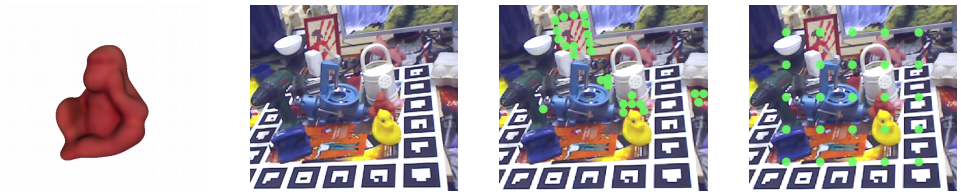
$$C(s) = \{m | d_c(s, m) < \alpha, m \in M\} \tag{4.1}$$

where $d_c()$ is a color distance metric between two points and α is a threshold bounding the similarity level. For a given model object M, the set of a scene reference points used to center the method attention is defined by the cardinality of color matching points as defined by Eq. 4.2,

$$R(S) = \{s| \ \ |C(s)| \geq \beta, s \in S\} \tag{4.2}$$

where β is a threshold bounding the minimum number of color matches for a scene point to be considered. This parameters is in practice fixed to 10 to avoid considering cases with few color matches. Overall, the attention process is defined by a color similarity function, $d_c()$, and two bounding thresholds, α and β. Fig. 4.2 shows a representation of the scene points with high cardinality for the object Ape.

In another direction, a voxel-grid structure is defined to divide the scene at fixed regular distance intervals on the 3 dimensions. These divisions are used to determine an homogeneous distributed set of potential points on the 3D space. Figure 4.2d shows a simplified 2D representation of the concept. In practice we propose to use a voxel size of 10% of the object diameter, in order to ensure that several points lie on the object.



(a) (b) (c) (d)

Figure 4.2: Representation of the attention-based reference points selection method. (**a**) Ape object model; (**b**) Scene containing the Ape object; (**c**) Salient points potentially being part of the Ape object. (**d**) 2D representation of the points distributed at a fixed distance in the scene.

4.4.2 Color Weighted Matching

In addition to rise the attention on potential scene zones, the object model color information can be use to improve the matching process. Choi and Christensen [30, 31] proposed a straight forward approach to use the color information underneath each point pair using the HSV color space to define 10 dimensional features, which include both the geometrical and color data. This solution, however, subordinates the 3D geometrical information to the quality of the color information, and vice versa. This subordination implies the requirement of high quality color models and scene data. Otherwise, the solution can dramatically decrease the method performance on low quality color scenarios produced by the discrepancy and distortion introduced by different sensor properties, illuminations and model creation process. We propose a different solution in which color information is used as a weight factor for geometric data, rewarding those model features that are consistent with the scene in terms of both geometrical and color information. In this direction, a weight value is applied for each LC on the accumulator to increase the value of those poses supported by color consistent point pairs. The weight value for a given scene-model corresponding point pairs, $s_r, s_s \in S$ and $m_r, m_s \in S$, is defined by Eq. 4.3,

$$W_{pp}(s_r, s_s, m_r, m_s) = 1 + W_c(s_r, m_r) \cdot W_c(s_s, m_s) \tag{4.3}$$

and Eq. 4.4,

$$W_c(s, m) = \begin{cases} \omega, & d_c(s, m) < \alpha, \\ 0, & \text{otherwise}, \end{cases} \tag{4.4}$$

where ω is a scalar factor which relates the value of the color information with respect to geometrical data. Notice that the multiplication factor links

the consistency of each point of the pair and the added unit accounts for the basic value of the geometrical matching.

As explained in Chapter 3, the method rescores the clustered candidate poses to compute a better fitting value. Therefore, the proposed weighting for each LC will only effect the two-dimensional corresponding grouping step and color information will not be taken into account after rescoring. In order to address this problem and compute a more robust score, we propose a novel improved rescoring approach which takes into consideration both geometrical and color data. Similar to Eq. 3.4, the fitting score is obtained by summing the model points that have a scene nearest neighbour within a threshold. However, for this case the score value for an object point is compute by adding the inlier maximum distance plus the additive inverse of the point's distance, i.e. euclidean distance between the object point and its nearest scene point. In this way, inliers that are far away from the surface provides lower score. Then, this geometric score is multiplied by one plus the color matching weight, in a similar way than the weighted matching of Eq. 4.3. For a given pose P which transforms the model M to the scene S, the score is computed as defined by the Eq. 4.5,

$$S_{color}(P) = \sum_{m \in \boldsymbol{M}} \begin{cases} (th - ||Pm - s_{nn}||) \cdot (1 + W_c(s_{nn}, m)), & ||Pm - s_{nn}|| < th, \\ 0, & \text{otherwise}, \end{cases} \tag{4.5}$$

where,

$$s_{nn} = \arg\min_{s \in \boldsymbol{S}} \{||Pm - s||\} \tag{4.6}$$

represents the nearest neighbour from a transformed model point to the scene surface and th represents the maximum geometric distance threshold.

The proposed color weight is computed for each voted LC during matching and each model point during rescoring. This requires to compute the color distance metric for the same scene-model combination multiple times, significantly increasing the method's running time. In order to mitigate this problem we propose to precompute the weight for each scene-model point combination in a lookup table. In addition, we propose to find out the attention points and compute the color weight simultaneously. In detail, we propose to create a kd-tree structure with the model color information, therefore we can efficiently obtain all similar colors for each scene point. Then, the similar colors of the model point will be used to compute the weight factors Wc and stored in a table position indexed by the scene and model index.

In this way, the given weight for any scene-model point combination can be found by accessing the lookup table in constant time.

4.4.3 Color Models and Distance Metrics

Color information can be affected by scene conditions (i.e. illumination and shadows), sensor properties (e.g. exposition time, white balance, resolution, etc), and object modeling processes. In this direction, we have taken into account several combinations of most used different color models and metrics to determine the most robust solution.

First, we consider the RGB color space [84], as the most standardized solution. We propose to use the L_2 norm as defined by Eq. 4.7,

$$L_2(s,m) = \sqrt{(R_s - R_m)^2 + (G_s - G_m)^2 + (B_s - B_m)^2}, \tag{4.7}$$

We also consider the HSV/HSL [84] spaces, due to their known illumination invariant properties. Similarly to RGB, we propose to use a variant of the L_2 metric, which takes into consideration the particularities of the Hue dimension of both spaces, this metric L_2Hue is defined by Eq. 4.8

$$\begin{aligned} &L_2Hue(s,m) = \sqrt{\Delta H^2 + \Delta S^2 + \Delta L^2} \\ &\Delta H = min(abs(H_s - H_m), 1 - abs(H_s - H_m)) \\ &\Delta S = S_s - S_m \\ &\Delta L = L_s - L_m \end{aligned} \tag{4.8}$$

Finally, we have also considered the CIELAB color space [40, 61, 74, 84], as a perceptually uniform space with respect to human vision. This color space provides a device-independent color model with respect to a defined white point. Although conceived and mostly used in the industry, this complex color space has been also tested before for other 3D computer vision methods [103]. In this case, the CIE94 delta E distance metric is used as a trade-off between accuracy and speed. In this case, we define the normalized

CIE94 metric as shown in Eq. 4.9,

$$
\begin{aligned}
&CIE94(s,m) = \frac{1}{128}\left[\left(\frac{\Delta L^*}{K_L S_L}\right)^2 + \left(\frac{\Delta C^*_{ab}}{K_C S_C}\right)^2 + \left(\frac{\Delta H^*_{ab}}{K_H S_H}\right)^2\right]^{1/2}, \\
&\Delta L^* = L^*_m - L^*_s \\
&C^*_m = \sqrt{a^{*2}_m + b^{*2}_m} \\
&C^*_s = \sqrt{a^{*2}_s + b^{*2}_s} \\
&\Delta C^*_{ab} = C^*_m - C^*_s \\
&\Delta H^*_{ab} = \sqrt{\Delta a^{*2} + \Delta b^{*2} - \Delta C^{*2}_{ab}} \\
&\Delta a^* = a^*_m - a^*_s \\
&\Delta b^* = b^*_m - b^*_s \\
&S_L = 1 \\
&S_C = 1 + 0.045 C^*_m \\
&S_H = 1 + 0.015 C^*_m
\end{aligned}
\tag{4.9}
$$

where the model point is considered as the standard reference and the parameters are set like graphic arts applications under reference conditions with $K_L = K_C = K_H = 1$. Notice that the LAB color space transformation has been done by using the X, Y and Z tristimulus reference values for a perfect reflecting diffuser, using the standard A illuminant (incandescent lamb) and 2°observer (CIE 1931), reader can refer to [74, 84] for more details about this color space and its metrics.

Chapter 5

Evaluation and Results: Analysis on a Comprehensive Pose Estimation Benchmark

5.1 The BOP Pose Estimation Benchmark

We had the opportunity to collaborate on the evaluation of a comprehensive novel benchmark for 6D object pose estimation. The benckmark, presented in Hodan et al. [49], introduces an standard evaluation procedure, an online platform and the combination of an extensive, variate and challenging sets of new and existing publicity available RGB-D datasets, tested with state-of-the-art methods. The benchmark, shown in Table 5.1, combines eight datasets including 89 object models and 62,155 test images with a total of 110,793 test targets. Each dataset is provided with textured-mapped 3D object models and training images from real or synthetic scenes. Notice that, for our method, only the 3D object model has been used with the texture information. The test images have distinct levels of complexity with occlusion and clutter including different types of objects, from common household objects to industrial-like pieces. For evaluation, the benchmark proposes to use a variation of the Visible Surface Discrepancy (VSD) evaluation metric [48], which is robust against ambiguous cases, explained in [49]. In this regard, all the presented results have been obtained using a misalignment tolerance $\tau = 20$ mm and correctness threshold $\theta = 0.3$. Due to the novelty of the benchmark, authors have published a subset of the original dataset to facilitate the comparison with the state-of-the-art and foster participation to the benchmark, in particular for slow methods. Based on the value of a robust evaluation metric and an extensive set of state-of-the-art results, we

have tested our method on the aforementioned subset.

Dataset	Objects	Training Images		Test Images		Test Targets	
		Real	Synt.	Used	All	Used	All
LM	15	-	1313	3000	18,273	3000	18,273
LM-O	8	-	1313	200	1214	1445	8916
IC-MI	6	-	1313	300	2067	300	2067
IC-BIN	2	-	2377	150	177	200	238
T-LESS	30	1296	2562	2000	10,080	9819	49,805
RU-APC	14	-	2562	1380	5964	1380	5911
TUD-L	3	>11,000	1827	600	23,914	600	23,914
TYO-L	21	-	2562	-	1680	-	1669
Total	89			7450	62,155	16,951	110,793

Table 5.1: BOP benchmark dataset [49]. Each dataset has several objects, training images and test images. Some test images have more than one object, defining several test targets. *Used* values represents the current subsets used for the methods evaluation.

5.2 Method's Step and Parameter Analysis

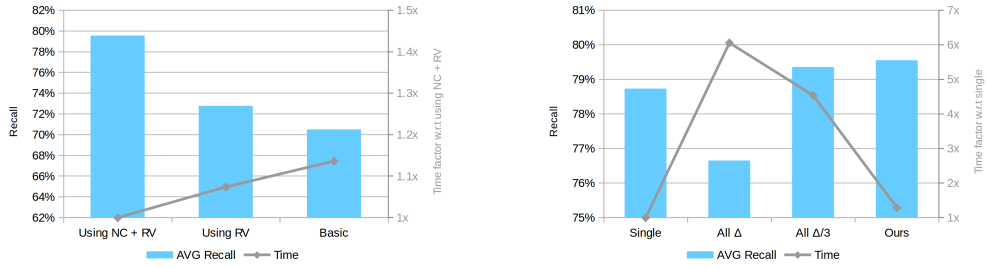
In this section, the effect of different method's steps and parameters are analyzed and compared. The main purpose of this part is to provide a picture of the method's steps relevancy, the parameter dependency and the best color metric for the proposed solution.

5.2.1 Normal Clustering, Matching and Rendered View

Initially, the contribution and value of the normal clustering, rendered view and proposed improved matching are analyzed. These tests has been conducted on the subset of datasets defined by the BOP benchmark [49]. If not explicitly indicated, all cases has been tested with the same parameters. The computational time difference between approaches is provided as a multiplication factor (e.g., two, three or four times slower) with respect to the faster approach in order to draw a more hardware-independent picture of the relation between recognition improvement and time cost.

First, the contribution of the proposed normal clustering downsampling step (NC), alongside the second averaging step, and the appropriateness of

using a model rendered view (RV) for the re-scoring process are evaluated. In order to draw a clear picture of their contribution to the final method result, the two approaches have been disabled and their simpler approaches used. In detail, a common average voxel-grid and a whole model re-scoring process have been used as the basic alternatives. As can be observed in Figure 5.1a, using a rendered view (RV) for re-scoring, reduces the running time and provides a slightly higher recall, probably as a result of estimating a better fitting score using less data. In addition, when this part is combined with the proposed normal clustering (NC) approach for downsampling, the computational time further decreases and a very significant improvement in the results can be observed. Indeed, this result supports our previous reasoning that the preprocessing step is a key part of the method performance.



(a) (b)

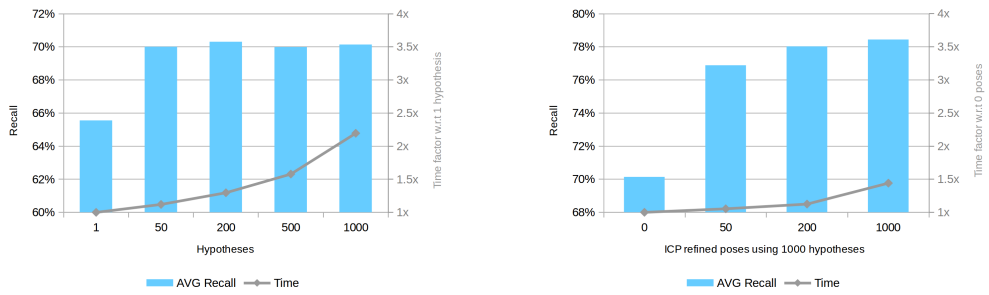
Figure 5.1: Performance comparison between different approaches. (**a**) comparison of the basic method against RV and NC improvements; (**b**) comparison between matching with single cell, all neighbors with Δ and $\frac{\Delta}{3}$ and our method using 16 cells maximum. Note: the left axis shows recall value in percentage and the right axis shows time factor.

Second, the four discussed strategies for decreasing the effect of sensor noise on the quantization feature space are compared. Figure 5.1b shows the recall score for each approach and the time factor with respect to the single cell checking case. The results are surprising in several ways. On the one hand, it can be seen that the contribution of this part, analyzing the overall recall for the tested datasets, is relatively small with less than 1% improvement for all cases. On the other hand, in our implementation, the proposed solution performance goes beyond the designed efficiency and provides, on average, better results than the other approaches with dramatically lower time. Although this effect is irregular for different types of objects and scenes, a plausible explanation for these results can be attributed to the increased correspondence distance between PPF. Indeed, the proposed approach provides a larger distance only for limited cases, effectively only slightly increasing

the overall distance, while avoiding the introduction of many matched pairs with a low similarity level to the voting scheme.

5.2.2 Rescoring and ICP

In this part, the effect of the method postprocessing parameters on the result has been studied by analyzing different cases. These tests have followed the same procedure than previous subsection. First, the effect of considering different number of hypotheses has been studied. Figure 5.2a shows the obtained average recall results for all datasets taking into consideration different number of hypotheses without using any ICP or verification step. As can be seen, the re-scoring process only accounts for a relatively small improvement with respect to the one hypothesis case, which represents the best hypothesis obtained after the clustering step. In addition, it can be observed that the re-scoring process alone does not provide any significant improvement using 50, 200, 500 or 1000 hypotheses. Second, a test for analyzing the ICP effect into the re-scoring process using 1000 hypotheses has been conducted. Notice that the verification steps have not been used. As shown in Figure 5.2b, after poses are refined using ICP, the re-scoring process becomes more effective and performance increases with respect to the number of poses refined. This improvement is slowly decreasing for a higher number of poses, suggesting that, in fact, hypotheses are sorted by their likelihood, as expected. These results also corroborates the value of the ICP step to estimate a more accurate fitting score value.



(a) (b)

Figure 5.2: Performance comparison using different post-processing parameters. (**a**) comparison using a different number of hypotheses, no ICP or verification step is applied; (**b**) comparison refining different number of poses for 1000 hypotheses, any verification step is applied. Note: left axis shows recall value in percentage and right axis shows time factor.

5.2.3 Alpha Value for Different Color Spaces

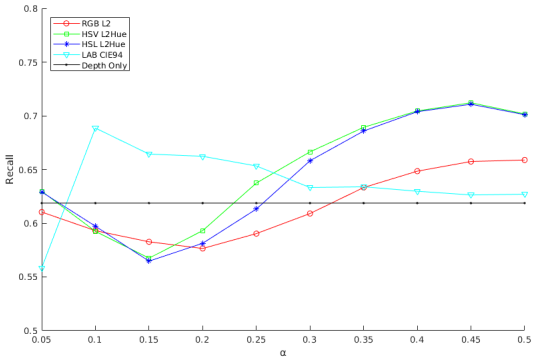
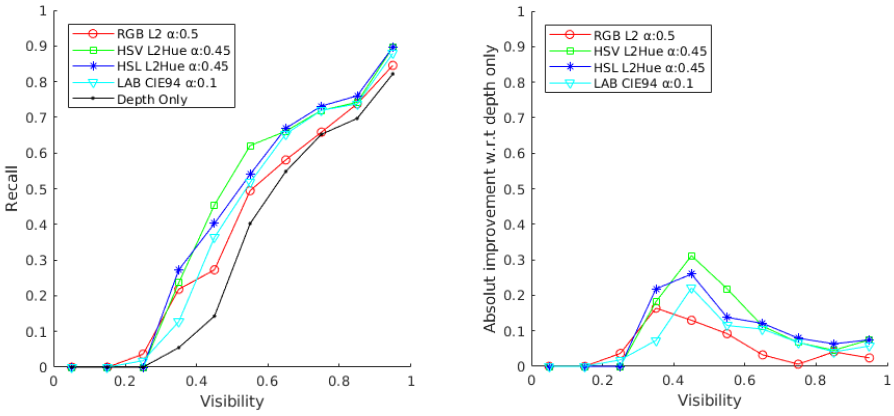


Figure 5.3: Evaluation of different color spaces and metrics combinations with respect to the alpha value.

At this point, the novel color solution presented in Chapter 4 is evaluated for the different discussed color spaces and metrics with respect to the alpha value. For this part, the method has been evaluated for the Linemod Occlusion (LM-O) dataset [22, 46] as a part of the BOP benchmark [49], due to its high occlusion level. Results are compared against the recall values obtained by the basic method of Chapter 3 using depth only. For this test, the omega parameter has been fixed to 5. Results for all four tested color spaces are shown in Fig. 5.3. As can be seen, all tested cases improve the results obtained by using the depth only, obtaining best recognition rates for alphas 0.5, 0.45, 0.45 and 0.1, for the RGB L_2, HSV L_2Hue, HSL L_2Hue and LAB CIE94 cases, respectively. HSV shows a very similar result to HSL, although HSV obtains a slightly better behavior and the highest recognition rate, with an overall recall of 71.21%, providing a very significant improvement with respect to the 61.87% obtained by using depth only. It is interesting to notice the difference between the LAB CIE94 results and the RGB, HSV and HSL colors spaces with the L_2 and L_2Hue metrics, for which the alpha values shows a more stable behaviour around 0.45.

The result in terms of recall and absolute recall improvement with respect to visibility rate are analyzed in Figure 5.4. In detail, the recognition rate for the best alpha, i.e. highest recall, has been plot with respect the visibility percentage of the recognized object. Notice that higher occluded cases have in general less test targets. Recognition rates obtained using the basic method presented in Chapter 3, using depth only, has been included as a reference to compare the improvement obtained for different visibility rates. As can be

seen, there is an improvement of the recognition rates on all cases for visibility levels higher than 20%. In particular, the results show the value of the proposed improvements on occluded cases, with the highest improvements on occlusion level lower than 60%, with improvements of around 20%, 30% and 20% for object with a 30%-40%, 40%-50%, 50%-60% visibility, respectively. The best overall results for all the visibility spectrum are obtained again on the HSV space, although recognition rates of 1.8% for LAB and 3.6% for RGB are obtained for a visibility rate of 20%-30%.



(a) (b)

Figure 5.4: Results obtained using different color and metric cases for the best alpha with respect to the object visibility level. (**a**) Overall recognition rate; (**b**) Absolute improvement rate with respect to [108].

The recall obtained for each object for different levels of visibility are shown in Figure 5.5. As can be observed, most improvements has been localized on relatively low levels of visibility, expanding most object minimum visibility level. Objects Ape, Can, Driller and Eggbox shows an improvement mainly on visibility levels lower than 80%. Overall, Can and Drill are the most robust objects with recognition rates near 100% for 50% occlusion. The improved robustness against occlusion provided by the proposed method can be specially seen on the objects Can and Duck, showing recognition rates of 71% and 50% for 30%-40% visibility, respectively, which represents a tremendous improvement with respect to the original 21% and 0%.

5.2.4 Omega Weight Factor

Following the previous experiments, we have analyzed the effect of the omega parameter on the results for the different color spaces and metric cases. This

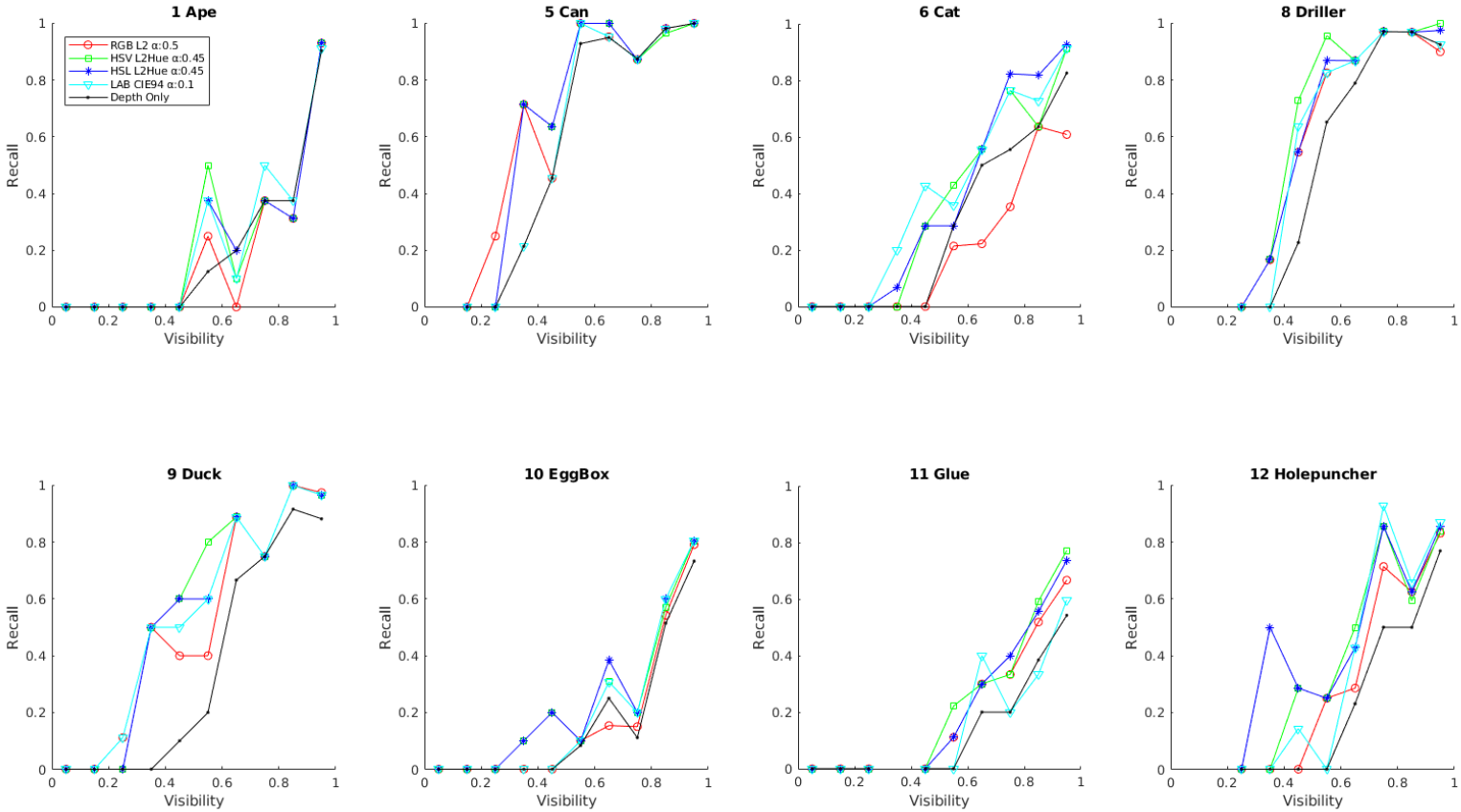


Figure 5.5: Recognition rate for each LM-O dataset object using different color space and metric cases for the best alpha with respect to the object visibility level.

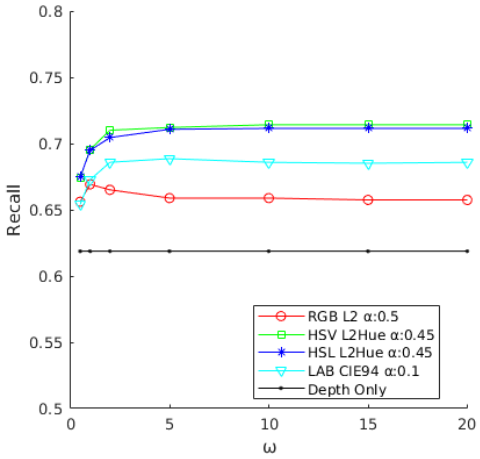


Figure 5.6: Evaluation of omega parameter for the best alpha value.

parameter relates the value of the color information with respect to the geometrical data. For this experiment, the best alpha value obtained for each case has been used. Obtained results are shown in Fig. 5.6. As can be seen, for HSV, HSL and LAB cases, the recall increases for bigger values of omega and quickly saturates for values higher than 5. Therefore, all these cases show a similar behavior and provide the best results for all tested values bigger than 5. On the other hand, the RGB case shows a somehow different behaviour, obtaining the maximum value for 1, while decreasing again, providing also a stable result after 5. This case shows a less stable behaviour providing the best result for 1, decreasing up to 4% for bigger values. In this case, the RGB color space provides low and less stable results.

5.3 Performance Evaluation Using Depth

The proposed method has been evaluated against 14 state-of-the-art methods on the BOP Benchmark. These methods cover all main research branches with local feature-based, template matching and machine learning approaches. For a fair evaluation, all methods have been tested using the same fixed set of parameters for all objects and datasets. Notice that some of the methods also make use of RGB data (e.g., Drost-10-edge, Hodan-15, Branchmann-16 and Kehl-16). Additional details about the evaluation metric and tested methods can be found in [49]. Notice that the results also includes preliminary work of this thesis published in [109], which become the top scoring method and won the SIXD Challenge 2017 competition.

Figure 5.7: Proposed method results in scenes from the BOP benchmark datasets. Scene RGB data is shown in gray. Object models are shown in color and inside a green bounding box. Notice that, for scenes with multiple instances, only the most supported instance is used.

Based on the step and parameter evaluation part, the proposed method has been evaluated using 200 hypotheses refined by ICP. This setup, although not providing the best possible recall, represents a good trade-off between speed and recognition rates. Figure 5.7 shows some examples of the proposed method results. Table 5.3 shows the result of the comparison.

As can be seen from the obtained results, the proposed method outperforms the rest of the solutions, obtaining an average recall of 79.5%. In general, the proposed method performance surpasses the evaluated state-of-the-art with a relative improvement of more than 6% with respect to the preliminary work presented in Vidal-18 [109]. For all datasets, the obtained average recalls show a significant improvement with respect to the state-of-the-art, with a very notable boost on *T-LESS* and *RU-APC*. In particular, for the *RU-APC* case, the proposed method obtains a relative improvement of 19%, moving from 37.83% obtained by Kehl-16 [58] to 44.92%. Overall, the obtained results show higher reliability for different types of objects, including household objects, e.g., *LM* or *TUD-L*, polygon shapes, e.g., *RU-APC*, and industrial-like pieces, e.g., *T-LESS*. In addition, results also suggest higher robustness against occluded scenarios, e.g., *LM-O* and *T-LESS*. Comparing different method types, the proposed method obtains the best recall within the feature-based approaches and outperforms the template matching and machine learning approaches. In detail, compared to the best feature-based approach, the preliminary work in [109], the more-discriminative preprocessing steps, improved re-scoring part and novel clustering approach shows a clear improvement for all datasets. Additionally, the proposed method moves Point Pair Features voting approaches away from the top template matching technique (Hodan-15 [50]), especially for *LM*, *IC-MI* and *RU-APC* datasets. Similarly, the method recall also improves with respect to the top machine learning technique (Brachmann-16 [21]), in particular for the *TUD-L* dataset, for which previously this method had the highest recall. Regarding time performance, the proposed method has an average execution time of 0.99 seconds per target on an Intel i7-5930K. Notice that this performance is obtained without using GPU.

Finally, we would like to notice that conclusions regarding the different methods' performance obtained from the benchmark [49] are significantly different than those of previous results presented in state-of-the-art. In detail, comparing the evaluations presented in [21, 22, 50], results lead to different conclusions, especially regarding the performance of the method proposed by Drost et al. [37], which seems underestimated in those previous cases. We attribute this discrepancy to the improved quality of the benchmark [49] with respect to the previous evaluation procedure, including a fixed training and testing framework with wider object domain and increased number of testing

Method	**LM**	**LM-O**	**IC-MI**	**IC-BIN**	**T-LESS**	**RU-APC**	**TUD-L**	**AVG**
Proposed method	**90.73**	**61.87**	**98.67**	**97.50**	**72.14**	**44.92**	**90.67**	**79.50**
Vidal-18 [109]	87.83	59.31	95.33	96.50	66.51	36.52	80.17	74.60
Drost-10-edge [1]	79.13	54.95	94.00	92.00	67.50	27.17	87.33	71.73
Drost-10 [1, 37]	82.00	55.36	94.33	87.00	56.81	22.25	78.67	68.06
Hodan-15 [50]	87.10	51.42	95.33	90.50	63.18	37.61	45.50	67.23
Brachmann-16 [21]	75.33	52.04	73.33	56.50	17.84	24.35	88.67	55.44
Hodan-15-nr [50]	69.83	34.39	84.67	76.00	62.70	32.39	27.83	55.40
Buch-17-ppfh [25]	56.60	36.96	95.00	75.00	25.10	20.80	68.67	54.02
Kehl-16 [58]	58.20	33.91	65.00	66.00	35.95	37.83	38.67	47.94
Buch-17-si [25]	33.33	20.35	67.33	59.00	13.34	23.12	41.17	36.81
Brachmann-14 [22]	67.60	41.52	78.67	24.00	0.25	30.22	0.00	34.61
Buch-17-ecsad [25]	13.27	9.62	40.67	59.00	7.16	6.59	24.00	22.90
Buch-17-shot [25]	5.97	1.45	43.00	38.50	3.83	0.07	16.67	15.64
Tejani-14 [101]	12.10	4.50	36.33	10.00	0.13	1.52	0.00	9.23
Buch-16-ppfh [26]	8.13	2.28	20.00	2.50	7.81	8.99	0.67	7.20
Buch-16-ecsad [26]	3.70	0.97	3.67	4.00	1.24	2.90	0.17	2.38

Table 5.2: Recall scores (%) for the BOP Benchmark [49] using the VSD metric with $\tau = 20$ mm and $\theta = 0.3$.

targets, fixed parameter requirements and improved evaluation metric. For these reasons, we did not evaluate the presented method against some other related approaches, like [47], which used this previous evaluation procedure.

5.4 Performance Evaluation Using Depth and Color

In this section, the proposed method using visual attention and color information with the HSV and L_2Hue metric is compared with the basic depth only solution presented in Chapter 3 and the other state-of-the-art methods tested in the BOP benchmark [49].

First, the method is evaluated on the LM-O [22, 46] dataset to analyze the performance on a highly occluded dataset in detail. Results are shown in Table 5.3. Examples of obtained results are shown in Fig. 5.8. As can be seen, the proposed method greatly outperform all the methods for all cases. In detail, the inclusion of visual attention principles and color information to the method presented in Chapter 3 improves the recognition rate of all objects, increasing the overall recall 9.34% reaching 71.21%. The biggest improvement has been obtained for object 8, moving from 76% obtained by using depth only to 89%. Similarly, object 12 has obtained an important improvement of 10 points. It is interesting the case of object 11, for which the last best approach was template matching [50], obtaining a boosting from previous best method from 34% to 50%. In addition, the difficulty of the occlusion cases and advantages of the proposed approach can be observed comparing the difference between the results obtained by the other top six approaches, with a range of values moving from 51.42% to 61.87%, while the proposed approach obtains 71.21%. The value of proposed method for occluded cases can also be observed comparing the Drost-10 and Drost-10-edges approaches, for which the inclusion of the edge information do not improve overall this dataset.

Method	**1**	**5**	**6**	**8**	**9**	**10**	**11**	**12**	**ALL**
Proposed - HSV L_2Hue	**69**	**89**	**56**	**89**	**84**	**50**	**50**	**73**	**71.21**
Proposed - Depth Only	66	84	48	76	72	43	34	62	61.87
Vidal-18 [109]	66	81	46	65	73	43	26	64	59.31
Drost-10-edge [1]	47	82	46	75	42	44	36	57	54.95
Drost-10 [1, 37]	62	75	39	70	57	46	26	57	55.36
Brachmann-16 [21]	64	65	44	68	71	3	32	61	52.04
Hodan-15 [50]	54	66	40	26	73	37	44	68	51.42
Brachmann-14 [22]	50	48	27	44	60	6	30	62	41.52
Buch-17-ppfh [25]	59	63	18	35	60	17	5	30	36.96
Hodan-15-nr [50]	47	35	24	12	63	9	32	53	34.39
Kehl-16 [58]	39	47	24	30	48	14	13	49	33.91
Buch-17-si [25]	54	63	11	2	16	9	1	3	20.35
Buch-17-ecsad [25]	29	29	0	0	7	8	1	0	9.62
Tejani-14 [101]	26	2	0	1	0	0	10	0	4.50
Buch-16-ppfh [26]	4	0	0	2	11	1	1	1	2.28
Buch-17-shot [25]	2	7	0	0	1	1	1	0	1.45
Buch-16-ecsad [26]	1	3	0	2	2	0	0	0	0.97

Table 5.3: Recall scores (%) for the Linemode Occlusion dataset [22, 43] as part of the BOP benchmark [49] using the VSD metric with $\tau = 20$ mm and $\theta = 0.3$. Recall score for each individual object and for all the dataset are reported. Objects are numerated as specified in [49].

Figure 5.8: Proposed method results in occluded scenes from the LM-O datasets as part of the BOP Benchmark.

Second, the method robustness is analyzed for all datasets, as shown in Table 5.4. In this case, the results also shows an outstanding improvement on LM and RU-APC datasets along side a good improvements in T-LESS and small improvement in IC-MI. In detail, results for LM dataset are clearly boosted using color data moving from 90.73% obtained with depth only to 93.57% obtained using color data. Significantly better results are also obtained for RU-APC dataset, augmenting from 44.92% to 51.08%. On the other hand, the performance of the T-LESS datasets decrease dramatically with respect to previously obtained results using depth only. The worse results can be attributed to the low quality color information of the model values and the lack of color features on the T-LESS scenes and objects. Overall, the obtained results outperform most cases for which relatively high quality color information is used, highly increasing the recognition rates and showing robustness for most cases. However, the loss of performance on the T-LESS also shows that the method is affected by very low quality color information.

Finally, it is interesting to notice that the method also shows better robustness for the TUD-L dataset, which uses different levels of light intensity,

Method	**LM**	**LM-O**	**IC-MI**	**IC-BIN**	**T-LESS**	**RU-APC**	**TUD-L**	**AVG**
Proposed - HSV L_2Hue	**93.57**	**71.21**	**99.33**	**97.50**	53.55	**51.98**	**91.83**	**79.85**
Proposed - Depth Only	90.73	61.87	98.67	**97.50**	**72.14**	44.92	90.67	79.50
Vidal-18 [109]	87.83	59.31	95.33	96.50	66.51	36.52	80.17	74.60
Drost-10-edge [1]	79.13	54.95	94.00	92.00	67.50	27.17	87.33	71.73
Drost-10 [1, 37]	82.00	55.36	94.33	87.00	56.81	22.25	78.67	68.06
Hodan-15 [50]	87.10	51.42	95.33	90.50	63.18	37.61	45.50	67.23
Brachmann-16 [21]	75.33	52.04	73.33	56.50	17.84	24.35	88.67	55.44
Hodan-15-nr [50]	69.83	34.39	84.67	76.00	62.70	32.39	27.83	55.40
Buch-17-ppfh [25]	56.60	36.96	95.00	75.00	25.10	20.80	68.67	54.02
Kehl-16 [58]	58.20	33.91	65.00	66.00	35.95	37.83	38.67	47.94
Buch-17-si [25]	33.33	20.35	67.33	59.00	13.34	23.12	41.17	36.81
Brachmann-14 [22]	67.60	41.52	78.67	24.00	0.25	30.22	0.00	34.61
Buch-17-ecsad [25]	13.27	9.62	40.67	59.00	7.16	6.59	24.00	22.90
Buch-17-shot [25]	5.97	1.45	43.00	38.50	3.83	0.07	16.67	15.64
Tejani-14 [101]	12.10	4.50	36.33	10.00	0.13	1.52	0.00	9.23
Buch-16-ppfh [26]	8.13	2.28	20.00	2.50	7.81	8.99	0.67	7.20
Buch-16-ecsad [26]	3.70	0.97	3.67	4.00	1.24	2.90	0.17	2.38

Table 5.4: Recall scores (%) for the BOP Benchmark [49] using the VSD metric with $\tau = 20$ mm and $\theta = 0.3$.

showing the robustness of the method against illumination changes. Overall, the results shows the value of the proposed method to significantly improve robustness on most colored scenes, specially for occluded cases, outperforming all tested methods for most datasets.

Chapter 6

Case Study: Automatic Robot Path Integration with Offline Programming and Range Data

In this chapter, the method presented in Chapter 3 is evaluated in depth on a practical scenario. In detail, the presented object recognition method is used to determine the pose of a workpiece element into an offline programming(OLP) platform for automatic robot integration, defining a novel automated offline programming solution (AOLP). This work has been jointly done with Amir Kumar Bedaka, author of the OLP platform. Both authors have equally contribute to the work, Amit worked on the AOLP architecture and the author of this thesis worked on the methodology and vision procedures. The combination of the autonomous object recognition method with Amit's flexible Offline programming platform defines an innovative, more flexible and productive industrial manufacturing solution.

6.1 Introduction

Industrial manufacturing has long relied on human operators to perform challenging and skilled tasks. Certainly, the introduction of industrial robotic systems has dramatically improved the production level, taking over most tasks with a predefined and repetitive nature. Applications such as pick and place, welding, painting or gluing are some of the common jobs carried out nowadays by robots. However, these tasks are usually programmed using conventional methods, like using a teach pendant, in an online programming manner on highly constrained environments. These integration processes require stopping the workcell while expert operators program in situ the

robot actions and trajectories for each specific task, spending a big amount of time and resources on non-flexible solutions with a very limited range of applications.

In view of the time, costs and difficulties associated with the manual online programming and reprogramming of industrial robotic systems in manufacturing scenarios, methods based on CAD model simulation of robot integration arise, known as offline programming (OLP). These software platforms provide all the necessary tools for a complete and realistic simulation of the robot manufacturing environment by using precise CAD model designs. Using these simulation platforms, the robot program can be carefully designed, planned, tested and generated out of the workcell, only requiring a brief stop for the final program download. In [77], Mitsi et al. present an OLP system including graphical simulation, robot kinematics, motion planning and automatic code generation for welding operations. Additionally, Larkin et al. [64] evaluate several OLP software packages used for welding, including ABB Robotics, Delmia from Dassault Systems, a Matlab based OLP system, and RinasWeld from Kranendonk Production System.

In the last decades, a significant amount of research has been done using commercial OLP platforms, such as KUKA Sim for Kuka, RobotStudio for ABB, MotoSim for Motoman, Delmia from Dassault Systems, RobCAD from Technomatix Technologies and Robotmaster from Jabez Technologies [4, 88]. As an example, the OLP system proposed by [112] joins the geometric functions of CATIA (e.g., curve/surface intersection, a projection of the points onto the surface, etc.) with the simulation function of KUKA Sim Pro (e.g., robot kinematics, collision detection, etc.). Their method focused on robotic drilling applications in aerospace manufacturing, improving the position accuracy by using bilinear interpolations model and redundancy resolution. Despite their efficiency, most commercial solutions are subjected to high-cost licenses and a limited range of applications and improvement. Some alternative solutions propose commercial general-purpose CAD packages to define more flexible and cost-effective platforms. Neto et al. explored the most suitable way to represent the robot motion in CAD drawings using Autodesk Inventor, the automatic extraction of the motion data and the mapping between virtual and real environment to generate the robot program [78, 79]. Similarly, some platforms proposed to integrate mechanical CAD features and robotics CAD models with SolidWorks Application Programming Interface (API) [9, 70]. Regardless of their advantages with respect to online programming methods, the OLP platforms highly depend on a precisely defined workcell and still requires a significant amount of human operation, including the selection of tag points (i.e. start and end positions) for path planning and to solve singularity and collision related problems.

Recently, Automated Offline Programming (AOLP) systems are gaining attention in research, providing autonomous alternatives to manual or semi-automatic OLP tasks [82]. These platforms provide significant advantages such as automated modeling of the environment or singularity-free trajectories by means of additional sensors and advanced techniques. Ames et al. [6] develop an AOLP solution to automatically generate complete robot programs without programming requirements to perform welding tasks. Similarly, Polden et al. [85] presents an automatic module for Delmia that provides automatic tag generation and trajectory planning stages for welding applications. Both systems can generate collision-free and singularity-free trajectories for the complete working path. However, the aforementioned AOLP systems still rely on precise CAD geometry of the working environment, defining error-prone and non-flexible solutions that require a high level of human intervention. These systems require a specifically designed environment or manual calibration of the CAD models for each different workpiece and environmental setup, which become not cost-effective in the long run. In this respect, the visual understanding of the environment is an important step towards a more flexible and efficient system.

In the manufacturing and production industries, machine vision has been widely used in different applications [73]. Many types of vision systems and sensors have been proposed to provide reliable solutions according to the particular requirements of individual applications. Some common examples include automatic inspection [81] or robot guidance [35]. In particular, the object recognition problem has been deeply studied towards fully autonomous systems which can work independently of human operators [7]. Solutions based on 2D visual object recognition have been successfully applied for simple pick and place operations or random bin-picking tasks [106]. However, these methods are still affected by environment illumination changes, background clutter, and low robustness. On the other hand, 3D recognition systems based on range data have been proposed, which are robust to illumination and show relatively good results under clutter and occlusion environments [42].

Recently, these intelligent vision-based solutions have advanced on autonomous manufacturing techniques, which do not require robot programming. In this direction, open-source robotics frameworks such as ROS Industrial [2] provide an extensive set of packages for autonomous manufacturing and simulation. However, these autonomous methods face several yet-to-be-solved intrinsic challenges related to real-time decision and sensors data interpretation, such as working range or camera view limitations, which are not faced by OLP systems. These challenges generate severe difficulties to the real applicability of these autonomous solutions [13], which still limit their

application to simple tasks, like pick and place [32, 38], or constrained environments [56]. In this sense, AOLP systems still stand out as a compromise solution for the manufacturing industry.

Vision-based techniques have also been proposed to automate OLP functions. Larking et al. proposed to use Time of Flight (TOF) sensors to map the workcell 3D environment for motion planning without using CAD data [63], effectively avoiding specific-purpose designs and CAD models calibration. However, their system does not provide precise information about the workpiece for automated manufacturing planning operations. In a different direction, Maiolino et al. proposed an AOLP solution for workpiece detection [71]. Their method connects the functions of a commercial offline RobotStudio software, by means of a specifically developed add-on, with an RGB-D recognition module using a UDP socket interface. Their work only analyzed the sensor performance in different illuminations and did not provide results regarding the performance of the system. One of their method's limitations is the nature of the object recognition approach, which requires an isolated object and relies on a segmentation step. This preprocessing step, based on plane filtering, increases the system complexity and may limit its performance. Another crucial limitation comes from the proposed system architecture using a UDP socket communication between the vision module and the commercial software. This add-on solution limits the system applicability and extension capability, making the platform unsuitable for advanced fully integrated tasks and control strategies that are required for complex intelligent robotic manufacturing, such as visual servoing [66] or automatic inspection [12] techniques.

This chapter presents a novel and more flexible AOLP solution that fully integrates a state-of-the-art 6D pose estimation approach into a flexible OLP platform, defining a novel solution that does not require the manual calibration of the workpiece position and can be used or integrated with other systems for advance intelligent manufacturing implementations. In detail, the contribution of this work is a novel AOLP system in an integrated modular architecture, which joints the benefits of latest three-dimensional vision recognition with a versatile OLP platform, proposing a more efficient and flexible solution to overcome the aforementioned limitations. In contrast with other OLP and AOLP approaches, the proposed solution does not require stopping the workcell, complex workpiece calibration procedures and specifically designed or constrained environments to determine the target workpiece in the robot cell. In the one hand, the autonomous three-dimensional vision module determines the position of the object in unconstrained scenes, in a global manner, without requiring pre-segmentation steps. On the other hand, the proposed fully integrated AOLP architecture overcomes the limitations

of previously proposed approaches [72, 78] by allowing our system to be coupled with other advanced intelligent solutions. Some potential applications include high precision tasks using visual servoing [66] and integrated manufacturing process with automatic optical inspection (AOI) [12], which require a fully integrated framework. To archive this characteristics, the proposed system has been based on a non-commercial and cost-effective OLP solution developed on Open Cascade opensource libraries, including an efficient path generation with automated tag creation from CAD primitives. On top of that, the AOLP platform bases its three-dimensional vision capabilities on a highly reliable Point Pair Features (PPF) approach [109] that allows an efficient and robust autonomous localization of the workpiece. The proposed system effectiveness and robustness is evaluated in a real-world environment with two different methodologies. First, the relative error of the system is computed for the X, Y, and Z directions. Second, the absolute error of the system is evaluated for multiple random poses against a human defined ground truth. The method robustness is discussed and analyzed with additional experiments for different illumination and object materials. Finally, the overall system features and precision are compared with other existing methods, which reveals the advantages of the proposed system with respect to the other available solutions. Overall, the presented system defines a novel, flexible and efficient fully integrated platform that focuses on reducing integration time and increasing productivity in manufacturing.

6.2 System Overview

Offline programming (OLP) systems are semi-automated platforms that rely on accurate CAD designs to simulate industrial manipulator manufacturing tasks out of the workcell, in order to avoid costly and time-consuming procedures on the robot production line. This CAD information is usually provided through specific-purpose designs or complex calibrations of the robot environment. In the presented novel automated approach, a depth sensor (Kinect) is employed to extract three-dimensional information of the environment to autonomously locate the workpiece on the workcell, regardless of illumination and background clutter. This object recognition capability is integrated with a user-friendly OCC-based OLP platform, which includes an automatic path planning based on CAD information, in order to plan, simulate, analyze and generate robot control code for manufacturing process on an industrial manipulator (Denso 6556). The architecture of the proposed system is presented in Fig. 6.1.

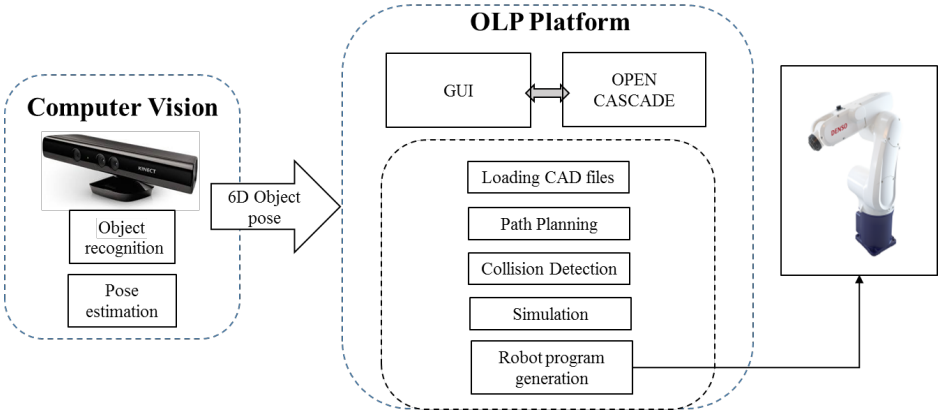


Figure 6.1: Architecture of the proposed platform.

6.2.1 Kinect Sensor

Kinect is an RGB-D sensor, capable of providing registered color and depth data, introduced in 2010 by Microsoft as a game device for the Xbox 360 platform. In 2012 a similar version, named "Kinect for Windows", was released for commercial use.

The sensor is based on infrared (IR) structured-light technology [41]. As shown in Fig. 6.2, the system uses one IR laser projector and two cameras. One camera is used to capture the color image (RGB data) and the other camera, designed to capture only IR light, is used to extract the depth data from the projected IR structured pattern. In addition, the sensor has a microphone array and a tilting motor.

Due to its relatively good precision, considerably high frame rate and low-cost, the sensor quickly became popular in research fields beyond computer entertainment. In addition, the usage of IR light for active sensing, provides depth data independent of visible illumination, working even in absence of light. These characteristics make the Kinect sensor a good choice for the proposed system. Nevertheless, any other range sensor with similar or better characteristics can be employed.

Figure 6.2: Kinect's hardware.

6.2.2 Off-line Programming Platform

The system uses at its core the flexible OLP platform proposed by Amit et al. [10]. This user-friendly OLP platform allows an efficient and automatic path generation, simulation, robot code generation and robot execution. The platform is based on the OCC library, which has become a standard solution for the design and development of open-source application oriented to CAD design and OLP platforms. Using simple mouse interactions, the platform allows to automatically generate robot trajectories by extracting and processing the CAD features of a workpiece with advanced techniques embedded in the OCC libraries, without the need of any commercial CAD packages. In addition, the proposed platform uses a virtual environment and simulates the robot trajectory in order to check issues related to the manipulator's reachability, possible collision along the path and singularities. After simulation, the robot program can be generated and sent directly to an industrial robot manipulator.

6.3 AOLP Integration

The AOLP automatic path planning system has been developed by the integration of the object recognition vision module with the flexible OLP platform. The flow chart of the proposed AOLP platform is shown in Fig. 6.3.

Initially, the OLP core system, which contains the CAD model and environment data of the process, requests the object pose to the vision module, which extracts the position and rotation of the object using the range sensor. This information is processed by the core system, which treats the pose with respect to the vision sensor frame. Consequently, the platform transforms the 6D pose data with respect to the robot frame and loads the relative object CAD model at the same position in the virtual environment. This task is performed autonomously, independently of the object pose and light conditions of the environment. After the recognition of the object pose, the OLP platform is able to generate the targeted path automatically extracting the CAD information.

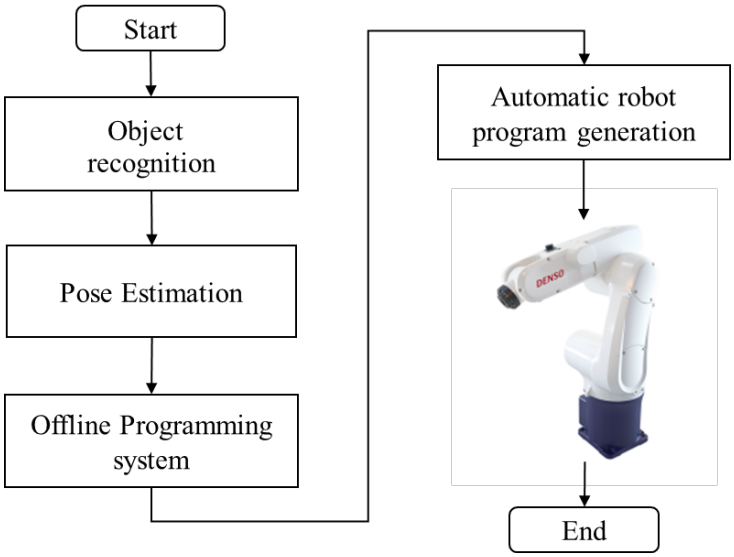


Figure 6.3: Flowchart of the AOLP platform.

6.3.1 Object Recognition

For this system, we propose to use a variation of the method presented in Chapter 3 using depth only. The method includes a Point-to-Plane Iterative Closest Point (ICP) [67] refinement step to ensure the maximum accuracy from the sensor data. In detail, the optimal transformation T_{fit} is defined by Eq. 6.1,

$$T_{fit} = \underset{T}{\operatorname{argmin}} \sum_{i} ((Tm_i - s_i) \cdot n_i) \tag{6.1}$$

where T is the model-to-scene transformation matrix, m_i is a point on the model surface, s_i is the scene destination point and n_i is the unit normal on s_i. For each new iteration, the destination point s_i is defined as the nearest scene point to the last iteration transformed model point m_i.

Overall, this solution defines an accurate and reliable 6D pose estimation module, focused on the requirements of industrial scenes, robust to light and background changes with relative robustness to unexpected partially occluded situations. Figure 6.4 shows a high-level flowchart of the proposed solution, where the *modeling* and *matching* parts are based in the steps presented in Chapter 3.

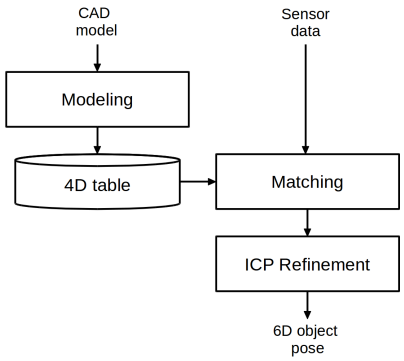


Figure 6.4: Flowchart of the 6D pose estimation module.

6.3.2 Workpiece Transformation

The pose estimation algorithm obtains and transmits precise information of the workpiece pose with respect to range sensor frame. This object pose information is represented as the homogeneous matrix ${}^{C}T_{O} \in SE(3)$, which represents the transformation of the object model to the camera frame, obtained from the three-dimensional recognition module. This transformation is transmitted to the OLP core system, which further transforms the object pose from the camera frame to the robot global frame, as shown in Fig. 6.5. This camera-to-robot transformation, ${}^{R}T_{C} \in SE(3)$, specifies the relative position and rotation of the range sensor with respect to the robot pose, which remains constant for any given industrial manipulator and range sensor fixed setup. This transformation is obtained by a calibration procedure for which a known pattern, i.e. a calibration grid, is attached to the robot end-effector and detected in several positions solving the well-known $AX = XB$ equation [98, 104], analogously to the hand-eye calibration with a fixed camera and a moving calibration grid. In addition, methods using depth data can also be applied. The reader can refer to [34, 51, 98, 104] for a detailed explanation and solutions to the hand-eye calibration problem. Therefore, camera-to-robot transformation ${}^{R}T_{O} \in SE(3)$ is defined by Eq. 6.2.

$$ {}^{R}T_{O} = {}^{R}T_{C}\ {}^{C}T_{O} \tag{6.2} $$

After the object transformation is obtained, the OLP core loads the related CAD model to its actual position in the simulated environment, defining an accurate representation of the robot workcell.

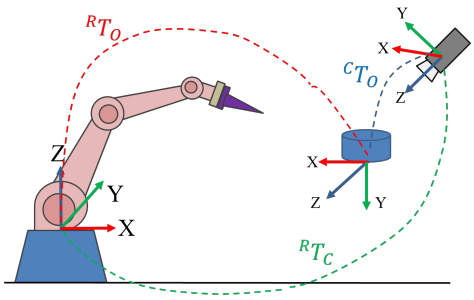


Figure 6.5: Camera and object position with respect to the robot.

6.3.3 Path generation by OLP

Once the CAD model (workpiece) is loaded into the environment, the robot path can be automatically extracted from the CAD information to perform the desired manipulation tasks. Using the approach proposed by Amit et al. [10, 11], the user indicates the working tasks with respect to the CAD model structure using an intuitive and friendly interactive platform. Using a mouse or a tactile screen, the user can automatically extract path references (tags) by selecting desired abstract CAD features, such as a face, wire, edges or vertexes, easily indicating the targeted working zone on the workpiece. For example, the user can select a face of the CAD object and indicate to work on its relative edges in few clicks, as shown in Fig. 6.6. Once, all target zones are selected, the related tag's edges are automatically analyzed and connected to generate a consistent path. Reader can refer to the work of Amit et al. [10, 11] for further details about path generation.

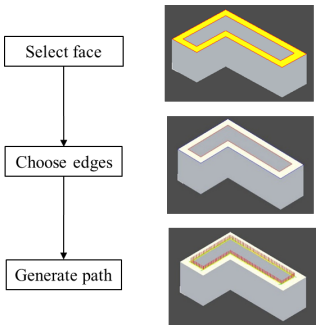


Figure 6.6: Steps to generate the robot path using the OLP platform proposed by Amit et al. [10, 11]

Overall, the proposed AOLP platform allows the user to generate a robot path, simulation, and program mapped with an industrial manipulator with-

out requiring to define the workpiece position. The complete execution process is shown in Fig. 6.7. Initially, the OLP platform sends an acquisition request to the vision module in order to receive the object pose information with respect to the real environment. Consequently, the model is loaded in the same position within the virtual environment. At this point, the user makes use of the platform interface to indicate the desired working path. After these actions, the robot path is automatically generated. The simulation is performed to verify the correctness of all the robot movements before mapping it with the real industrial robot. Finally, the robot program is executed on the real robot performing the desired task.

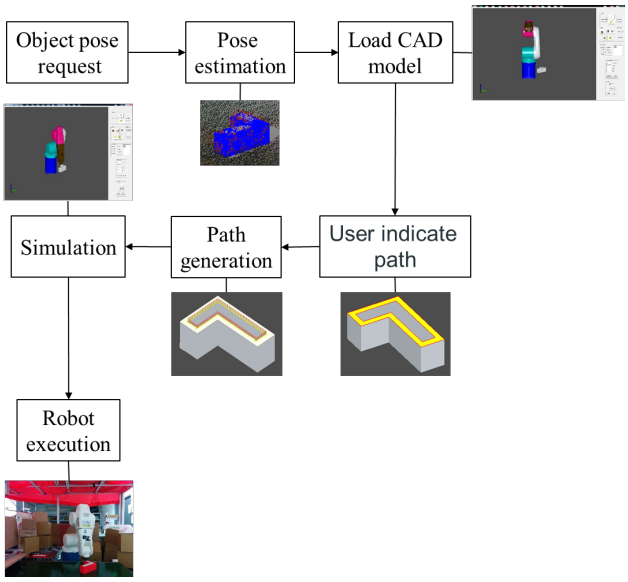


Figure 6.7: System's execution steps.

6.4 Experimental Results

The proposed AOLP system was fully implemented in C++ using OCC [3] and Point Cloud Library (PCL) [94] open-source libraries. The system evaluation is divided into four sets of experiment. The first experiment evaluates the relative error of the system while moving the robot on X, Y, and Z direction. The second experiment evaluates the absolute error against a human defined ground truth. Third and fourth experiments analyze the robustness of the system for different illumination levels and object materials. All experiments have been conducted on a standard setup with the industrial robot manipulator working on a flat surface, as shown in Fig. 6.8. The Kinect

sensor was placed at about 30 to 35 degrees and 100cm away from a Denso 6556 robot, within the best resolution distance and far enough to avoid obstructing the robot working space. Notice that all experiments have been conducted on scenarios with uncontrolled clutter. However, the robustness of the system against background clutter has not been analyzed in these tests as an exhaustive analysis and comparison of the object recognition module in terms of recognition rate, clutter and occlusion performance can be found in Chapter 6. In addition, due to the nobility of the presented AOLP solution, relying on three-dimensional object pose estimation, to the best of our knowledge, no similar AOLP system experimentation results has been presented before. Therefore, we provide a comparison table with different method's features, discussing the strengths and limitations of the proposed method against other available solution for industrial manufacturing.

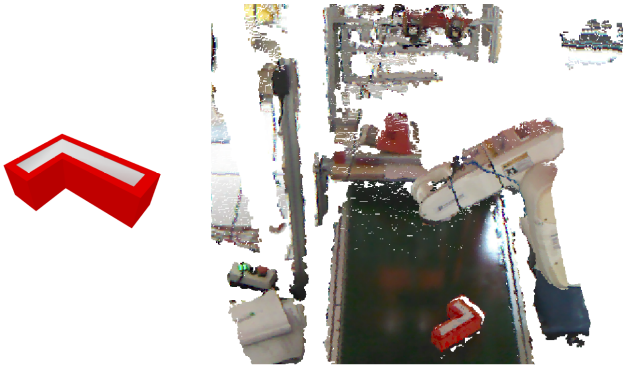


Figure 6.8: (**a**) CAD Model; (**b**) Kinect sensor with CAD model recognized.

6.4.1 Evaluation of the System Error

First, the relative error of the system has been evaluated for X, Y, and Z directions. In this experimentation, the workpiece object has been attached to the end-effector and a fixed point on the workpiece has been selected. Then, repeatedly, the robot manipulator is moved by one fixed step on a given axis, defining the relative error of the system as the difference between the displacement of a workpiece fixed point and the predefined step distance. This test has been conducted for 5mm steps on the three main robot X, Y and Z axis. Fig. 6.9 and Table. 6.1 shows the obtained results. As can be seen, the system provides similar errors for all axis and all tested directions, showing a maximum relative error of $\pm 2mm$. In addition, the system shows a stable performance, with an overall Euclidean error smaller than 2.2mm in all directions.

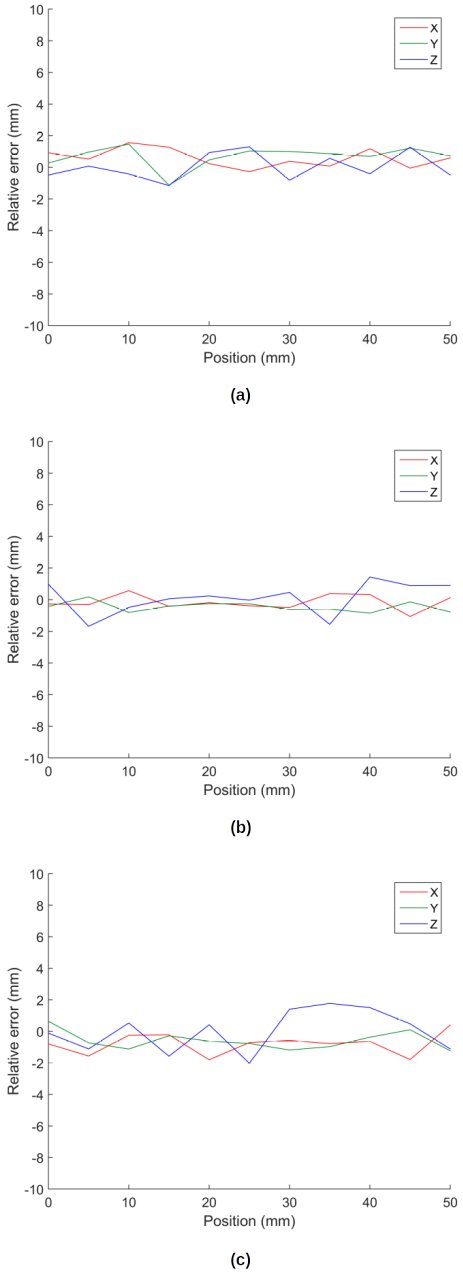


Figure 6.9: Relative error of the system with respect to the industrial manipulator for 5mm steps. (**a**) X-axis; (**b**) Y-axis; (**c**) Z-axis.

Second, the absolute error of the system with respect to a manually defined ground truth has been evaluated. In this experiment, the generated system path for 10 randomly located poses has been compared against a hu-

Table 6.1: Relative error with respect to the industrial manipulator for 5mm steps on X, Y and Z robot axis. Results in mm.

Test	**Mean error**			**Std. deviation**		
Dir.	**X**	**Y**	**Z**	**X**	**Y**	**Z**
X	0.490	-0.159	-0.793	0.676	0.472	0.688
Y	0.690	-0.450	-0.595	0.688	0.318	0.581
Z	0.036	0.108	0.014	0.854	1.012	1.310

man defined ground truth, using a teach pendant. For each pose, 10 different tests have been conducted, with a total of 100 evaluations. The evaluation process is as follows: Initially, for a given random located object, the ground truth path, defined by 4 discrete distinguishable points, was manually found using the teach pendant. After that, the AOLP platform starts the object recognition module and the CAD model is properly located in the OLP platform, where the user can select the path. After the selection, the path is automatically generated from the CAD information and a simulation of the robot motion is performed, allowing the user to check the correctness of the trajectory. Finally, the automatically generated robot program is executed on the Denso industrial manipulator. Then, the system error is computed by comparing the captured ground truth reference points with the corresponding generated path. Figure 6.10 shows the whole process diagram.

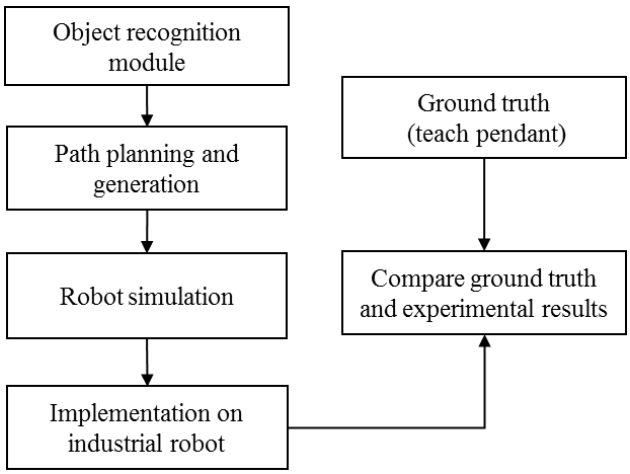


Figure 6.10: Steps to compare the performance of the platform.

Table 6.3: Overall absolute error for all poses, with 10 test per pose using 4 reference points. Results in mm.

Mean error			**Std. deviation**		
X	**Y**	**Z**	**X**	**Y**	**Z**
2.320	2.509	2.297	1.025	1.055	1.055

Table 6.2: Absolute error per pose, with 10 tests per pose using 4 reference points. Results in mm.

Pose	**Mean error**			**Std. deviation**		
	X	**Y**	**Z**	**X**	**Y**	**Z**
1	1.867	1.507	2.805	0.686	1.136	0.796
2	1.987	2.312	2.603	0.869	0.582	0.863
3	2.550	2.355	2.710	1.934	0.491	1.122
4	2.908	1.685	2.358	1.065	1.116	1.160
5	2.474	3.043	2.408	0.562	1.072	0.818
6	2.017	3.317	0.876	0.655	1.253	1.093
7	2.517	3.591	2.136	1.566	0.538	1.348
8	2.944	1.734	2.091	0.467	0.663	0.751
9	1.678	2.696	3.122	0.305	0.880	0.574
10	2.264	2.853	1.866	0.945	0.344	0.625

The obtained results for each pose are presented in Table 6.2. The overall system results for all poses are presented in Table 6.3. Figure 6.11 shows the results for one test of an automatically obtained trajectory on simulation and real-world scenario. Supporting previous experimentation, the obtained absolute error between the system path and human defined ground truth also shows similar results for all axis. Overall the systems show a mean positive error of around 2mm with an std. deviation of 1mm, for the X, Y and Z axis. Similarly, the systems show stable results for all different poses, obtaining consistent precision error for all cases. In this direction, the always positive overall consistent mean error of around 2.4mm, ranging from 0.8mm to 3.6mm for different poses, can be probably attributed to the camera-to-robot calibration.

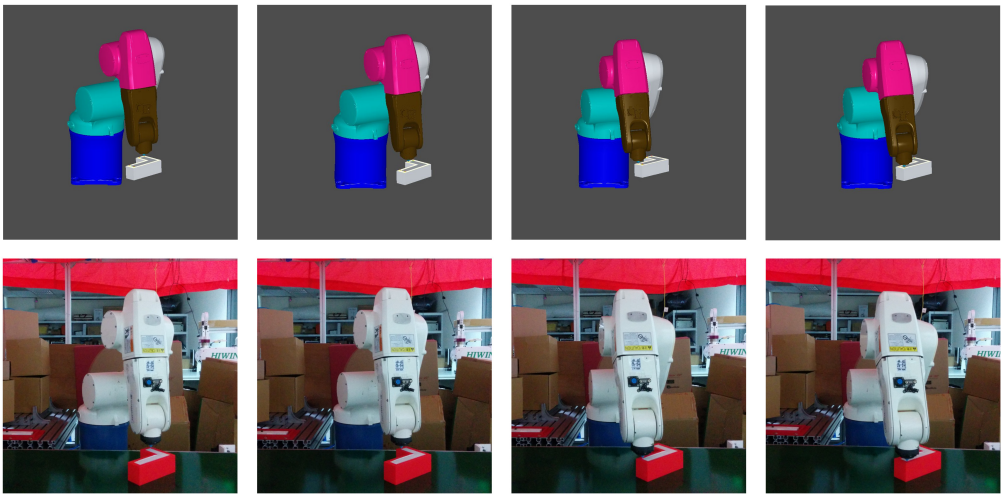


Figure 6.11: Virtual and real-world results for one trajectory generated by the AOLP system.

6.4.2 System Robustness Analysis

The performance and robustness of the three-dimensional vision system capabilities, in terms of recognition rate, rely solely on the vision module and has been already extensively analyzed on a comprehensive range of variate household and industrial objects for more than 60000 challenging test images in Chapter 6. These results, reaching recognition rates up to 100% for non-occluded cases, shows the validity and robustness of the object recognition method for a wide range of different scenarios. In this sense, this characteristic has been widely evaluated on literature joining efforts from different authors, with solid and detailed knowledge available, therefore no further tests have been conducted in this direction.

In another direction, focusing on the proposed integrated AOLP system performance for industrial manufacturing, we conduct a set of experiments to evaluate the precision performance of the integrated system against different illumination levels and different object materials. First, the system was tested under 6 different light conditions, from a highly illuminated environment to a completely dark scene, as shown in Fig. 6.12. In these experiments, no additional modification or parameter tuning has been used, following exactly the same procedures described in the previous section for the absolute error. Experimental results, presented in Fig 6.13, shows the robustness of the proposed system for all different levels of illumination, obtaining consistent results with previous experiments. Although local minor variations occur, no critical drop of the system performance can be observed for different light

conditions. These results show the validity of the proposed integrated system for working on different illumination environments within the same range of precision.

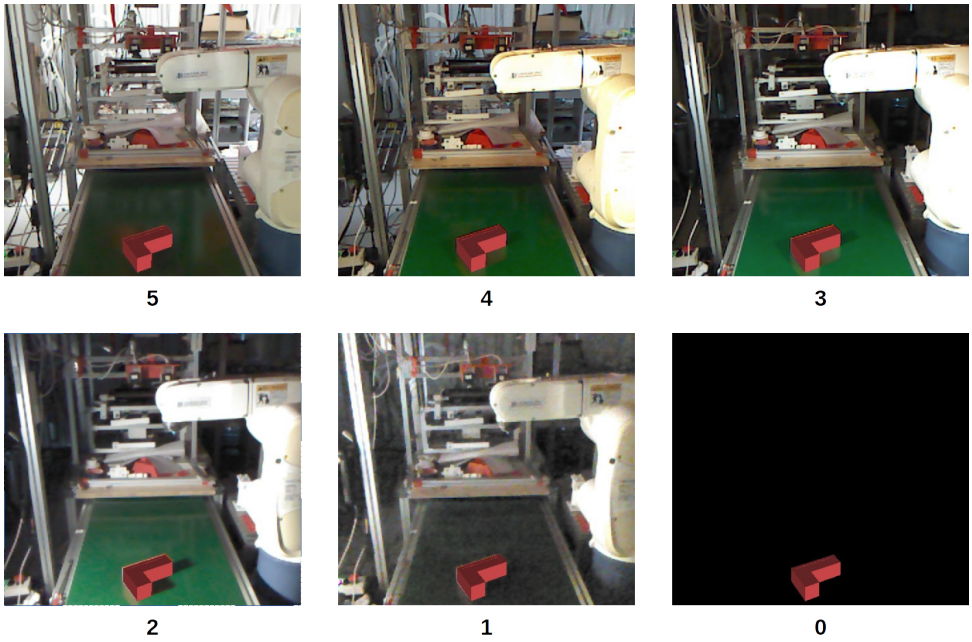


Figure 6.12: Different tested scene illumination levels.

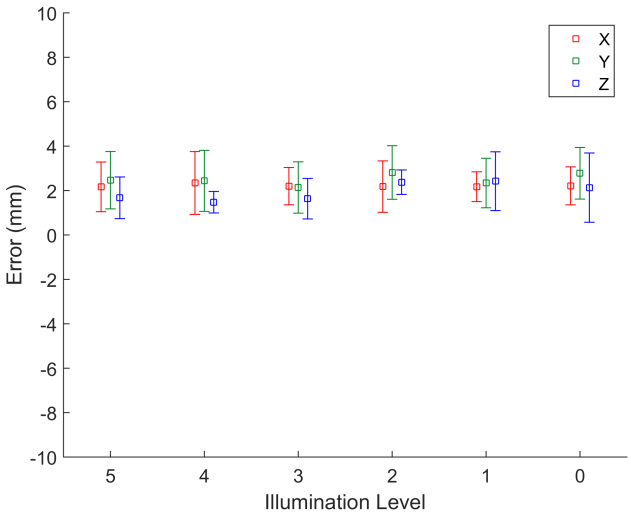


Figure 6.13: System error for different illumination levels.

Second, the performance of the system for different object materials is evaluated. In this experiment the proposed system precision is compared for 4 different objects made of foam, wood, metal and plastic, as shown in Fig. 6.14. The objects where located on the same working position and tested on the same light conditions. Experimental results are presented in Fig. 6.15. As can be seen, the system shows robustness against all 4 different cases with non-critical minor variations between materials. Specifically, we can notice a slightly higher precision on wood and metal than plastic, which can be arguably attributed to their somehow smoother surface. In addition, the foam object shows an slightly higher error, which we attribute to the softness and non-rigidity of the object material. Overall, the obtained results show the robustness for different object materials with consistent precision.

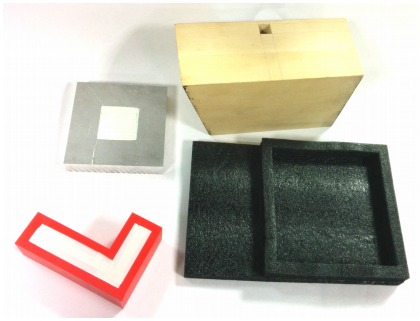


Figure 6.14: Tested objects with different surface materials.

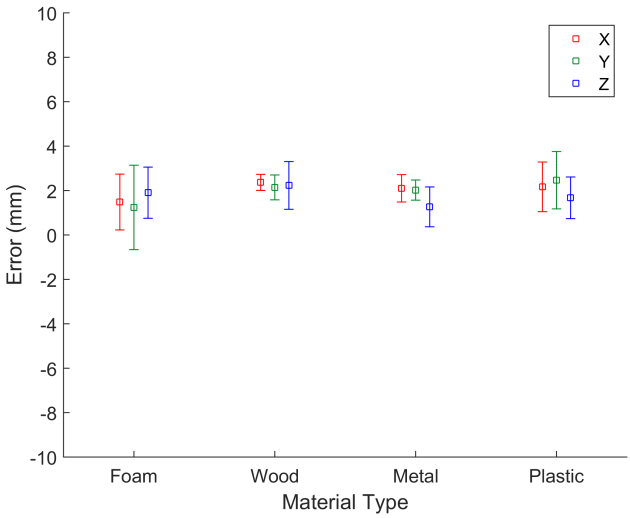


Figure 6.15: System error for objects made by different types of material.

6.4.3 Comparison and Discussion

Increasing demands of productivity on industrial manufacturing require the more precise definition of target workpieces and a higher level of control on manipulation and inspection processes. The challenges defined by tasks of different nature and high precision operations request novel automatic manufacturing approaches that can integrate benefits from several systems and techniques, defining innovative solutions to fulfill the requirements of those complex manufacturing processes. Some examples include; integrated systems with cooperative robotics [27] for highly complex procedures, visual servoing techniques [66] for high precision tasks or novel 3D-based rendering techniques for automated optical inspection (AOI) [12]. These solutions could be employed to boost the productivity of still challenging industrial operations such as insertion, high precision welding, 3D laser-cutting, inspection of catastrophic failure and quality defects and dual arm robotized assembly. In this direction, most available OLP systems, relying on commercial platforms, lack of the necessary characteristics and flexibility required for this type of integration. Although an autonomous system has been proposed as an alternative and optimal solution, at present, their capabilities are limited to simple tasks and they have not yet reached the necessary robustness required for complex manufacturing.

The presented solution proposes a novel approach that integrates the autonomous recognition capabilities of the three-dimensional vision systems with the user-friendly and workcell-free programming advantages from OLP platforms, defining a more productive and flexible solution. In this sense, the proposed system is based on a modular, independent and adaptable OLP platform, allowing to define a fully integrated architecture that can be coupled with other systems and extended its characteristics to face more challenging and complex manufacturing procedures. These characteristics can be in part archived by the implementation of the open CAD technology provided by Open Cascade. Therefore, the proposed system does not only join the autonomous workpiece detection capabilities of the three-dimensional vision but define a platform with an integrated architecture that can be applied and extended to a variety of directions and tasks, increasing productivity by means of a more efficient automatic robot integration.

The proposed method features are compared against other available state-of-the-art approaches for industrial manufacturing on 3D objects in Table 6.4. As can be seen, the proposed method combines most features while still providing a competitive precision. In this sense, solutions using custom automated path generation approaches are only designed to be applied to a limited range of cases. These specific solutions, usually costly and time-

Table 6.4: Comparison table between different methods' features for automatic industrial manufacturing on 3D objects

Method		Maiolino [71]	Caruso [28]	Neto [78]	Rocha [90]	Shah [96]	Our Method
OLP	**Platform**	RobotStudio Software	No	Autodesk Inventor API	No	No	OCC-based
	Auto path from CAD	Yes	-	No	-	-	Yes
	Robot code generation	Yes	-	Yes	-	-	Yes
	Commercial	Yes	-	Yes	-	-	No
Vision system		Kinect v1 Only Depth	Kinect v2 RGB-D	No	Laser triangulation	Basler RGB	Kinect v1 Only Depth
Path generation		OLP	Projected Contour	OLP	No	Edge segmentation	OLP
Object recognition		SHOT/Plane Segmentation	No	No	SVM & Perfect Match	No	Improved PPF
Architecture		Sofware UDP communication	Single module	Standalone Out-of-process	Single module	-	Modular Integration
Extension and Integration		Very Limited UDP/Add-on	Yes	Limited by OLP API	Yes	Yes	Yes
Experimentation		Yes	Yes	Yes	Yes	Yes	Yes
Shown robustness to		Illumination	Different Objects	-	Different Objects	-	Illumination Materials Clutter Diff. Objects
System Precision		-	$< 1.5mm$	-	$< 8mm$	$< 7mm$	$< 4mm$

consuming to program, are based on constrained scenarios and are difficult to change for other purposes. In another direction, OLP methods provide tools to simplify the generation of robotic paths for all types of scenarios. However, these methods have a very limited range of automatic functions, rely usually on non-flexible commercial platforms and require a precise definition of the CAD environment, requiring costly human interventions and specific-purpose designs. In addition, their proposed architectures show limitations regarding their extension and integration into other systems. In order to overcome the problems related to the environment and workpiece definition, methods using object recognition can help to generalize automated approaches to various scenarios. The proposed combination of the OCC-based OLP platform with a state-of-the-art object recognition method by using a modular fully integrated architecture represents a compromise approach to most system limitations defining a more flexible and productive manufacturing solution. In addition, the proposed system has shown robustness in terms of vision recognition rates and overall system precision in different scenarios, showing their values for practical industrial applications.

Overall, the obtained results show the effectiveness and viability of the proposed system, showing consistent results in all experiments and tested scenarios, with a relatively good accuracy for low-demanding precision manipulation tasks. In addition, the flexibility of the integrated architecture allows the system to be easily coupled with more accurate 3D sensors, or integrated with other systems, such as visual servoing [66], to extend its functionality to automatic high precision tasks. On top of that, the intuitive and user-friendly platform allows the user to define the robot path, perform the simulation and generate the robot code with a few simple steps, defining a flexible solution for all types of requirements and manufacturing tasks.

Chapter 7

Conclusions

This thesis proposes and analyses novel solutions based on the top performing Point Pair Features voting approach to define a novel feature-based method for robust recognition and 6D pose estimation of partially occluded objects in cluttered scenes. First, novel preprocessing steps to extract relevant point cloud data, a more efficient feature matching approach, which mitigates quantization errors, and an improved hierarchical clustering step are proposed. In addition, this is complemented with several postprocessing steps, including a novel view-dependent re-scoring process for candidate hypotheses and efficient verifications steps to discard false-positive cases. These processes are presented in an integrated local feature-based pipeline divided into six consecutive steps. Second, an innovative solution based on visual attention and color cues to boost performance on highly occluded cases is proposed. In this direction, a novel visual attention, weighted matching and rescoring steps for the Point Pair Features voting approach using color information are introduced. Third, the method is analyzed for different steps and parameters showing the improvement of the proposed solutions and the robustness of the method. The performance of the method is evaluated against 14 state-of-the-art solutions on comprehensive publicly available benchmark including different types of object in highly cluttered scenes with occlusions. The presented results show that the proposed method using depth only outperforms all the other methods for all datasets, obtaining an overall average recall of 79.5%. In addition, the best result obtained for occluded cases using color information, with the HSV color space and L_2Hue metric, shows the higher robustness of the method, reaching an overall recall of 71.21% for the Linemod occluded dataset. In particular, the method shows an outstanding improvement of up to 30% on relatively low occlusion levels between 30% to 70%. In addition, the method shows higher robustness for most colored datasets, even with illumination changes. However, low colored scenes

and bad colored objects can dramatically decrease the performance of the method. Fourth, a practical case study is presented where the proposed recognition method is integrated with a flexible offline programming platform to define a novel automated offline programming solution for intelligent manufacturing. The real-world testing results and feature comparison with other existing solutions show the validity of the method and its advantages with respect to other state-of-the-art solutions. Overall, the proposed system represents a more flexible, cost-effective and productive alternative to the existing approaches, showing the benefits and potential of the object recognition method, opening the door to highly advanced visually-guided autonomous solutions.

Bibliography

[1] MVTec HALCON. https://www.mvtec.com/halcon/. Accessed: 2018-06-07.

[2] ROS-Industrial. https://rosindustrial.org/. Accessed: 2018-09-10.

[3] OPEN CASCADE. www.opencascade.com, 2017. Accessed 02-12-2018.

[4] Sett A. and K Vollmann. Computer based robot training in a virtual environment. In *IEEE International Conference on Industrial Technology, 2002. IEEE ICIT '02*, volume 2, pages 1185–1189, Dec. 2002.

[5] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. B. Rusu, and G. Bradski. Cad-model recognition and 6dof pose estimation using 3d cues. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 585–592, Nov 2011.

[6] A. L. Ames, E. M. Hinman-Sweeney, and J. M. Sizemore. Automated generation of weld path trajectories. In *(ISATP 2005). The 6th IEEE International Symposium on Assembly and Task Planning: From Nano to Macro Assembly and Manufacturing, 2005.*, pages 182–187, July 2005.

[7] Alexander Andreopoulos and John K. Tsotsos. 50 years of object recognition: Directions forward. *Computer Vision and Image Understanding*, 117(8):827 – 891, 2013.

[8] Farshid Arman and Jake K Aggarwal. Model-based object recognition in dense-range images—a review. *ACM Computing Surveys (CSUR)*, 25(1):5–43, 1993.

[9] Khelifa Baizid, Sasa Cukovic, Jamshed Iqbal, Ali Yousnadj, Ryad Chellali, Amal Meddahi, Goran Devedzic, and Ionut Ghionea. IRoSim: Industrial robotics simulation design planning and optimization platform based on cad and knowledgeware technologies. *Robotics and Computer-Integrated Manufacturing*, 42:121 – 134, 2016.

[10] A. K. Bedaka and C. Y. Lin. Autonomous path generation platform for robot simulation. In *2017 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, pages 63–68, Sept 2017.

[11] A K Bedaka and C-Y Lin. Cad-based robot path planning and simulation using open cascade. *Procedia Computer Science*, 133:779–785, July 2018.

[12] Amit Kumar Bedaka, Alaa M. Mahmoud, Shao-Chun Lee, and Chyi-Yeu Lin. Autonomous robot-guided inspection system based on offline programming and rgb-d model. *Sensors*, 18(11), Nov 2018.

[13] Michael Beetz, Ulrich Klank, Ingo Kresse, Alexis Maldonado, Lorenz Mösenlechner, Dejan Pangercic, Thomas Rühr, and Moritz Tenorth. Robotic roommates making pancakes. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 529–536. IEEE, 2011.

[14] Mohammed Bennamoun and George J Mamic. Introduction. In *Object Recognition*, pages 3–28. Springer, 2002.

[15] Paul J Besl. The free-form surface matching problem. In *Machine vision for three-dimensional scenes*, pages 25–71. Elsevier, 1990.

[16] Paul J. Besl and Ramesh C. Jain. Three-dimensional object recognition. *ACM Comput. Surv.*, 17(1):75–145, March 1985.

[17] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–607. International Society for Optics and Photonics, 1992.

[18] T. Birdal and S. Ilic. Point pair features based object detection and pose estimation revisited. In *2015 International Conference on 3D Vision*, pages 527–535, Oct 2015.

[19] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.

[20] Jonathan Blackledge and Dmitryi Dubovitskiy. Object detection and classification with applications to skin cancer screening. *ISAST Transactions on Intelligent Systems*, 1:34–45, 2008.

[21] E. Brachmann, F. Michel, A. Krull, M. Y. Yang, S. Gumhold, and C. Rother. Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3364–3372, June 2016.

[22] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 536–551, Cham, 2014. Springer International Publishing.

[23] Inês Bramão, Alexandra Reis, Karl Magnus Petersson, and Luís Faísca. The role of color information on object recognition: A review and meta-analysis. *Acta Psychologica*, 138(1):244 – 253, 2011.

[24] Rodney A. Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligence*, 17(1):285 – 348, 1981.

[25] A. G. Buch, L. Kiforenko, and D. Kraft. Rotational subgroup voting and pose clustering for robust 3d object recognition. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4137–4145, Oct 2017.

[26] Anders G. Buch, Henrik G. Petersen, and Norbert Krüger. Local shape feature fusion for improved matching, pose estimation and 3d object recognition. *SpringerPlus*, 5(1):297, Mar 2016.

[27] Fabrizio Caccavale and Masaru Uchiyama. *Cooperative Manipulators*, pages 701–718. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[28] L. Caruso, R. Russo, and S. Savino. Microsoft kinect v2 vision system in a manufacturing application. *Robotics and Computer-Integrated Manufacturing*, 48:174 – 181, 2017.

[29] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145–155, 1992.

[30] C. Choi and H. I. Christensen. 3d pose estimation of daily objects using an rgb-d camera. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3342–3349, Oct 2012.

[31] Changhyun Choi and Henrik I. Christensen. Rgb-d object pose estimation in unstructured environments. *Robotics and Autonomous Systems*, 75:595 – 613, 2016.

[32] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman. Analysis and observations from the first amazon picking challenge. *IEEE Transactions on Automation Science and Engineering*, 15(1):172–188, Jan 2018.

[33] Jian S. Dai. Euler—rodrigues formula variations, quaternion conjugation and intrinsic connections. *Mechanism and Machine Theory*, 92:144 – 152, 2015.

[34] Konstantinos Daniilidis. Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286–298, 1999.

[35] G. N. Desouza and A. C. Kak. Vision for mobile robot navigation: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):237–267, Feb 2002.

[36] B. Drost and S. Ilic. 3d object detection and localization using multimodal point pair features. In *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 9–16, Oct 2012.

[37] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 998–1005, June 2010.

[38] Clemens Eppner, Sebastian Höfer, Rico Jonschkowski, Roberto Martín-Martín, Arne Sieverling, Vincent Wall, and Oliver Brock. Lessons from the amazon picking challenge: Four aspects of building robotic systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pages 4831–4835. AAAI Press, 2017.

[39] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan 2015.

[40] Mark D Fairchild. *Color appearance models*. John Wiley & Sons, 2013.

[41] Jason Geng. Structured-light 3d surface imaging: atutorial. *Adv. Opt. Photon.*, 3(2):128–160, Jun 2011.

[42] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, and J. Wan. 3d object recognition in cluttered scenes with local surface features: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2270–2287, Nov 2014.

[43] S. Hinterstoisser, C. Cagniart, S. Ilic, P. Sturm, N. Navab, P. Fua, and V. Lepetit. Gradient response maps for real-time detection of textureless objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):876–888, May 2012.

[44] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *2011 International Conference on Computer Vision*, pages 858–865, Nov 2011.

[45] S. Hinterstoisser, V. Lepetit, S. Ilic, P. Fua, and N. Navab. Dominant orientation templates for real-time detection of texture-less objects. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2257–2264, June 2010.

[46] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In Kyoung Mu Lee, Yasuyuki Matsushita, James M. Rehg, and Zhanyi Hu, editors, *Computer Vision – ACCV 2012*, pages 548–562, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[47] Stefan Hinterstoisser, Vincent Lepetit, Naresh Rajkumar, and Kurt Konolige. Going further with point pair features. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 834–848, Cham, 2016. Springer International Publishing.

[48] Tomáš Hodaň, Jiří Matas, and Štěpán Obdržálek. On evaluation of 6d object pose estimation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 606–619, Cham, 2016. Springer International Publishing.

[49] Tomas Hodan, Frank Michel, Eric Brachmann, Wadim Kehl, Anders GlentBuch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiri Matas, and Carsten Rother. Bop: Benchmark for 6d object pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[50] Tomáš Hodaň, Xenophon Zabulis, Manolis Lourakis, Štěpán Obdržálek, and Jiří Matas. Detection and fine 3d pose estimation of texture-less objects in rgb-d images. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4421–4428, Sept 2015.

[51] Radu Horaud and Fadi Dornaika. Hand-eye calibration. *The International Journal of Robotics Research*, 14(3):195–210, 1995.

[52] B. K. P. Horn. Extended gaussian images. *Proceedings of the IEEE*, 72(12):1671–1686, Dec 1984.

[53] Du Q. Huynh. Metrics for 3d rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision*, 35(2):155–164, Oct 2009.

[54] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, Nov 1998.

[55] Anil K Jain and Chitra Dorai. 3d object recognition: Representation and matching. *Statistics and Computing*, 10(2):167–182, 2000.

[56] L. Jing, J. Fengshui, and L. En. Rgb-d sensor-based auto path generation method for arc welding robot. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 4390–4395, May 2016.

[57] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1530–1538, Oct 2017.

[58] Wadim Kehl, Fausto Milletari, Federico Tombari, Slobodan Ilic, and Nassir Navab. Deep learning of local rgb-d patches for 3dobject detection and 6d pose estimation. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 205–220, Cham, 2016. Springer International Publishing.

[59] Lilita Kiforenko, Bertram Drost, Federico Tombari, Norbert Krüger, and Anders Glent Buch. A performance evaluation of point pair features. *Computer Vision and Image Understanding*, 166:66–80, 2018.

[60] E. Kim and G. Medioni. 3d object recognition in range images using visibility context. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3800–3807, Sept 2011.

[61] Georg A Klein and Todd Meyrath. *Industrial color physics*, volume 154. Springer, 2010.

[62] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. In *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, pages 3–24, Amsterdam, The Netherlands, The Netherlands, 2007. IOS Press.

[63] N. Larkin, Z. Pan, S. Van Duin, and J. Norrish. 3d mapping using a tof camera for self programming an industrial robot. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 494–499, July 2013.

[64] Nathan Larkin, Aleksandar Milojevic, Zengxi Pan, Joseph Polden, and John Norrish. Offline programming for short batch robotic welding. In *16th Joining of Materials (JOM) conference 2012*, pages 1 – 6, 2011.

[65] X. Li, L. Zhao, L. Wei, M. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang. Deepsaliency: Multi-task deep neural network model for salient object detection. *IEEE Transactions on Image Processing*, 25(8):3919–3930, Aug 2016.

[66] V. Lippiello, B. Siciliano, and L. Villani. Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration. *IEEE Transactions on Robotics*, 23(1):73–86, Feb 2007.

[67] Kok-Lim Low. Linear least-squares optimization for point-to- plane icp surface registration. Technical Report TR04-004, University of North Carolina, 2004.

[68] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157 vol.2, Sept 1999.

[69] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355 – 395, 1987.

[70] Nemer Mahmoud and Konukseven E., Ilhan. Off-line nominal path generation of 6-dof robotic manipulator for edge finishing and inspection processes. *The International Journal of Advanced Manufacturing Technology*, pages 1–12, Dec 2016.

[71] Perla Maiolino, Richard Woolley, David Branson, Panorios Benardos, Atanas Popov, and Svetan Ratchev. Flexible robot sealant dispensing cell using rgb-d sensor and off-line programming. *Robotics and Computer-Integrated Manufacturing*, 48:188 – 195, 2017.

[72] Perla Maiolino, Richard A. J. Woolley, Atanas Popov, and Svetan Ratchev. Structural quality inspection based on a rgb-d sensor: Supporting manual-to-automated assembly operations. *SAE International Journal of Materials and Manufacturing*, 9(1):12–15, 2016.

[73] Elias N Malamas, Euripides G.M Petrakis, Michalis Zervakis, Laurent Petit, and Jean-Didier Legat. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*, 21(2):171 – 188, 2003.

[74] R. McDonald and ed. Roderick. *Colour Physics for Industry*. Society of Dyers and Colourists, 1997.

[75] Ajmal S. Mian, Mohammed Bennamoun, and Robyn Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1584–1601, October 2006.

[76] Ajmal S Mian, Mohammed Bennamoun, and Robyn A Owens. Automatic correspondence for 3d modeling: an extensive review. *International Journal of Shape Modeling*, 11(02):253–291, 2005.

[77] S. Mitsi, K. D. Bouzakis, G. Mansour, D. Sagris, and G. Maliaris. Off-line programming of an industrial robot for manufacturing. *The International Journal of Advanced Manufacturing Technology*, 26(3):262–267, Sept 2004.

[78] Pedro Neto and Nuno Mendes. Direct off-line robot programming via a common cad package. *Robotics and Autonomous Systems*, 61(8):896 – 910, 2013.

[79] Pedro Neto, Nuno Mendes, Ricardo Araújo, J. Norberto Pires, and A. Paulo Moreira. High-level robot programming based on cad: dealing with unpredictable environments. *Industrial Robot: the international journal of robotics research and application*, 39(3):294–303, 2012.

[80] Ramakant Nevatia and Thomas O. Binford. Description and recognition of curved objects. *Artificial Intelligence*, 8(1):77 – 98, 1977.

[81] Timothy S. Newman and Anil K. Jain. A survey of automated visual inspection. *Computer Vision and Image Understanding*, 61(2):231 – 262, 1995.

[82] Z. Pan, J. Polden, N. Larkin, S. V. Duin, and J. Norrish. Recent progress on programming methods for industrial robots. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, pages 1–8, June 2010.

[83] Chavdar Papazov and Darius Burschka. An efficient ransac for 3d object recognition in noisy and occluded scenes. In Ron Kimmel, Reinhard Klette, and Akihiro Sugimoto, editors, *Computer Vision – ACCV 2010*, pages 135–148, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[84] Konstantinos N Plataniotis and Anastasios N Venetsanopoulos. *Color image processing and applications*. Springer Science & Business Media, 2013.

[85] Joseph Polden, Zengxi Pan, Nathan Larkin, Stephen Van Duin, and John Norrish. Offline programming for a complex welding system using delmia automation. In Tzyh-Jong Tarn, Shan-Ben Chen, and Gu Fang, editors, *Robotic Welding, Intelligence and Automation*, pages 341–349, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[86] Ekaterina Potapova, Michael Zillich, and Markus Vincze. Survey of recent advances in 3d visual attention for robotics. *The International Journal of Robotics Research*, 36(11):1159–1176, 2017.

[87] L. Qu, S. He, J. Zhang, J. Tian, Y. Tang, and Q. Yang. Rgbd salient object detection via deep fusion. *IEEE Transactions on Image Processing*, 26(5):2274–2285, May 2017.

[88] Peter K. Radovan H., Daynier R. D. S. and Roman R. Offline programming of an abb robot using imported cad models in the robotstudio software environment. *Applied Mechanics and Materials*, 693,:62–67, Dec 2014.

[89] Lawrence G Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963.

[90] Luís F. Rocha, Marcos Ferreira, V. Santos, and A. Paulo Moreira. Object recognition and pose estimation for industrial applications: A

cascade system. *Robotics and Computer-Integrated Manufacturing*, 30(6):605 – 621, 2014.

[91] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

[92] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[93] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162, Oct 2010.

[94] R. B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). In *2011 IEEE International Conference on Robotics and Automation*, pages 1–4, May 2011.

[95] Giovanna Sansoni, Marco Trebeschi, and Franco Docchio. State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine, and criminal investigation. *Sensors*, 9(1):568–601, 2009.

[96] Hairol Nizam Mohd Shah, Marizan Sulaiman, Ahmad Zaki Shukor, and Zalina Kamis. An experiment of detection and localization in tooth saw shape for butt joint using kuka welding robot. *The International Journal of Advanced Manufacturing Technology*, 97(5):3153–3162, Jul 2018.

[97] Linda G. Shapiro and George C. Stockman. 3d models and matching. In *Computer Vision*. Prentice Hall, Upper Saddle River, NJ, 2001.

[98] Y. C. Shiu and S. Ahmad. Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax=xb. *IEEE Transactions on Robotics and Automation*, 5(1):16–29, Feb 1989.

[99] Carsten Steger. Occlusion, clutter, and illumination invariant object recognition. *International Archives of Photogrammetry and Remote Sensing*, 34(3/A):345–350, 2002.

[100] Yaoru Sun and Robert Fisher. Object-based visual attention for computer vision. *Artificial Intelligence*, 146(1):77 – 123, 2003.

[101] Alykhan Tejani, Danhang Tang, Rigas Kouskouridas, and Tae-Kyun Kim. Latent-class hough forests for 3d object detection and pose estimation. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 462–477, Cham, 2014. Springer International Publishing.

[102] Jan Theeuwes. Top—down and bottom—up control of visual selection. *Acta Psychologica*, 135(2):77 – 99, 2010.

[103] F. Tombari, S. Salti, and L. Di Stefano. A combined texture-shape descriptor for enhanced 3d feature matching. In *2011 18th IEEE International Conference on Image Processing*, pages 809–812, Sept 2011.

[104] R. Y. Tsai and R. K. Lenz. A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 5(3):345–358, Jun 1989.

[105] Shimon Ullman et al. *High-level vision: Object recognition and visual cognition*, volume 2. MIT press Cambridge, MA, 1996.

[106] Markus Ulrich and Carsten Steger. Performance evaluation of 2d object recognition techniques. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(3/A):368–374, 2002.

[107] K. van de Sande, T. Gevers, and C. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, Sept 2010.

[108] Joel Vidal, Chyi-Yeu Lin, Xavier Lladó, and Robert Martí. A method for 6d pose estimation of free-form rigid objects using point pair features on range data. *Sensors*, 18(8), 2018.

[109] Joel Vidal, Chyi-Yeu Lin, and Robert Martí. 6d pose estimation using an improved method based on point pair features. *CoRR*, abs/1802.08516, 2018.

[110] Wei Wang, Lili Chen, Ziyuan Liu, Kolja Kühnlenz, and Darius Burschka. Textured/textureless object recognition and pose estimation using rgb-d image. *Journal of Real-Time Image Processing*, 10(4):667–682, Dec 2015.

[111] W. Wohlkinger and M. Vincze. Ensemble of shape functions for 3d object classification. In *2011 IEEE International Conference on Robotics and Biomimetics*, pages 2987–2992, Dec 2011.

[112] Weidong Zhu, Weiwei Qu, Lianghong Cao, Di Yang, and Yinglin Ke. An off-line programming system for robotic drilling in aerospace manufacturing. *The International Journal of Advanced Manufacturing Technology*, 68(9):2535–2545, Oct 2013.