



**HAL**  
open science

# Détection et estimation de pose d'instances d'objet rigide pour la manipulation robotisée

Romain Brégier

► **To cite this version:**

Romain Brégier. Détection et estimation de pose d'instances d'objet rigide pour la manipulation robotisée. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université Grenoble Alpes, 2018. Français. NNT : 2018GREAM039 . tel-01977050

**HAL Id: tel-01977050**

**<https://inria.hal.science/tel-01977050>**

Submitted on 11 Jan 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## THÈSE

Pour obtenir le grade de

### **DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES**

Spécialité : **Informatique**

Arrêté ministériel : 25 mai 2016

Présentée par

**Romain BRÉGIER**

Thèse dirigée par **James L. CROWLEY**  
et codirigée par **Frédéric DEVERNAY**

préparée au sein du **Laboratoire d'Informatique de Grenoble,**  
**à Inria Grenoble - Rhône-Alpes,**  
et de l'**École Doctorale des Mathématiques,**  
**Sciences et Technologies de l'Information et Informatique**

## **Détection et estimation de pose d'instances d'objet rigide pour la manipulation robotisée**

Thèse soutenue le **11 juin 2018,**  
devant le jury composé de :

**M. Éric MARCHAND**

Professeur à l'Université de Rennes 1, Président

**M. Vincent LEPETIT**

Professeur à l'Université de Bordeaux, Rapporteur

**M. Liming CHEN**

Professeur à l'École Centrale de Lyon, Rapporteur

**M. Frédéric DEVERNAY**

Chargé de recherche à Inria Grenoble - Rhône-Alpes, Examineur

**Mme. Laetitia LEYRIT**

Docteur à Siléane, Examinatrice

**M. James L. CROWLEY**

Professeur à Grenoble INP, Directeur de thèse





## Résumé

La capacité à détecter des objets dans une scène et à estimer leur pose constitue un préalable essentiel à l'automatisation d'un grand nombre de tâches, qu'il s'agisse d'analyser automatiquement une situation, de proposer une expérience de réalité augmentée, ou encore de permettre à un robot d'interagir avec son environnement.

Dans cette thèse, nous nous intéressons à cette problématique à travers le scénario du *dévracage industriel*, dans lequel il convient de détecter des instances d'un objet rigide au sein d'un vrac et d'estimer leur pose – c'est-à-dire leur position et orientation – à des fins de manipulation robotisée. Nous développons pour ce faire une méthode basée sur l'exploitation d'une image de profondeur, procédant par agrégation d'hypothèses générées par un ensemble d'estimateurs locaux au moyen d'une forêt de décision.

La pose d'un objet rigide est usuellement modélisée sous forme d'une transformation rigide 6D dans la littérature. Cette représentation se révèle cependant inadéquate lorsqu'il s'agit de traiter des objets présentant des symétries, pourtant nombreux parmi les objets manufacturés. Afin de contourner ces difficultés, nous introduisons une formulation de la notion de pose compatible avec tout objet rigide physiquement admissible, et munissons l'espace des poses d'une distance quantifiant la longueur du plus petit déplacement entre deux poses. Ces notions fournissent un cadre théorique rigoureux à partir duquel nous développons des outils permettant de manipuler efficacement le concept de pose, et constituent le socle de notre approche du problème du dévracage.

Les standards d'évaluation utilisés dans l'état de l'art souffrant de certaines limitations et n'étant que partiellement adaptés à notre contexte applicatif, nous proposons une méthodologie d'évaluation adaptée à des scènes présentant un nombre variable d'instances d'objet arbitraire, potentiellement occultées. Nous mettons celle-ci en œuvre sur des données synthétiques et réelles, et montrons la viabilité de la méthode proposée, compatible avec les problématiques de temps de cycle, de performance et de simplicité de mise en œuvre du dévracage industriel.

**Mots-clés** Vision par ordinateur, robotique, estimation de pose, dévracage, pose, symétrie,  $SE(3)$ .

## Detection and pose estimation of instances of a rigid object for robotic bin-picking.

### Abstract

Visual object detection and estimation of their poses – *i.e.* position and orientation for a rigid object – is of utmost interest for automatic scene understanding. In this thesis, we address this topic through the *bin-picking* scenario, in which instances of a rigid object have to be automatically detected and localized in bulk, so as to be manipulated by a robot for various industrial tasks such as machine feeding, assembling, packing, etc.

To this aim, we propose a novel method for object detection and pose estimation given an input depth image, based on the aggregation of local predictions through an Hough forest technique, that is suitable with industrial constraints of performance and ease of use.

Overcoming limitations of existing approaches that assume objects not to have any proper symmetries, we develop a theoretical and practical framework enabling us to consider any physical rigid object, thanks to a novel definition of the notion of pose and an associated distance. This framework provides tools to deal with poses efficiently for operations such as pose averaging or neighborhood queries, and is based on rigorous mathematical developments.

Evaluation benchmarks used in the literature are not very representative of our application scenario and suffer from some intrinsic limitations, therefore we formalize a methodology suited for scenes in which many object instances, partially occluded, in arbitrary poses may be considered. We apply this methodology on real and synthetic data, and demonstrate the soundness of our approach compared to the state of the art.

**Keywords** Computer vision, robotics, pose estimation, bin-picking, pose, symmetry,  $SE(3)$ .

# Remerciements

Ce travail n'aurait pu être possible sans le soutien direct ou indirect d'un nombre incalculable de personnes, qu'il m'est impossible de tous évoquer ici.

Je souhaite néanmoins remercier tout particulièrement Frédéric Devernay pour avoir accepté de superviser ma thèse et pour la pertinence de ses conseils, ainsi que Laetitia Leyrit pour son esprit de synthèse et l'investissement dont celle-ci a fait preuve concernant le suivi de mes travaux.

Merci à James L. Crowley pour avoir endossé la responsabilité de la direction de ma thèse ainsi que pour ses conseils d'organisation, et merci à la société Siléane pour avoir accueilli mon projet de thèse CIFRE avec enthousiasme.

Merci à mes collègues de Siléane et de l'équipe Inria IMAGINE pour leur présence et le plaisir de leur conversation, et tout particulièrement à Florian, Matthieu, Amaury, Cyril, Brice, Jean-Louis, Greg, Guillaume, Sandra et Alex avec lesquels j'ai eu la chance d'avoir des échanges privilégiés au cours de ces dernières années.

Un grand merci au système éducatif et scientifique français (et par la même au contribuable) pour avoir financé mes études et une part importante de cette thèse, et m'avoir ainsi permis d'exercer ma curiosité là où tant d'autres n'ont pas cette chance. Merci à l'ensemble des personnels s'investissant dans cette noble tâche, et pour n'en citer que deux, tout particulièrement à messieurs Kloetty et Klopfenstein, enseignants respectifs en collège et lycée, pour avoir consacré quelques heures de leur temps libre à m'initier aux rudiments de l'informatique.

Merci à mes parents et à ma famille pour m'avoir toujours soutenu dans mes choix et pour avoir fait de moi ce que je suis aujourd'hui, je vous dois tout. Merci à Julie pour son soutien sans faille, sa présence et sa bonne humeur, malgré les difficultés.

Enfin, merci aux membres du jury qui ont eu la responsabilité d'évaluer ces travaux, et merci à toi, lecteur potentiel. Si la rédaction d'un manuscrit de thèse est un travail exigeant, je conçois que sa lecture l'est tout autant. Je te souhaite donc bonne lecture, et espère que tu y trouveras quelque intérêt.

## Notations

$a$	Scalaire, noté en caractère maigre.
$A$	Matrice, notée en gras majuscule.
$A^\top$	Transposée de $A$ .
$x$	Vecteur, noté en gras minuscule, représenté dans sa base de référence comme une matrice colonne $(x_1, \dots, x_n)^\top$ avec $n \in \mathbb{N}^*$ .
$A \triangleq B$	Définition de la variable $A$ comme égale à $B$ .
$x \times y$	Produit vectoriel de $x$ et $y$ .
$x \propto y$	Relation de proportionnalité : il existe $\alpha \in \mathbb{R}^*$ tel que $x = \alpha y$ .
$\text{Tr}(A)$	Trace de $A$ .
$\text{vec}(A)$	Vecteur représentant l'ensemble des composantes de la matrice $A$ , ordonnées par colonne, puis par ligne.
$\text{diag}(\alpha_1, \alpha_2, \dots, \alpha_n)$	Matrice diagonale $\begin{pmatrix} \alpha_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \alpha_n \end{pmatrix}$ .
$SO(3)$	Groupe des rotations 3D.
$SE(3)$	Groupe des transformations rigides de l'espace euclidien à 3 dimensions.
$\mathcal{C}$	Espace des poses d'un objet rigide, pouvant être assimilé à $SE(3)$ si ce dernier ne présente pas de symétries propres.
$T = (R, t)$	Transformation rigide, consistant en la composition d'une rotation $R \in SO(3)$ et d'une translation de vecteur $t \in \mathbb{R}^3$ .
$\mathcal{P} \in \mathcal{C}$	Pose d'une instance d'objet rigide.
$[T]$	Pose associée à la transformation rigide $T$ .
$(e_x, e_y, e_z)$	Famille orthonormée de vecteurs définissant un repère 3D direct.
$f, c_u, c_v$	Paramètres intrinsèques d'une caméra suivant le modèle du sténopé, de focale $f$ et de centre optique de coordonnées $(c_u, c_v)$ dans le plan image.
$u = \Pi(x)$	Projeté du point $x$ sur le plan image d'une caméra.
$Z$	Image de profondeur, qui à un pixel $(u, v) \in \mathbb{R}^2$ associe la distance du point correspondant au plan image de la caméra.
$D$	Image de distance, qui à un pixel $(u, v) \in \mathbb{R}^2$ associe la distance du point correspondant au centre optique de la caméra.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>8</b>
1.1	Dévracage industriel robotisé . . . . .	9
1.2	Manipulation robotique . . . . .	10
1.3	Stratégies de dévracage . . . . .	12
1.3.1	Prédiction de prise . . . . .	13
1.3.2	Notion d'instance d'objet . . . . .	14
1.4	Objet de ces travaux . . . . .	17
1.5	Publications . . . . .	18
<b>2</b>	<b>État de l'art</b>	<b>19</b>
2.1	Énoncé du problème . . . . .	20
2.1.1	Définitions . . . . .	20
2.1.2	Reconnaissance d'objets 3D à partir d'une unique vue arbitraire . . . . .	21
2.1.3	Objet de cette thèse : détection et estimation de pose d'instances d'objet . . . . .	22
2.2	Modalités d'acquisition . . . . .	23
2.3	Données d'apprentissage . . . . .	27
2.4	Approches globales . . . . .	30
2.4.1	Recherche de patrons . . . . .	31
2.4.2	Détection et estimation de pose . . . . .	35
2.4.3	Fond et occultations . . . . .	39
2.5	Approches locales . . . . .	40
2.5.1	Recherche de sous-graphes de primitives . . . . .	40
2.5.2	Appariement de points d'intérêt . . . . .	42
2.5.3	Aggrégation d'hypothèses . . . . .	45
2.6	Raffinement et vérification d'hypothèses . . . . .	48
2.7	Gestion des symétries . . . . .	50
2.8	Synthèse et pistes de recherche . . . . .	51
<b>3</b>	<b>Notion de pose d'objet, et distance associée</b>	<b>55</b>
3.1	Introduction . . . . .	56
3.2	Définition de la pose . . . . .	58
3.2.1	Lien entre l'espace de pose et $SE(3)$ . . . . .	58
3.2.2	Pose comme classe d'équivalence de $SE(3)$ . . . . .	58
3.2.3	Le groupe des symétries propres . . . . .	59
3.3	État de l'art relatif aux distances sur l'espace de pose . . . . .	60
3.3.1	Objectivité . . . . .	61



3.3.2	Approximation par des hyper-rotations . . . . .	61
3.3.3	Décomposition en position et orientation . . . . .	61
3.3.4	Approches géométriques . . . . .	63
3.3.5	Paramétrisation locale . . . . .	64
3.4	Distance proposée . . . . .	64
3.4.1	Définition de la distance entre poses . . . . .	64
3.4.2	Objectivité . . . . .	65
3.4.3	Interprétation géométrique . . . . .	65
3.4.4	Anisotropie de rotation . . . . .	67
3.5	Calculs efficaces de distance . . . . .	68
3.5.1	Recherche de voisinage . . . . .	69
3.5.2	Décomposition en translation et orientation . . . . .	72
3.5.3	Objet sans symétrie propre . . . . .	72
3.5.4	Objet de révolution sans invariance par rotoréflexion . . . . .	73
3.5.5	Objet sphérique . . . . .	74
3.5.6	Objet de révolution avec invariance par rotoréflexion . . . . .	75
3.5.7	Objet présentant un nombre fini de symétries propres . . . . .	76
3.5.8	Objet 2D . . . . .	76
3.6	Symétries des représentants . . . . .	77
3.7	Projection sur l'espace de pose . . . . .	80
3.7.1	Objet sphérique . . . . .	80
3.7.2	Objet de révolution . . . . .	80
3.7.3	Objet présentant un nombre fini de symétries propres . . . . .	81
3.7.4	Objet 2D . . . . .	81
3.8	Moyenne de poses . . . . .	82
3.8.1	Objets admettant un unique représentant par pose . . . . .	83
3.8.2	Objets admettant plusieurs représentants par pose . . . . .	84
3.8.3	Conditions suffisantes de cohérence d'un n-uplet . . . . .	88
3.9	Propriétés locales de la distance proposée . . . . .	90
3.10	Exemple applicatif . . . . .	92
3.10.1	Détection et estimation de poses via l'algorithme Mean Shift . . . . .	92
3.10.2	Comparison avec une métrique classique de $SE(3)$ . . . . .	96
3.11	Synthèse . . . . .	98
3.11.1	Résumé pratique des principaux résultats . . . . .	98
3.11.2	Conclusion . . . . .	103
<b>4</b>	<b>Solution proposée</b> . . . . .	<b>104</b>
4.1	Régression probabiliste locale . . . . .	105
4.1.1	Points de référence . . . . .	107
4.1.2	Formulation probabiliste . . . . .	110
4.2	Forêt de Hough . . . . .	111
4.2.1	Classifieur faible . . . . .	112
4.2.2	Aggrégation des votes . . . . .	119
4.3	Extraction d'hypothèses de pose . . . . .	120
4.4	Raffinement d'hypothèses . . . . .	120
4.5	Vérification et filtrage . . . . .	123
4.5.1	Qualité intrinsèque d'une hypothèse . . . . .	123
4.5.2	Cohérence globale . . . . .	128
4.6	Apprentissage d'un arbre de décision . . . . .	128

4.6.1	Données d'apprentissage . . . . .	130
4.6.2	Distribution a priori de pose . . . . .	131
4.6.3	Procédure d'apprentissage . . . . .	132
4.6.4	Critère de sélection d'un classifieur faible . . . . .	134
4.6.5	Représentation synthétique d'un apprentissage . . . . .	139
4.7	Synthèse . . . . .	140
<b>5</b>	<b>Évaluation expérimentale</b>	<b>141</b>
5.1	Méthodologie d'évaluation . . . . .	142
5.1.1	Les approches d'évaluation existantes et leurs limitations . . . . .	142
5.1.2	Génération de jeux de données annotés . . . . .	147
5.1.3	Quantification des performances . . . . .	152
5.2	Expérimentations sur différents jeux de données de vrac . . . . .	160
5.2.1	Analyse des performances de notre approche . . . . .	160
5.2.2	Cas d'échecs . . . . .	167
5.2.3	Comparaison avec deux méthodes de référence . . . . .	169
5.2.4	Prise en compte des symétries . . . . .	174
5.2.5	Pertinence de données de synthèse pour l'évaluation . . . . .	174
5.3	Expérimentations sur le jeu de données LINEMOD . . . . .	178
5.4	Caractérisation de notre forêt de décision . . . . .	182
5.4.1	Importance de la silhouette dans la détection et l'estimation de pose . . . . .	183
5.4.2	Effort d'apprentissage . . . . .	184
5.4.3	Profondeur et nombre d'arbres . . . . .	187
5.5	Influence de l'occultation . . . . .	189
5.6	Robustesse au bruit . . . . .	190
5.7	Synthèse . . . . .	192
<b>6</b>	<b>Conclusion</b>	<b>196</b>
6.1	Résumé . . . . .	196
6.2	Contributions . . . . .	197
	<b>Annexes</b>	<b>199</b>
A	Méthodes de calcul pour un maillage triangulaire . . . . .	199
B	Simplification de l'expression de la distance proposée dans le cas d'un objet de révolution sans invariance par rotoréflexion . . . . .	200
C	Condition de cohérence pour un doublet de représentants . . . . .	201
D	Détail de calcul : distance minimale entre représentants . . . . .	202
E	Robustesse au bruit . . . . .	203

# Chapitre 1

## Introduction

---

1.1 Dévracage industriel robotisé . . . . .	9
1.2 Manipulation robotique . . . . .	10
1.3 Stratégies de dévracage . . . . .	12
1.4 Objet de ces travaux . . . . .	17
1.5 Publications . . . . .	18

---

L'accès à une puissance de calcul et une masse de données toujours plus importante est à l'origine d'une explosion des applications en vision par ordinateur, en robotique et en intelligence artificielle. Si les progrès techniques sont impressionnants, ceux-ci sont également sources de questionnements philosophiques d'importance. La collecte massive de données personnelles et le pouvoir d'influence toujours grandissant d'une informatique se voulant ubiquitaire laisse entrevoir le risque d'un futur orwellien, tandis que la possibilité de remplacer un humain par une machine dans l'exécution de son travail, y compris pour des tâches de haut niveau comme le diagnostic médical, pose la question de la place de l'homme et de son utilité dans la société.

Nos travaux s'inscrivent pourtant dans cette tendance, et ont pour but de remplacer l'humain par une machine dans l'exécution de son travail. Plus précisément, ceux-ci portent sur le domaine de la robotique industrielle et visent à automatiser les opérations de *dévracage* (ou *bin-picking*), qui consistent à manipuler des objets initialement en vrac de manière à en faire quelque chose d'utile. À notre crédit, il s'agit là d'une tâche pénible, ingrate et fastidieuse, réalisée à la chaîne. Sa robotisation est donc a priori bénéfique pour les individus, pourvu d'une juste redistribution des richesses découlant de cette automatisation.

Dans cette thèse, nous concentrons nos efforts sur un aspect particulier de cette tâche, qui consiste en la détection et l'estimation de pose d'instances d'objets au sein d'une scène de vrac, et dont le présent chapitre introduit le contexte.

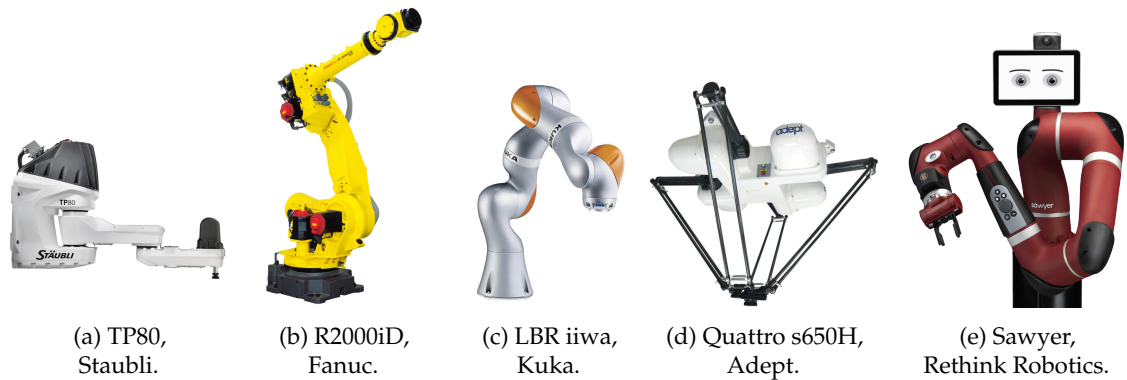


FIGURE 1.1 – Exemples de robots industriels.

## 1.1 Dévracage industriel robotisé

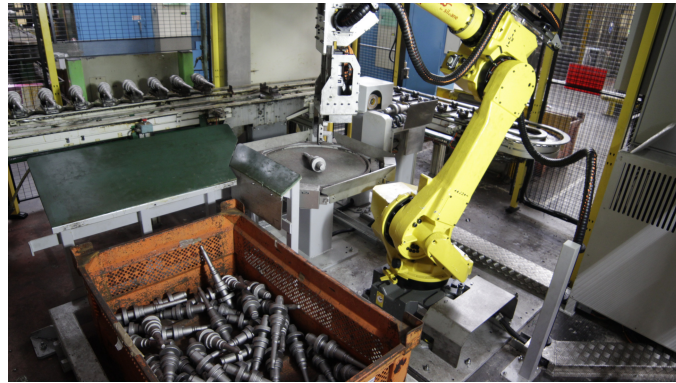
**Robot industriel** Dans le milieu industriel, on considère comme robot un système manipulateur programmable, à l’instar de ceux illustrés figure 1.1. Les robots permettent de remplacer des opérateurs humains dans l’exécution automatique de tâches répétitives, pénibles, dangereuses ou tout bonnement humainement irréalisables pour des raisons de temps de cycle, d’hostilité de l’environnement, ou simplement hors de portée de nos capacités physiques.

**Dévracage** Encore aujourd’hui, la robotique industrielle est majoritairement développée dans des environnements très maîtrisés et peu changeants, où il s’agit de répéter sans cesse des opérations identiques, comme par exemple souder des éléments toujours positionnés de manière similaire sur une chaîne automobile. Cette approche est rendue possible par la capacité à maîtriser les principales sources de variabilité au sein d’une chaîne de production : la luminosité peut être réglée par l’utilisation d’un éclairage artificiel, la température par l’usage d’un thermostat, la position d’une pièce au moyen de brides adaptées, etc. La maîtrise de l’environnement ne peut néanmoins pas être totale, et l’industriel se trouve notamment tributaire de la variabilité du flux d’entrée de sa chaîne de production. Dans le cas où ce flux est constitué d’objets solides – comme par exemple des pièces brutes qu’il s’agit d’usiner, d’assembler, d’emballer, etc.–, ces objets sont souvent livrés dans des configurations variables, voire complètement en vrac. La problématique de chargement de machine consiste alors à dépasser cette variabilité, en séparant les produits et en les mettant en position afin de pouvoir par la suite réaliser les opérations souhaitées. C’est ce qu’on appellera dans cette thèse le *dévracage*.

Si le dévracage peut être réalisé à la main par un opérateur, il s’agit cependant d’une tâche fastidieuse et physiquement éprouvante. Certains dispositifs mécaniques tels que le bol vibrant figure 1.2a permettent donc d’automatiser ce procédé, en séparant, positionnant et distribuant unitairement des pièces initialement en vrac au moyen d’un astucieux système de vibration. De telles machines sont cependant relativement spécifiques, et peuvent ne pas être envisageables dans le cas d’objets lourds, fragiles, ou



(a) Solution mécanique de dévracage : le bol vibrant (AVITEQ).



(b) Robot industriel dévracant des pièces de fonderie afin de les positionner en entrée d'une chaîne d'usinage (Siléane).

FIGURE 1.2 – Dévracage industriel.

encore ne présentant pas une forme adéquate.

**Dévracage robotisé** L'état de la technique et des connaissances permet néanmoins de dépasser pour partie cette limitation en confiant la réalisation du dévracage à un robot autonome doté d'une certaine dose d'intelligence lui permettant d'être adaptable à des configurations voire des objets divers, tel que celui illustré figure 1.2b. Cette capacité d'adaptation est essentielle dans un contexte industriel qui s'oriente toujours plus vers une production « agile » de petites et moyennes séries personnalisées; et c'est cette problématique du dévracage robotisé qui constitue la visée applicative de nos travaux.

L'idée de confier à un robot le soin de manipuler automatiquement des pièces initialement en vrac n'est pas nouvelle, et déjà en 1983 [Horn et Ikeuchi \(1983\)](#) écrivaient sur ce sujet, connu sous le nom de *bin-picking* dans le monde anglophone. Malgré les beaux succès obtenus depuis, maquettes expérimentales à l'appui, celui-ci reparait régulièrement dans la littérature scientifique et commence seulement à se développer dans l'industrie. Nous voyons là une preuve qu'il y a encore matière à progrès dans ce domaine, d'où l'intérêt de cette thèse, mais également le symptôme d'un défaut dans l'analyse et l'évaluation des progrès accomplis.

## 1.2 Manipulation robotique

**Préhension** Afin d'interagir avec leur environnement, les robots sont typiquement dotés d'organes de préhension et de manipulation. Si dans l'industrie des dispositifs tels que des pinces ou des ventouses sont largement utilisés de par leur simplicité et leur robustesse, de nombreux autres effecteurs ont également été envisagés selon les applications, tels que ceux illustrés figure 1.3. L'exécution de tâches de manipulation avancée peut être facilitée par l'usage d'organes présentant de nombreux degrés de liberté comme la main du robot iCub figure 1.3c qui présente 20 articulations. Le

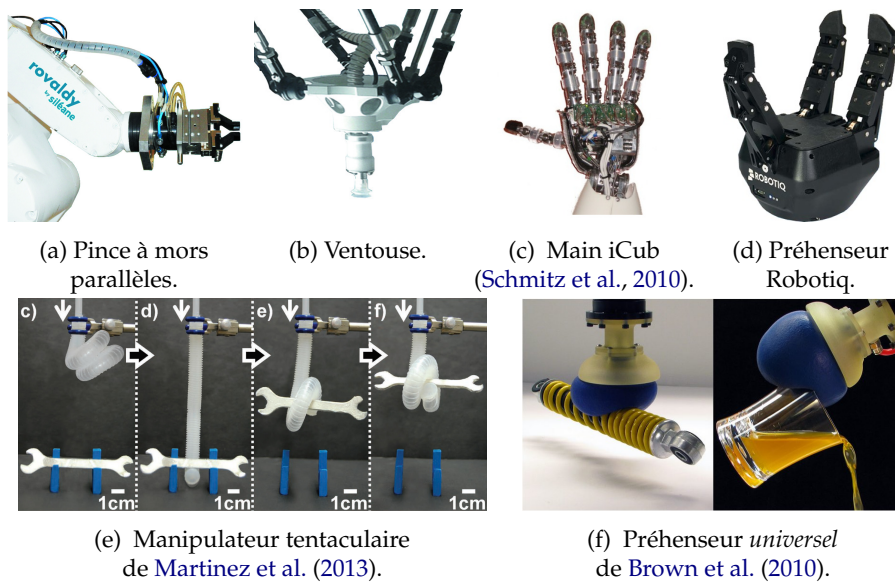
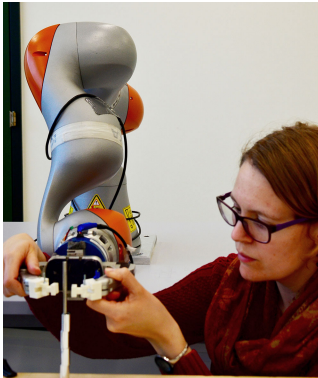


FIGURE 1.3 – Exemples de la variabilité de préhenseurs utilisables en robotique.

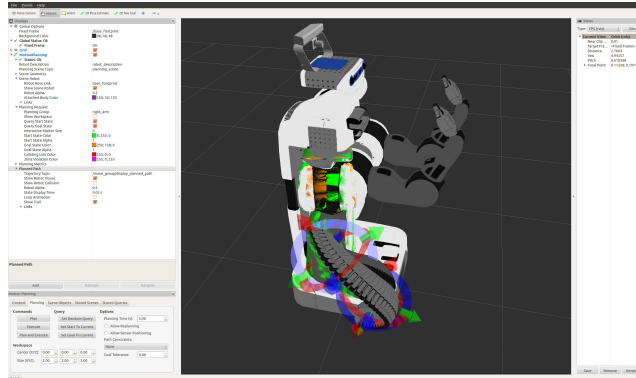
Le pilotage de ces derniers peut cependant s'avérer complexe. Aussi une tendance de fond consiste-t-elle en l'usage de dispositifs sous-actionnés : la main citée précédemment n'est par exemple mue que par 9 actionneurs distincts ; la flexibilité de ses doigts ainsi que les contraintes auxquelles elle est soumise (imposées notamment par la forme de l'objet à saisir) se chargeant de déterminer sa conformation finale selon les lois de la mécanique. Cette tendance est le maître mot de la *soft robotics*, branche qui cherche à maximiser l'adaptabilité du robot à son environnement en limitant l'usage de contraintes dures de position, et dont les préhenseurs de Martinez et al. (2013) et Brown et al. (2010) figures 1.3e et 1.3f sont de parfaites illustrations. Ce dernier notamment, permet de saisir une variété impressionnante d'objets au moyen d'un dispositif simple actionné par création d'une dépression dans une membrane, en exploitant le phénomène de congestion observé dans certains milieux granulaires<sup>1</sup>.

**Capacités sensorielles** L'adaptabilité à l'environnement n'est néanmoins pas nécessairement uniquement d'origine passive, et la disponibilité de capteurs de pression et d'efforts permet l'utilisation de stratégies de manipulations s'adaptant à ces perceptions sensorielles (Hsiao et al., 2010). Des solutions d'asservissement *compliantes* telles que celles décrites par Fuchs et al. (2010) sont aujourd'hui disponibles sur étagère pour la plupart des robots industriels et permettent de piloter ceux-ci en position avec une certaine relaxation de contraintes, de manière à leur permettre de venir au

1. Phénomène connu sous le terme de *jamming*. Transition de phase où la viscosité augmente significativement avec la densité, du fait de phénomènes d'arc-boutements. Ce phénomène est notamment observable avec le café moulu, usuellement capable de s'écouler mais qui présente une forte « rigidité » lorsqu'il est conditionné de manière compacte sous vide.



(a) Enseignement par démonstration (Kronander Automation).



(b) Planification de mouvement (MoveIt! (Şucan et Chitta, 2017)).

FIGURE 1.4 – Possibilités offertes par les capacités des robots modernes.

contact de l'environnement en maîtrisant leur force. Hormis les applications directes offertes par ces technologies, celles-ci ouvrent également des possibilités d'interaction intéressantes entre l'homme et le robot telles que l'enseignement kinestésique (Kronander et Billard, 2014), dans lequel un humain enseigne à un robot les mouvements à accomplir en guidant ce dernier à la main, comme illustré figure 1.4a.

**Planification de mouvement** Les robots industriels usuels sont des machines conçues pour la répétabilité et la précision de leurs mouvements<sup>2</sup>. Il est donc possible de planifier précisément leurs trajectoires avant exécution de manière à optimiser celles-ci et éviter des collisions malheureuses dès lors que l'environnement est convenablement perçu, et il existe aujourd'hui pour cela des solutions logicielles sur étagère relativement éprouvées telles que MoveIt! (Şucan et Chitta, 2017) illustré figure 1.4b. Le lecteur est renvoyé pour d'avantage de information aux travaux de thèse de Diankov (2010) qui dressent un panorama très large de ce sujet.

L'étendue des capacités de manipulation des robots modernes est donc grande, et les difficultés freinant aujourd'hui le développement du dévracage robotisé se concentrent selon nous d'avantage dans la définition d'une stratégie fiable de dévracage plutôt que dans l'exécution de celle-ci.

### 1.3 Stratégies de dévracage

L'exécution par un robot d'une tâche de dévracage nécessite de ce dernier certaines capacités cognitives lui permettant d'adapter ses actions au contexte d'un environnement changeant (le vrac d'objets). Celui-ci doit en effet s'appuyer sur sa perception de la scène – typiquement au moyen d'une caméra ou d'un dispositif d'acquisition 3D – afin de décider des actions

<sup>2</sup>. Les spécifications du robot FANUC M-10iA/10M annoncent par exemple une répétabilité de  $\pm 0.04\text{mm}$  pour une portée de 1422mm.

à accomplir, selon les résultats de son analyse. Du fait des modalités de perception utilisées, la définition d'une stratégie de dévracage constitue donc un problème de vision par ordinateur.

Il est possible de distinguer deux grandes familles d'approches concernant ce problème, que nous détaillons dans la suite de cette section : les *approches basées prise* où la stratégie de dévracage consiste à identifier une manière pertinente de saisir quelque chose dans la scène et qui sont applicables dans une grande variété de situations, et les *approches basées objet* où la manipulation robotisée est assujettie à la localisation préalable d'un objet d'intérêt dans la scène, et qui permet une planification d'actions de plus haut niveau.

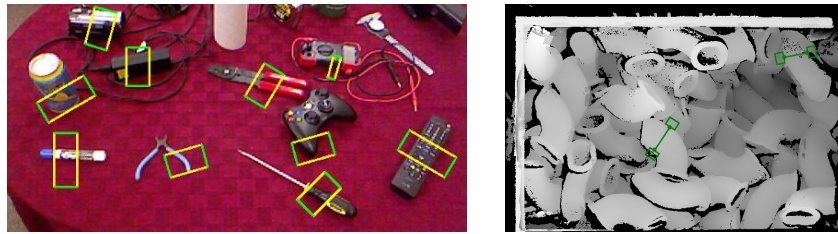
### 1.3.1 Prédiction de prise

La prédiction de prise consiste à être en mesure de proposer étant donné une scène une manière pertinente de saisir quelque chose (ce que l'on nomme une *prise*), ainsi qu'illustré figure 1.5. Il s'agit d'une approche intéressante qui permet à un robot de définir une stratégie d'action étant donné une scène quelconque, sans a priori sur les objets qui la composent. Celle-ci est notamment utilisée au sein de certaines machines de dévracage *Kamido* dont le développement chez Siléane constitue un préambule à cette thèse, et est particulièrement utile pour des applications où il s'agit de séparer des objets arbitraires non connus, telles que du tri de déchet, de la préparation de commande avec un grand nombre de références, etc.

La prédiction de prise est nécessairement spécifique au type de préhenseur utilisé par le robot, aussi les recherches publiées se sont-elles focalisées sur les modèles les plus courants : la pince à mors parallèles, et la ventouse. Il a été montré qu'il est possible de générer des candidats de prises raisonnables à partir de données 2D ou 3D et de méthodes heuristiques (Domae et al., 2014; Asif et al., 2014), par exemple en cherchant des bords parallèles où placer les mors d'une pince. La recherche s'oriente néanmoins globalement vers l'apprentissage automatique d'une stratégie de préhension, afin de tirer parti des progrès grandissants dans ce domaine. Au moyen de données d'apprentissage annotées – manuellement (Jiang et al., 2011; Lenz et al., 2013), par simulation (Mahler et Goldberg, 2017), ou même par un robot menant ses propres expérimentations (Pinto et Gupta, 2016) –, les chercheurs tentent de faire apprendre à un robot à prédire la viabilité d'une prise et ainsi être en mesure de sélectionner la meilleure, voire même à générer automatiquement des prises pertinentes (Redmon et Angelova, 2015).

D'autres travaux plus prospectifs étudient quant à eux la possibilité pour des robots d'apprendre entièrement une stratégie d'asservissement visuel de bas niveau. On citera notamment les expériences de Levine et al. (2016) qui ont, afin d'apprendre à des robots à saisir des objets, fait expérimenter cette opération à 14 d'entre eux en continu pendant deux mois tout en partageant leurs apprentissages.





(a) Exemple des travaux de [Lenz et al. \(2013\)](#). (b) Exemple de prédiction à partir d'une image de profondeur (Siléane).

FIGURE 1.5 – Prédiction de prises viables pour une pince à mors parallèles.

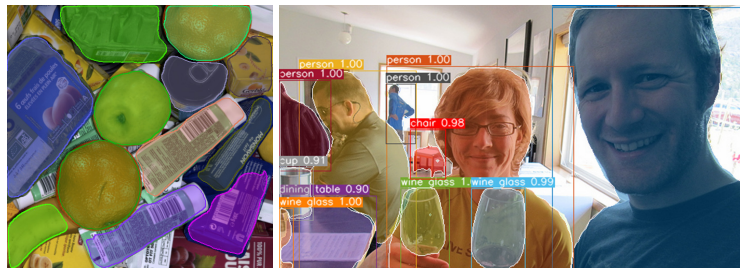


FIGURE 1.6 – Segmentation d'instances. À gauche : méthode de [Grard et al.](#) (à paraître) utilisée sur une scène de vrac. À droite : segmentation d'instances et classification de celles-ci par *Mask R-CNN* ([He et al., 2017](#)).

### 1.3.2 Notion d'instance d'objet

Si l'apprentissage d'une stratégie de préhension adaptable à des scènes arbitraires constitue un terrain d'expérimentation attrayant pour les chercheurs, l'applicabilité d'une telle stratégie seule demeure limitée. En effet, la manipulation robotisée n'a généralement pas pour finalité de saisir quelque chose, mais plutôt d'effectuer quelque traitement précis avec un ou des objets particuliers : séparation unitaire, mise en position, inspection, etc. Pour permettre l'exécution de ces tâches, il convient donc que le robot ait une certaine notion du concept d'objet, ainsi que la capacité d'identifier des instances de l'objet d'intérêt.

**Segmentation d'instance** La détection et la segmentation d'instances représente sans doute le niveau de représentation de base de cette notion d'objet nécessaire à la manipulation, et consiste à identifier les régions des données correspondant à des instances d'objet distinctes, par exemple sous forme de masques binaires dans le cas d'images 2D (voir figure 1.6).

La segmentation des instances permet notamment de mettre en œuvre des stratégies de préhension évitant de saisir plusieurs objets à la fois ou limitant le couple requis pour manipuler un objet en le saisissant au plus proche de son centre. Elle a donc été étudiée dans ce contexte depuis les débuts de la vision par ordinateur jusqu'à nos jours, sous forme d'heuristiques – supposant par exemple qu'une instance présente une surface lisse ([Ikeuchi](#)

et al., 1986) et relativement convexe (Asif et al., 2014) – mais aussi à l’aide de méthodes basées uniquement sur de l’apprentissage automatique (Grard et al., à paraître). L’état de l’art affiche des performances encourageantes pour ce qui est de segmenter des instances de personnes, voitures, etc. dans des images (He et al., 2017) comme illustré figure 1.6 notamment grâce à la disponibilité de données d’apprentissage idoines annotées ; et l’usage de données synthétiques d’apprentissage semble présenter une piste intéressante pour le dévissage robotisé (Grard et al., à paraître).

**Détection d’objet et estimation de pose** L’exécution par un robot d’une tâche impliquant un objet nécessite cependant souvent une analyse plus fine de la scène qu’une simple segmentation de celui-ci dans les données. Beaucoup de tâches de manipulation supposent en effet d’interagir avec un objet différemment suivant sa configuration et nécessitent donc d’être en mesure d’identifier celle-ci – afin par exemple de toujours positionner un flacon correctement pour en verser le contenu indépendamment de sa configuration initiale.

Il n’a pas encore été établi clairement dans quelle mesure et comment des approches d’apprentissage peu supervisées par un humain, telles que celle proposée par Jang et al. (2017) où des robots apprennent à la fois comment saisir un objet et identifier ce dernier, sont en mesure de construire des représentations du concept d’objet et de sa configuration. Quoi qu’il en soit, il convient dans le cadre d’une application pratique d’être en mesure de spécifier au robot ce qu’on attend de lui. Cela est loin d’être évident et fait l’objet de travaux de recherches actifs, tels que ceux de Sermanet et al. (2017) qui cherchent à enseigner à un robot à réaliser des actions comme ouvrir une porte, uniquement à partir de démonstrations que ce dernier observe et cherche à reproduire.

Afin de dépasser ces difficultés et faciliter le dialogue homme-machine, il nous semble donc pertinent d’attendre d’un robot industriel qu’il soit à même de produire et manipuler des concepts similaires à ceux d’un humain. Dans le cadre de cette thèse, le concept décrivant la configuration d’un objet est décrit sous le terme de *pose*, et recouvre notamment les notions de position et d’orientation dans le cas d’un objet rigide. La capacité de détecter des instances d’objet au sein d’une scène et d’estimer leurs poses (ainsi qu’illustré figure 1.7) offre donc une représentation intermédiaire de grand intérêt pour la manipulation robotisée. Une fois la pose d’une instance d’objet connue, il est possible de planifier comment interagir avec celle-ci pour atteindre le but recherché – qui peut par exemple être de placer l’instance d’objet dans une pose donnée. Il est notamment possible de prédire rapidement une manière adéquate de saisir cette instance, à l’aide d’une base de données de « bonnes » prises comme celle illustrée figure 1.8, qui peut être définie manuellement, par expérimentations (Detry et al., 2009), ou par le biais de simulations numériques (Miller et Allen, 2004; León et al., 2010; Vahrenkamp et al., 2013; Weisz et Allen, 2012) s’appuyant sur des travaux théoriques permettant d’évaluer la qualité de préhension (Suárez et al., 2006).

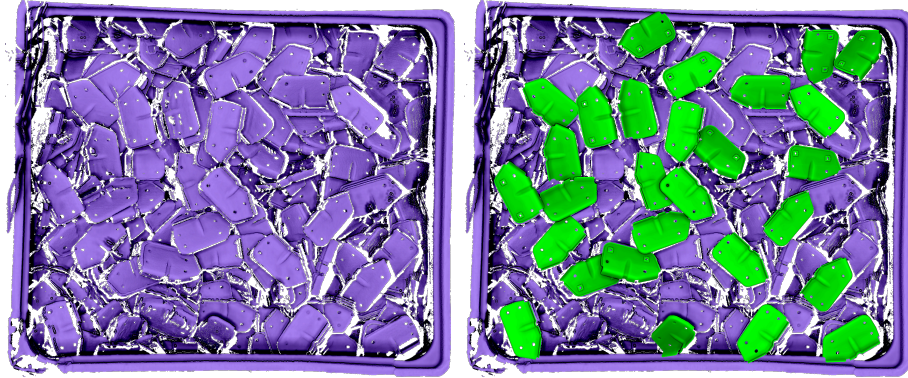


FIGURE 1.7 – Détection et estimation de pose d’instances d’un objet rigide connu. À gauche : Données d’entrée, représentées sous forme d’un nuage de points 3D projeté. À droite : superposition d’un rendu d’un modèle de l’objet en les poses d’instances détectées à l’aide de la méthode développée dans cette thèse.



FIGURE 1.8 – Exemple de base de données de prises avec une pince à mors parallèles, permettant une fois la pose d’un objet estimée de définir comment le saisir.



FIGURE 1.9 – Exemple de scènes contenant plusieurs instances en vrac d’un même objet rigide sans variabilité ni texture, qui constituent notre cas d’application typique.

## 1.4 Objet de ces travaux

Nous nous focalisons dans le cadre de cette thèse sur le problème de la **détection d’instances d’objet et de l’estimation de la pose de ces derniers**, car ainsi qu’on l’a vu sa résolution permet d’envisager l’exécution de tâches de manipulation robotisées avancées, et dont la spécification est facilitée par l’utilisation d’un concept appréhendable par l’humain qu’est la notion de pose.

Hormis la robotique, ce problème de vision par ordinateur présente notamment des applications en réalité augmentée, en analyse de scène, en contrôle qualité ou en suivi d’objet. Nous mettons cependant dans nos travaux l’accent sur un scénario de dévracage industriel de par la nature appliquée de nos recherches, et plus spécifiquement sur le cas de scènes représentant des instances d’une unique classe d’objet connue, sans variabilité intraclasse<sup>3</sup> et sans texture particulière, telles que celles illustrées figure 1.9.

Après un état de l’art général sur la problématique de détection et d’estimation de pose que nous synthétisons chapitre 2, nous constatons qu’il y a encore matière à recherche dans ce domaine, et notamment vis-à-vis des questions suivantes, sur lesquelles nous avons choisi de focaliser notre attention :

### Comment représenter de manière pertinente le concept de pose d’objet ?

La notion de pose est la plupart du temps considérée comme acquise au sein de l’état de l’art. Celle-ci n’est cependant pas aussi triviale qu’il n’y paraît, notamment dès lors que l’objet présente certaines propriétés de symétrie – ce qui est fréquent dans le cas des objets manipulés dans l’industrie. Nous revenons donc sur cette question et proposons dans le chapitre 3 un cadre théorique général permettant d’y répondre pour un objet rigide arbitraire.

3. On entend par là que les instances d’objet sont toutes considérées comme identiques les unes aux autres, ce qui implique notamment que l’objet est nécessairement rigide.

**Comment exploiter cette notion de pose afin de détecter et localiser des instances d'objet dans des scènes de vrac de manière automatique et efficace ?**

Nous proposons pour cela une approche adaptée aux problématiques du dévracage industriel, exploitant une image de profondeur et entraînée au moyen d'un modèle 3D à discriminer les poses potentielles de l'objet. Celle-ci est présentée dans le chapitre 4.

**Comment évaluer les performances ?**

Malgré les années de recherche en détection et en estimation de pose d'objet, les performances de l'état de l'art demeurent en effet peu claires – notamment en ce qui concerne la problématique du dévracage – faute de protocole d'évaluation adapté. Aussi proposons-nous une méthodologie d'évaluation se voulant plus pertinente que l'existant au regard de cette problématique, et l'employons-nous afin d'évaluer notre approche. C'est là l'objet du chapitre 5.

## 1.5 Publications

Les différents travaux de cette thèse ont été réalisés sous couvert de secret industriel. Certains aspects ont néanmoins pu donner lieu à des publications internationales avec comité de lecture : un article publié dans le *International Journal of Computer Vision (IJCV)* (Brégier et al., 2017b) présentant notre formalisation du concept de pose et dont le contenu est repris chapitre 3, ainsi qu'un papier primé du *Best Paper Award* lors du *3rd International Workshop on Recovering 6D Object Pose* organisé à l'occasion de l'*International Conference on Computer Vision (ICCV)* (Brégier et al., 2017a). Ce dernier présente la méthodologie d'évaluation développée dans le cadre de ces travaux, et qui constitue une partie du chapitre 5. Enfin, nos recherches ont donné lieu à une publication conjointe (Grard et al., à paraître) sur la segmentation d'instances d'objet dans le cadre du dévracage industriel, qui ne fait pas l'objet de ce manuscrit.

## Chapitre 2

# État de l'art

*Nous sommes comme des nains assis sur des épaules de géants. Si nous voyons plus de choses et plus lointaines qu'eux, ce n'est pas à cause de la perspicacité de notre vue, ni de notre grandeur, c'est parce que nous sommes élevés par eux.*

– Bernard de Chartres, d'après Jean de Salisbury, *Metalogicon*.

---

2.1	Énoncé du problème . . . . .	20
2.2	Modalités d'acquisition . . . . .	23
2.3	Données d'apprentissage . . . . .	27
2.4	Approches globales . . . . .	30
2.5	Approches locales . . . . .	40
2.6	Raffinement et vérification d'hypothèses . . . . .	48
2.7	Gestion des symétries . . . . .	50
2.8	Synthèse et pistes de recherche . . . . .	51

---

Comme il a été énoncé dans l'introduction, nos travaux de recherches portent sur la détection et l'estimation de pose d'instances d'objet au sein d'une scène, de manière automatique. Il ne s'agit pas là d'une problématique nouvelle dans le domaine de la vision par ordinateur ; au contraire celle-ci est étudiée depuis plus de trente ans. Nous cherchons donc dans ce chapitre à dresser un panorama des différentes approches étudiées et publiées par le passé en lien avec ce sujet, et qui constitue une étape préalable essentielle à la définition de nos propres pistes de recherche, que nous présentons section 2.8.

Pour ce faire, nous commençons section 2.1 par revenir sur la définition de ce problème, et passons rapidement en revue section 2.2 les principales modalités de perception exploitables afin de réaliser cette tâche. Afin de pouvoir reconnaître un objet il est nécessaire de disposer d'un certain a priori sur ce dernier, aussi nous abordons également section 2.3 la question des modalités de données utilisées pour apprendre à réaliser cette tâche.

Dans un second temps, nous dressons un panorama des différentes approches envisagées pour résoudre ce problème dans la littérature, en distinguant de manière relativement arbitraire entre les méthodes globales section 2.4, qui prennent en compte l'entièreté de l'objet afin de détecter et

localiser ses instances, et celles se concentrant sur des caractéristiques plus locales afin de produire des hypothèses de pose, section 2.5.

La section 2.6 est quant à elle consacrée à la question du raffinement et de la vérification de ces hypothèses de pose, tandis que nous revenons dans la section 2.7 sur les difficultés introduites par les objets symétriques.

## 2.1 Énoncé du problème

En préambule de toute recherche, il est essentiel de définir précisément le problème étudié. Dans cette section, nous revenons donc sur la définition de certains problèmes clé de la vision par ordinateur (sous-section 2.1.1), ainsi que sur une formulation proche de notre sujet d'intérêt ayant été proposée dans l'état de l'art (sous-section 2.1.2) et à partir de laquelle nous définissons finalement sous-section 2.1.3 l'objet de nos travaux.

### 2.1.1 Définitions

Il existe un certain nombre de problèmes typiques en vision par ordinateur dont la distinction n'est pas toujours claire dans la littérature, ces derniers étant intimement liés. Nous revenons donc sur quelques éléments de terminologie, illustrés figure 2.1, afin de limiter les malentendus.

**Reconnaissance d'objet (*object recognition*)** La reconnaissance d'objet consiste à identifier la classe d'objet auquel appartient l'instance représentée par les données. Les classes considérées peuvent être diverses (humain, personne particulière, œil, cheval, voiture, etc.), et comprennent usuellement une classe représentant le complément des autres classes d'intérêt<sup>1</sup>, de sorte que la reconnaissance d'objet n'est en réalité qu'une dénomination spécifique du problème général de classification rencontré dans l'ensemble des domaines de l'intelligence artificielle.

**Détection d'objet (*object detection*)** La détection d'objet consiste en la capacité d'identifier la présence d'une instance d'objet dans des données. Cette terminologie est notamment largement utilisée en analyse d'images, où la détection d'objet prend typiquement la forme de génération de boîtes englobantes délimitant des zones de l'image contenant une instance d'objet. Le célèbre détecteur de visage de [Viola et Jones \(2001\)](#) en est un bon exemple. Celui-ci procède en scannant l'image par fenêtre glissante à différentes échelles et en classifiant si l'imagette contenue dans cette fenêtre représente un visage ou non. La détection n'est cependant pas nécessairement réalisée par classification, à l'instar d'approches telles que celle de [Hough \(1959\)](#); [Duda et Hart, 1972](#)) pour détecter des segments de droite, sur laquelle nous revenons section 2.5.3.

**Estimation de pose (*pose estimation* ou *pose recovery*)** Enfin, l'estimation de pose consiste à déterminer la *pose* d'une instance d'objet, c'est

---

1. Un module de reconnaissance de photographies d'animaux sera par exemple conçu pour discriminer entre des photos de chien, de chat, etc. ainsi que des photos ne représentant pas un animal.

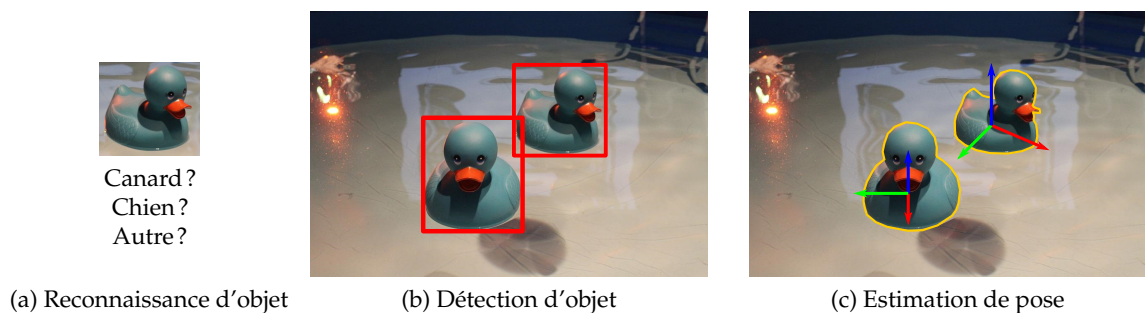


FIGURE 2.1 – Problèmes typiques de vision par ordinateur.

à dire la manière dont cet objet est configuré dans l'espace. Cette configuration est usuellement modélisée au moyen d'un ensemble de paramètres qu'il s'agit de recouvrer. Dans le cas d'un objet rigide il s'agit typiquement de sa position et de son orientation qu'on peut décrire au moyen d'une transformation rigide, mais il peut également s'agir de modèles plus avancés tels que dans le cas de la pose humaine la position d'un ensemble de points clés (tête, coude, poignet, etc.).

### 2.1.2 Reconnaissance d'objets 3D à partir d'une unique vue arbitraire

Les problématiques précédentes sont souvent traitées de manière conjointe car ces dernières participent toutes de l'analyse et de l'interprétation de scène. Dans une revue de l'état de l'art en 1985, [Besl et Jain \(1985\)](#) ont formalisé un cas particulier d'une telle analyse sous le nom de *reconnaissance d'objets 3D à partir d'une unique vue arbitraire*. Leur formalisation ne semble pas s'être imposée au sein de la communauté, cependant celle-ci a le mérite d'être éclairante sur l'objet de nos travaux, aussi nous reprenons celle-ci ici :

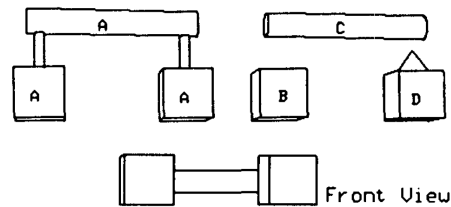
On considère un ensemble de classes d'objets solides distinctes étiquetées et une scène statique.

**Apprentissage** Chaque objet peut être examiné pourvu qu'il ne soit pas déformé, et des modèles étiquetés correspondants peuvent être générés à partir de cette examen.

**Reconnaissance** A partir des données d'un capteur prises selon un point de vue particulier et de l'apprentissage préalable, il s'agit de détecter pour chaque classe d'objet la présence ou l'absence de celle-ci dans les données, le nombre d'instances de cette classe, ainsi que de déterminer la position 3D et l'orientation 3D de chacune des instances relativement à un repère connu.

Besl et Jain ajoutent de plus un troisième objectif optionnel consistant en la caractérisation des régions de données ne correspondant à aucun des objets d'apprentissage, de sorte d'être en mesure de les reconnaître en cas d'occurrences futures.





(a) Exemple d'interprétations multiples envisageables (en haut) à partir d'une même vue d'une scène (en bas). Extrait de (Besl et Jain, 1985).



(b) Illusion célèbre admettant deux interprétations viables et illustrant la manière dont nous choisissons une interprétation possible en cas d'ambiguïté. *My Wife and My Mother-In-Law*, William Hill, 1915.

FIGURE 2.2 – Ambiguïté d'interprétations d'une scène.

**Gestion des ambiguïtés** L'objectif pour ces auteurs ne se limite de plus pas seulement à trouver une interprétation valide des données sous forme de la détection, reconnaissance et localisation d'un ensemble d'instances d'objets, mais consiste plutôt à trouver l'ensemble des différentes interprétations possibles des données, en cas de scène présentant une ambiguïté comme celle représentée figure 2.2a. La levée des ambiguïtés consiste alors en un problème différent, et qui peut notamment être abordé en exploitant des acquisitions de la scène suivant d'autres points de vue.

Cette position vis-à-vis des ambiguïtés, assez représentative d'une époque où la vision par ordinateur attachait une importance particulière à l'exactitude et l'exhaustivité, semble cependant différer de la manière dont notre propre perception fonctionne comme l'illustre la figure 2.2b, et ne semble de plus guère réalisable en pratique. Il existe en effet pour certaines vues de scène une infinité non dénombrable d'interprétations possibles, et ce notamment dès lors que certaines instances d'objets sont fortement occultés. Dans le cas des scènes de vrac qui constituent notre sujet d'étude, la multiplication de points de vue peut même vraisemblablement ne pas toujours suffire à lever l'indétermination, l'occultation de certains objets pouvant être importante pour l'ensemble des points de vue envisageables. Nous abordons donc nos recherches avec un objectif moins ambitieux, explicité dans la sous-section suivante.

### 2.1.3 Objet de cette thèse : détection et estimation de pose d'instances d'objet

Nos travaux de recherche ont pour objet la *détection et l'estimation de pose* d'instances d'un objet dans une scène, de manière automatique, à partir de données représentant celle-ci. De même que Besl et Jain (1985), nous considérons le cas d'une classe d'objet solide, sans variabilité intra-classe, c'est à dire dont les instances ne présentent pas de différences perceptibles entre elles,

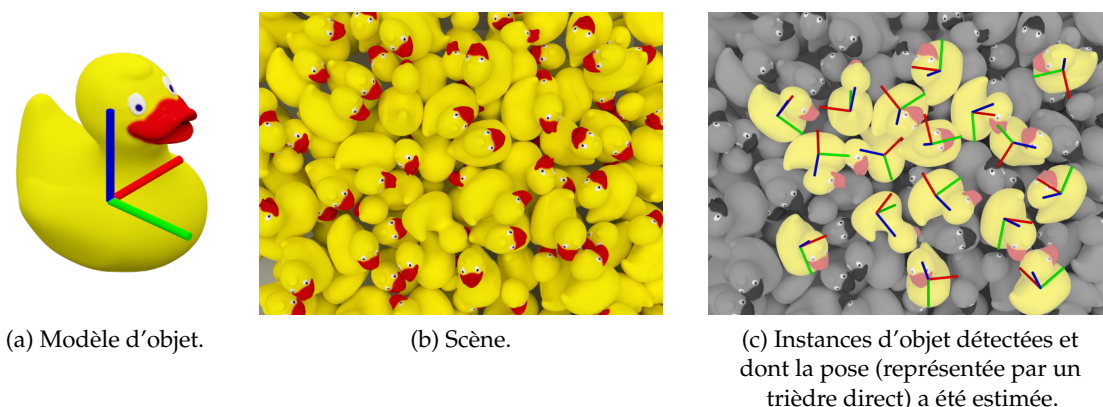


FIGURE 2.3 – Objet de cette thèse : détection et estimation de pose d'instances d'un objet rigide connu dans une scène ne contenant que des instances de ce dernier.

et laissons la possibilité pour la machine d'apprendre les caractéristiques de l'objet d'intérêt au préalable.

Du fait de notre visée applicative de débrassage, nous laissons cependant de côté la problématique de reconnaissance d'objet en considérant que l'ensemble des données représentent des instances d'une unique classe, comme illustré figure 2.3. Notre objectif n'est de plus pas de proposer l'ensemble des interprétations possibles de la scène, mais plus modestement de générer des hypothèses pertinentes concernant la pose d'instances d'objet présentes dans celles-ci, potentiellement associées à une notion de confiance en ces hypothèses.

## 2.2 Modalités d'acquisition

La détection et l'estimation de pose d'instances d'objet est conditionnée aux données à partir desquelles cette opération est exécutée. Afin de percevoir rapidement l'ensemble d'une scène, il est pertinent d'utiliser pour ce faire des capteurs sans contact, dont le traitement automatique des données est l'objet de la vision par ordinateur.

**Caméra 2D** La caméra matricielle est certainement un des plus usuels de ces capteurs, et permet d'acquérir un point de vue d'une scène sous forme d'une image 2D, mono- ou polychrome. Dans ce document, nous considérons par simplicité une caméra perspective usuelle<sup>2</sup> et modélisons la projection 2D effectuée par celle-ci au moyen du modèle du sténopé représenté figure 2.4, une fois les éventuelles distorsions de l'objectif et du capteur corrigées. Suivant ce modèle, un point de l'espace de coordonnées  $(x, y, z)^T \in \mathbb{R}^3$  dans le repère de la caméra se projette en un pixel de coordonnées  $(u, v)^T \in \mathbb{R}^2$  suivant l'opération suivante, exprimée en coordonnées

2. D'autres types de caméra sont couramment employés en vision industrielle telles que des caméras linéaires ou à projection orthographiques (au moyen d'objectifs télécentriques).

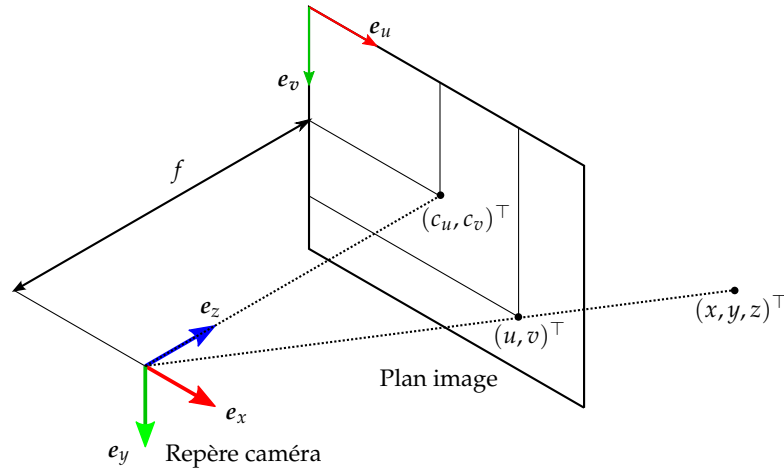


FIGURE 2.4 – Modèle de caméra : le sténopé.

homogènes :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \propto \underbrace{\begin{pmatrix} f & 0 & c_u \\ 0 & f & c_v \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{K}} \begin{pmatrix} x \\ y \\ z \end{pmatrix}. \quad (2.1)$$

La matrice  $\mathbf{K}$  représente les paramètres intrinsèques de la caméra, comprenant la distance focale  $f$  (en pixel) du centre optique du sténopé au plan image, et  $(c_u, c_v)^\top \in \mathbb{R}^2$  les coordonnées du point principal de la caméra, qui correspond à la projection du centre optique sur le plan image. Ces paramètres peuvent être estimés par calibrage préalable, par exemple au moyen de la méthode de [Zhang \(2000\)](#), de même que la position et l'orientation de la caméra dans l'espace, et le lecteur intéressé par ces aspects géométriques de la vision par ordinateur est renvoyé au livre de [Hartley et Zisserman \(2003\)](#).

Si la détection et l'estimation de pose d'un objet à partir d'une unique image de caméra 2D est envisageable (c'est notamment l'approche retenue dans les récents travaux de [Kehl et al. \(2017\)](#) et [Rad et Lepetit \(2017\)](#)), une telle estimation est en générale trop imprécise pour pouvoir être exploitée seule à des fins de manipulation robotique. En effet, la pose d'une instance d'objet sera typiquement estimée en ajustant un certain modèle de l'objet aux données de l'image observées comme illustré figure 2.5a. Un tel ajustement ne peut être parfait et souffre d'une certaine incertitude, incertitude qui se répercute sur la pose estimée. La profondeur  $z \in \mathbb{R}^{+*}$  d'un objet par rapport à la caméra peut notamment être estimée en première approximation par sa taille apparente  $s \in \mathbb{R}^{+*}$  dans l'image<sup>3</sup>. En notant  $\delta_s$  l'incertitude sur cette

3. Via la relation  $\frac{z}{S} = \frac{f}{s}$  où  $S$  représente la taille réelle de l'objet dans le cas d'un objet plan orthogonal au plan image.

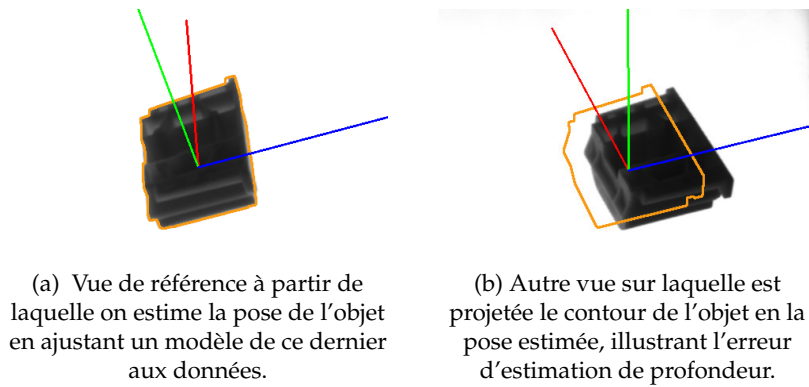


FIGURE 2.5 – Incertitude d'estimation de pose à partir d'une unique vue 2D.

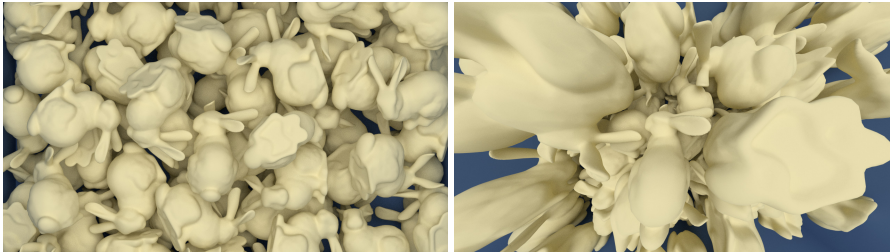


FIGURE 2.6 – Nécessité d'un recul suffisant pour imager une scène présentant un certain relief telle qu'un vrac. **À gauche** : scène imagée de loin avec une caméra d'angle de champ de  $15^\circ$ . **À droite** : même scène imagée de près avec un capteur grand angle ( $145^\circ$ ) couvrant un champ similaire mais où de nombreuses instances sont peu visibles du fait d'occultations.

taille apparente, l'incertitude  $\delta_z$  sur la profondeur de l'objet s'exprime alors

$$\delta_z = z \frac{\delta_s}{s}, \quad (2.2)$$

et est linéaire en la distance  $z$  de l'objet à la caméra et en l'incertitude relative ( $\delta_s/s$ ) de l'ajustement du modèle à l'image. Or, afin de pouvoir embrasser du regard une scène de dimensions conséquentes devant celles de l'objet, il convient de prendre un recul tout aussi conséquent (c.-à-d.  $z$  significatif devant  $S$ ), l'usage d'un point de vue rapproché et d'un objectif grand angle introduisant des problèmes d'occultation importants dès lors que la scène présente un certain relief comme illustré figure 2.6. L'incertitude en profondeur d'une pose estimée à partir d'une seule vue est donc dès lors elle aussi significative devant les dimensions de l'objet, comme l'illustre la figure 2.5. Une telle incertitude n'est pas gênante pour des applications de réalité augmentée, mais est problématique dans un contexte de manipulation, à moins de mettre en place un système de compensation d'erreur tel qu'un asservissement visuel ou tactile.

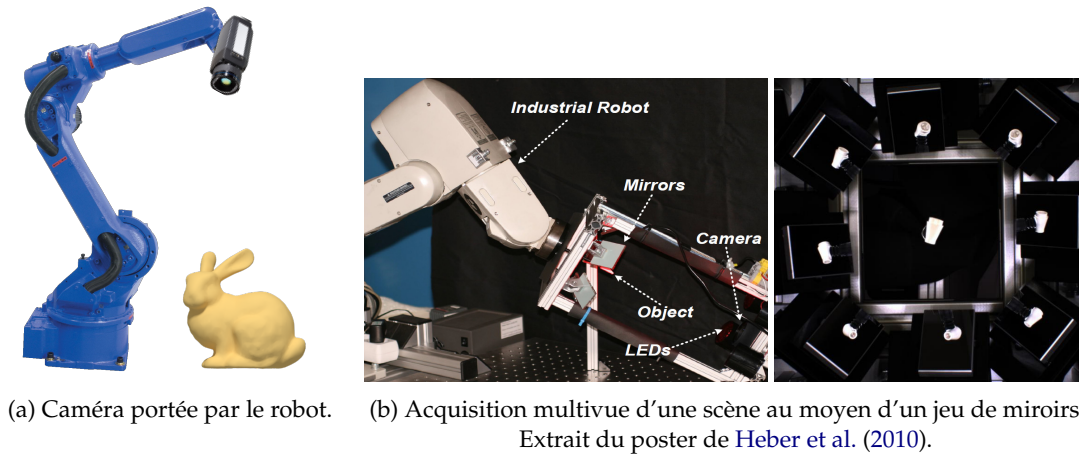


FIGURE 2.7 – Dispositifs permettant de multiplier les points de vue.

**Acquisition multivues** Afin de dépasser cette limitation, il est possible de *triangler* la position de l'objet en observant la scène selon plusieurs points de vue. Cette observation peut être réalisée à l'aide de plusieurs caméras positionnées en des points différents, ou encore au moyen d'une unique caméra que l'on déplace dans la scène. Ce second scénario – équivalent au premier pourvu que la scène soit statique – peut notamment être réalisé en fixant la caméra sur l'effecteur du robot, suivant un montage dit *eye-in-hand* dans la littérature robotique illustré figure 2.7a. Dès lors, il est envisageable de pratiquer une vision dite *active*, où les points de vue utilisés pour imager la scène sont sélectionnés à la volée, à la manière de Dumanoglou et al. (2016) qui proposent de prédire le prochain point de vue à considérer de façon à maximiser le gain d'information relativement à la pose incertaine d'instances d'objets.

Les points de vue peuvent également être multipliés sans pour autant augmenter le nombre d'acquisitions aux moyens de dispositifs optiques particuliers, à l'instar de celui de Heber et al. (2010) intégrant plusieurs miroirs illustré figure 2.7b.

**Acquisition géométrique** Les données d'une caméra 2D peuvent cependant présenter de grandes variations suivant les conditions d'éclairage, les matériaux utilisés et leurs propriétés optiques. Ces variations et la complexité des phénomènes optiques en jeu rendent les images 2D difficiles à exploiter de manière automatique. Afin de s'affranchir de ces difficultés, Agrawal et al. (2010) ont proposé un dispositif illustré figure 2.8 permettant de détecter les bords de profondeur d'un objet dans une image indépendamment de la texture de la scène, en réalisant plusieurs acquisitions 2D avec un éclairage selon des directions différentes. Bien que l'usage de leur système ne soit que peu répandu, ce type d'approche consistant à travailler à partir de données purement géométriques insensibles à l'éclairage et dont les propriétés sont bien connues – telles que des points 3D, des surfaces et leurs normales, etc.–, est usuel dans l'industrie. Des données 3D peuvent

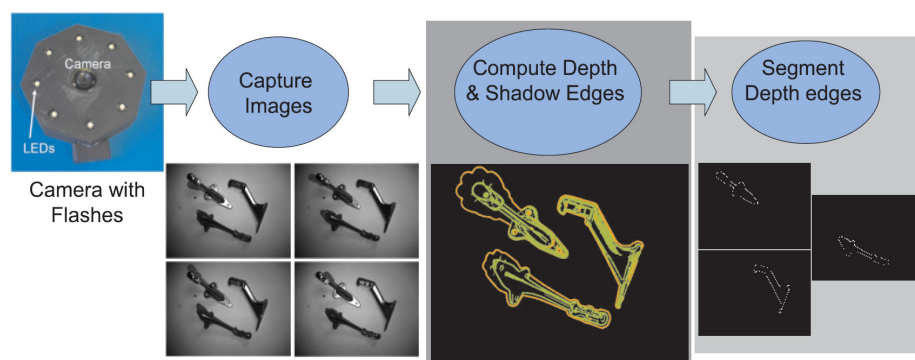


FIGURE 2.8 – Dispositif pour détecter les bords de profondeur dans une image, à partir d’ombres portées par différents éclairages . Extrait de (Agrawal et al., 2010).

être organisées sous forme d’une image dite « 2.5D » ou *image de profondeur* illustrée figure 2.9a, qui à chaque pixel associe la distance du capteur au point 3D correspondant à ce pixel<sup>4</sup>, ou encore désorganisées sous forme d’un nuage de points 3D illustré figure 2.9b et qui permet une représentation simple d’informations provenant de plusieurs vues. Une grande variété de capteurs permettent d’acquérir de telles informations, par triangulation (stéréoscopie passive ou active, scanner avec projection de franges ou de ligne LASER, etc.), exploitation d’une profondeur de champ limitée (Depth From Focus / Defocus), estimation du temps de vol aller-retour du capteur à la scène (caméra temps de vol, LIDAR) ou encore par des approches plus spécifiques, telles que l’imagerie polarimétrique dans le cas de matériaux spéculaires (Morel, 2005). L’apparition de capteurs destinés au marché de masse tels que le Microsoft Kinect en 2010 a largement augmenté l’engouement pour cette modalité de profondeur dans la recherche en vision par ordinateur, faisant de l’image RGBD – combinant une image couleur et une image de profondeur – le standard utilisé dans de nombreuses recherches en estimation de pose.

## 2.3 Données d’apprentissage

Afin qu’une machine soit en mesure de détecter et estimer automatiquement la pose d’instances d’un objet, celle-ci doit engranger au préalable un certain nombre d’information vis-à-vis de ce dernier et de son environnement, qui peuvent lui être fournies sous plusieurs formes.

**Exemples annotés** Ces informations peuvent notamment consister en des acquisitions annotées avec la pose des instances d’objet à détecter. L’annotation manuelle de données est fastidieuse et coûteuse en temps ; aussi bien

4. On suppose alors que les objets de la scène sont totalement opaques de sorte qu’il n’est pas possible de voir au travers d’un objet. De manière paradoxale, Lysenkov et Rabaud (2013) proposent d’exploiter cette incapacité à définir la profondeur en présence d’objets transparents afin de détecter ceux-ci.

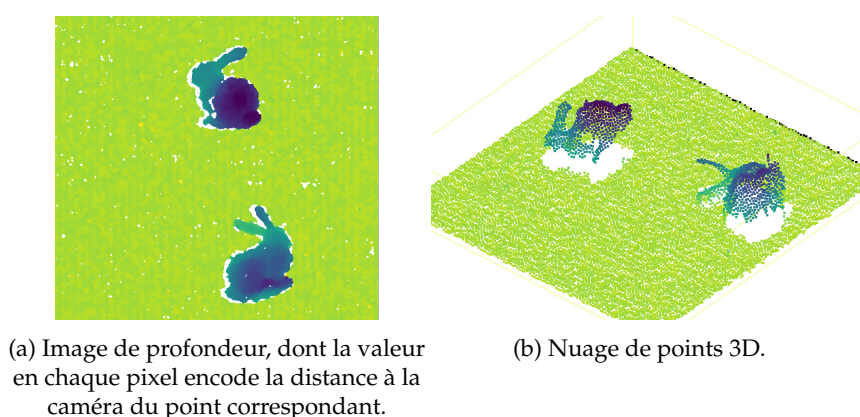


FIGURE 2.9 – Représentations de données 3D d'une même scène.

qu'elle ait pu être envisagée pour certains cas spécifiques tel que la pose humaine (Lin et al., 2014) comme illustré figure 2.10a, il ne s'agit pas d'une approche généralisable facilement à tout type de scénario. Une alternative mise en œuvre notamment par Hinterstoisser et al. (2011); Hodaň et al. (2017) pour un objet rigide consiste à multiplier les acquisitions suivant des points de vue connus autour d'une instance d'objet de référence afin d'obtenir des données annotées à peu de frais, tandis que Novotny et al. (2017) utilisent quant à eux astucieusement un algorithme de *Structure From Motion* afin d'annoter automatiquement des vidéos représentant une instance d'une classe d'objet (par exemple une voiture) sous différentes poses. Cette procédure se fait néanmoins au dépend de la représentativité des exemples annotés, car l'annotation automatique impose certaines conditions (instance d'objet isolée dans une pose de référence, etc.) qui ne sont pas nécessairement celles rencontrées en phase de test.

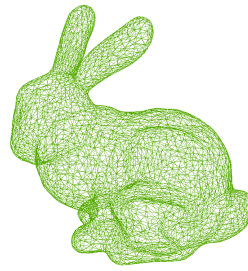
**Modèle d'objet** Dans de nombreux cas, il est cependant possible de disposer d'un modèle explicite et pertinent de l'objet au préalable, et celui-ci peut alors être utilisé directement au lieu d'exemples annotés. Bolles et Horaud (1986) proposaient par exemple dans les années 1980 d'exploiter un modèle d'objet rigide sous forme d'un ensemble de primitives représentant les arrêtes de ce dernier. Aujourd'hui la disponibilité d'outils de reconstruction 3D sur étagère permettent d'obtenir relativement facilement un modèle numérique d'un objet rigide, lorsqu'un modèle CAO<sup>5</sup> de ce dernier n'est pas directement disponible. Cette modalité est donc utilisée par la plupart des approches modernes sous des formes telles qu'un nuage de points (Johnson et Hebert, 1999; Drost et al., 2010) ou encore un maillage polygonal (Park et al., 2010; Hinterstoisser et al., 2012b; Kehl et al., 2017) tel qu'illustré figure 2.10b.

**Exemples synthétiques** Il est également possible d'entraîner un algorithme à partir de données synthétiques, de manière à lui spécifier un

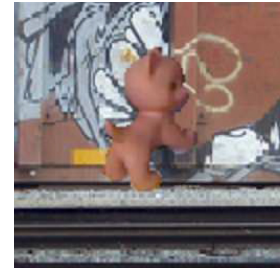
5. Conception Assisté par Ordinateur.



(a) Annotations manuelles de pose, ici humaine sous forme d'un ensemble de points clés 2D. (challenge MS COCO (Lin et al., 2014)).



(b) Modèle a priori de l'objet (ici sous forme de maillage polygonal).



(c) Données synthétique d'apprentissage pour l'estimation de pose dans un environnement en fouillis (Rad et Lepetit, 2017).

FIGURE 2.10 – Exemples de données utilisées afin d'apprendre à détecter et estimer la pose d'instances d'un objet.

apriori sur l'environnement lorsque celui-ci est difficile à formaliser. Ce type d'approche prend tout son sens dans le cadre de l'apprentissage profond qui nécessite de grandes quantités d'exemples d'apprentissage. Ces données peuvent prendre la forme d'une simple « augmentation » d'un jeu de données réel existant, en y appliquant des transformations diverses (ajout de bruit, rotations d'images, etc.) afin de spécifier à l'algorithme comment réagir face à ces transformations ; mais également de données entièrement générées par ordinateur. Shotton et al. (2013a) entraînent par exemple leur algorithme d'estimation de pose humaine dans une image de profondeur à partir d'images synthétiques produites à l'aide de modèles paramétriques de corps humain, associés à un modèle d'apriori de pose constitué au préalable par capture du mouvement d'acteurs réels. Kehl et al. (2017) et Rad et Lepetit (2017) entraînent quant à eux leurs méthodes à estimer la pose d'un objet rigide particulier dans une scène présentant un fouillis important par une combinaison de données réelles et de rendus synthétiques de l'objet sur des fonds arbitraires comme illustré figure 2.10c, Rad et Lepetit (2017) simulant également des phénomènes d'occlusion partielle.

**Adaptation de domaine** L'apprentissage à partir d'exemples comporte néanmoins la difficulté que ceux-ci doivent être autant que possible représentatifs des données que la machine percevra lors de ses tests. L'usage de données synthétiques ou de procédures d'apprentissage spécifiques conduit cependant nécessairement à produire une distribution d'exemples différente de celle rencontrée en réalité, et il convient dès lors que l'apprentissage puisse être transposables au domaine d'utilisation, processus parfois qualifié d'*adaptation de domaine* dans la littérature.

Au moyen d'une conception judicieuse, il est possible d'atteindre (partiellement tout du moins) cette adaptation de domaine en focalisant l'apprentissage sur des caractéristiques conçues afin d'être relativement invariantes au domaine, telles que des caractéristiques géométriques extraites d'images 3D de la scène (Johnson et Hebert, 1999; Drost et al., 2010; Shotton et al.,



2013a; Tejani et al., 2014; Brachmann et al., 2014) ou encore les contours de la silhouette d'un objet dans une image (Hinterstoisser et al., 2012b). L'exploitation de caractéristiques définies manuellement ne va cependant pas dans le sens de l'évolution de l'apprentissage par ordinateur, qui tend à privilégier l'apprentissage profond et suppléer à l'expertise humaine l'accumulation d'exemples d'apprentissage ou d'expériences. L'adaptation de domaine constitue alors un enjeu essentiel de recherche.

Une solution pratique couramment utilisée dans les techniques d'apprentissage profond et connue sous le nom de *transfer learning* consiste à transférer au problème d'intérêt un apprentissage réalisé de manière relativement extensive sur un domaine et un problème suffisamment proche du problème étudié comme la détection et la reconnaissance d'objet, de sorte à profiter des capacités d'abstraction apprises pour ce dernier en espérant que celles-ci soient suffisantes pour résister au changement de domaine. Rad et Lepetit (2017) ainsi que Kehl et al. (2017) procèdent par exemple ainsi afin de détecter et estimer la pose d'un objet dans une image.

Certains chercheurs tentent de plus d'apprendre à leurs algorithmes à faire abstraction de la différence entre deux domaines et ainsi améliorer le transfert d'apprentissage, à l'instar de Rad et al. (2017) qui cherchent à produire une représentation robuste aux variations d'illumination, ou de Massa et al. (2016) qui apprennent à adapter les descripteurs d'une image réelle afin qu'ils soient similaires à ceux produits par un rendu d'un modèle CAO. D'autres comme Bousmalis et al. (2017) explorent la possibilité de générer des données synthétiques réalistes à partir de rendus CAO en hallucinant des détails ou un environnement de manière à produire des données d'apprentissage plus représentatives de celles de test. Il s'agit cependant d'un problème encore largement ouvert et qui fait l'objet de recherches actives.

Diverses modalités peuvent ainsi être fournies à un algorithme afin d'entraîner ce dernier à détecter et estimer la pose d'un objet. Ces modalités ne sont pas incompatibles entre elles et les travaux de Oberweger et al. (2015) en constituent un excellent exemple. Ces chercheurs entraînent en effet une méthode à estimer la pose d'une main à partir de données 3D annotées. Leur approche comprend un modèle génératif de données lui-même appris, et qui est utilisé afin de synthétiser certains exemples pour entraîner l'étape final de leur approche. Ces différentes modalités forment donc un ensemble de sources d'informations disponibles plus ou moins facilement suivant les applications, et qu'il s'agit d'exploiter au mieux.

## 2.4 Approches globales

Afin de détecter et estimer la pose d'instances d'objet rigide, différents types d'approches ont été proposés dans la littérature. Nous introduisons dans cette section celles que l'on pourrait qualifier de *globales* – parfois également évoquées sous le terme *holistique* –, et qui exploitent une représentation de l'objet dans son ensemble lors de leurs phases d'inférences.

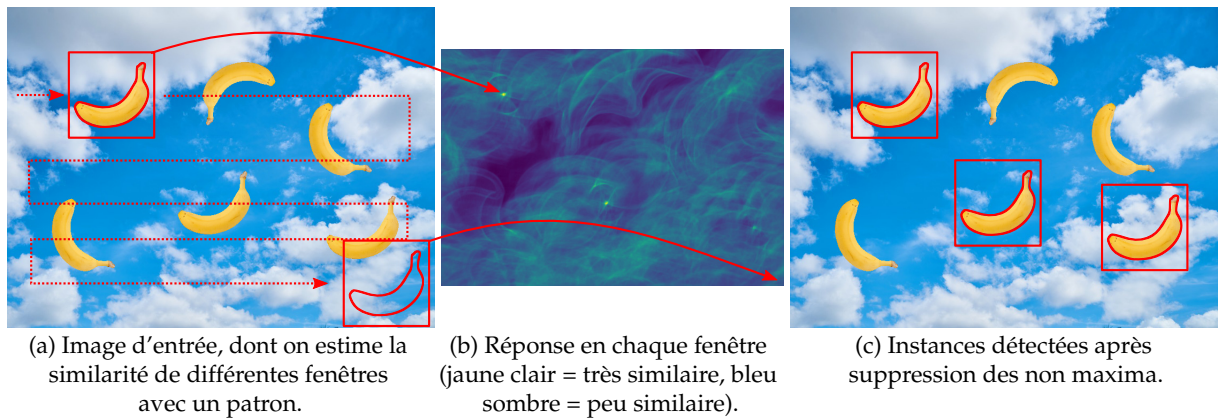


FIGURE 2.11 – Recherche de patrons 2D par fenêtre glissante.

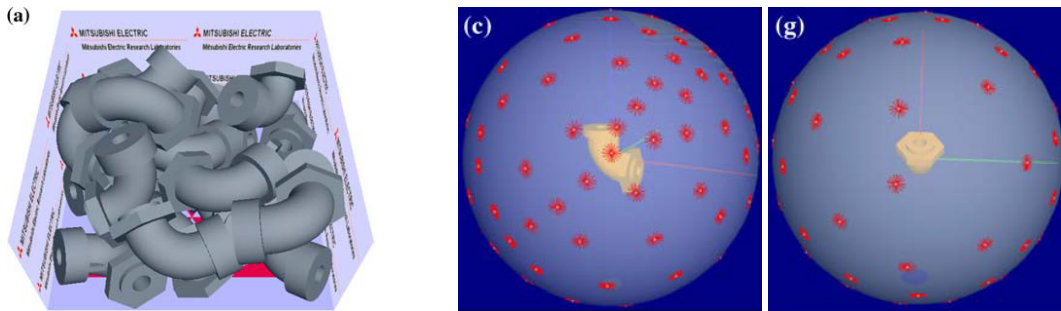
### 2.4.1 Recherche de patrons

La recherche de patrons – ou *template matching* – est sans doute une des approches les plus intuitives afin de détecter et estimer la pose d'instances d'objet. Celle-ci consiste à passer en revue différentes zones des données afin d'évaluer si celles-ci correspondent à un patron représentant l'objet dans une pose particulière, pour conclure le cas échéant à la détection d'une instance d'objet dans cette pose. La recherche de patrons constitue une approche fondamentale dans le domaine de la détection 2D d'instances d'objet, souvent mise en œuvre au moyen d'une approche par *fenêtre glissante* (*sliding window*) en déplaçant une fenêtre d'intérêt dans l'image 2D afin d'évaluer si celle-ci correspond ou non au patron recherché comme illustré figure 2.11.

**Recherche 2D** La recherche de patrons est typiquement mise en œuvre dans le cadre d'une modalité de données présentant une structure 2D, telle qu'une image RGB, RGBD ou une image de profondeur. L'objet d'intérêt est alors représenté par un ensemble de patrons couvrant les orientations potentielles de ce dernier, ainsi qu'éventuellement diverses échelles modélisant diverses distances de l'objet à la caméra. La détection d'un patron dans une image fournit dès lors non seulement une information concernant la position 2D de l'instance correspondante dans l'image, mais également son orientation et sa profondeur.

Si la recherche d'un unique patron dans une image peut être exécutée rapidement de manière exhaustive, le nombre de patrons à considérer afin de détecter des instances d'objet en des poses arbitraires est cependant suffisamment important pour rendre la recherche de l'ensemble de ceux-ci en un temps limité une tâche difficile. Il faut en effet de l'ordre de 8500 patrons afin de couvrir l'ensemble des orientations potentielles d'un objet 3D avec une résolution angulaire de  $15^\circ$  – relativement grossière –, et même 28000 avec une résolution de  $10^\circ$ <sup>6</sup>, valeur qui doit encore être multipliée

6. Il faut de l'ordre de  $3\pi/(4 \sin^3(\alpha/4))$  échantillons afin de couvrir l'espace des orientations 3D avec une dispersion inférieure à un angle  $\alpha/2$ . Valeur correspondant au nombre de régions de



(a) Simulation physique de scènes permettant d'apprendre la distribution de probabilité d'orientation de l'objet.

(b) Représentation de l'échantillonnage des patrons sur l'espace des orientations. Orientation 3D représentée en un point d'une sphère (azimuth, élévation) par un vecteur (rotation dans le plan image).

FIGURE 2.12 – Échantillonnage adaptatif des patrons selon la distribution de probabilité d'orientation de l'objet. Extrait de (Park et al., 2010).

par le nombre d'échelles prises en considération afin de représenter les variations de distance de l'objet à la caméra. Diverses approches ont donc été envisagées dans la littérature afin de rendre cette recherche praticable.

**Limitation de l'espace de recherche** Le temps de calcul global nécessaire à la recherche de patrons peut en effet être réduit au moyen d'une implémentation efficace de la recherche d'un patron, mais également en limitant l'espace de recherches de ces derniers.

Dans le cadre d'un scénario de dévracage, Park et al. (2010) exploitent ainsi une implémentation GPU de recherche de patrons dans des images de profondeur, tout en limitant le nombre de ceux-ci à 1024. Ils procèdent pour se faire à une analyse préalable de la distribution de probabilité d'orientation d'un objet par le biais de simulations physiques de génération de vracs de pièces, de façon à limiter la recherche aux orientations les plus probables, comme illustré figure 2.12. Cette recherche n'est de plus pas effectuée dans toute l'image, mais en privilégiant certaines zones au sommet du vrac, où les instances d'objet sont typiquement moins occultées et qui sont a priori plus intéressante en terme de préhension d'objet, afin de limiter la durée de calcul totale nécessaire – de l'ordre de 0.5s pour une recherche dans 3 zones d'une image  $128 \times 128$  en 2010.

Liu et al. (2012) mesurent quant à eux la similarité d'un patron avec une fenêtre d'une image à partir de la distance entre les contours détectés dans chacun. Ils introduisent pour se faire une distance de chanfrein permettant une évaluation de durée linéaire en le nombre de segments dont sont constitués les contours du patron, et réduisent l'espace de recherche en ne testant que des poses pour lesquelles un contour du patron et un contour de l'image sont alignés. Ulrich et al. (2012) procèdent également

l'espace des rotations 3D de rayon angulaire  $\alpha/2$  (représentées par des 3-boules de rayon  $\sin(\alpha/4)$  sur l'espace des quaternions unitaires, de volume  $4/3\pi \sin(\alpha/4)^3$ ) à considérer pour atteindre une mesure équivalente à l'aire totale de l'espace des orientations (qui correspond à la moitié d'une 3-sphère unité dans l'espace des quaternions unitaires, d'aire  $2\pi^2$ ).

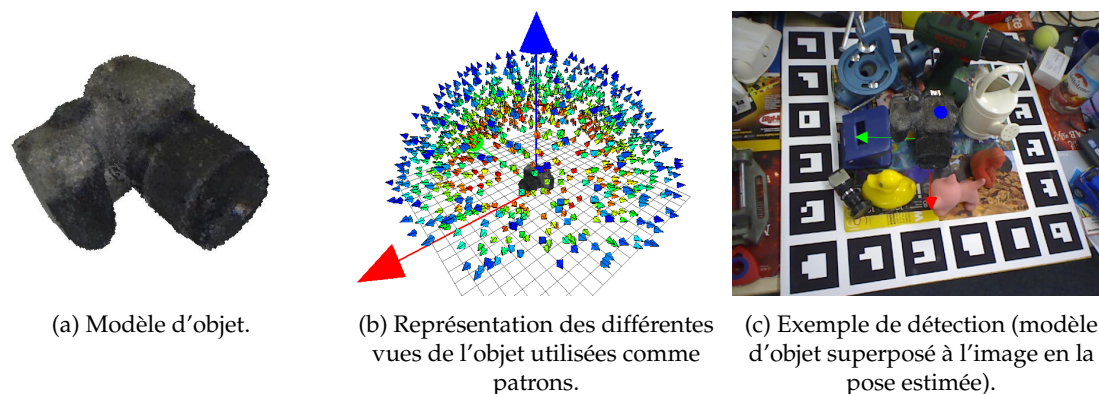


FIGURE 2.13 – Méthode de recherche de patrons LINEMOD. Extrait de (Hinterstoisser et al., 2012b).

à une recherche des contours de patrons dans une image, mais accélèrent celle-ci au moyen d'une approche hiérarchique multi-résolution, considérant initialement une faible résolution d'image ainsi qu'un petit nombre de patrons pour augmenter progressivement l'effort de recherche autour des détections potentielles.

Enfin, Hinterstoisser et al. (2012a) introduisent une approche connue sous le nom de *LINEMOD*, instituée depuis comme une référence dans la littérature. Celle-ci évalue également la vraisemblance de la présence d'un patron dans une fenêtre d'une image à partir des contours de ce patron, combinés à une mesure de similarité des normales 3D du patron et de l'imagette afin d'exploiter l'information fournie par un capteur RGBD. Contrairement aux approches précédentes, cette évaluation n'est cependant pas réalisée de manière dense, mais uniquement en quelques pixels pour des raisons de rapidité. Au moyen d'une implémentation optimisée pour tirer partie de l'architecture des machines modernes, mais également en réduisant l'espace de recherche d'un facteur  $64^7$  comparé à une recherche exhaustive par fenêtre glissante – grâce à une certaine invariance de leur mesure de similarité aux petits déplacements –, les auteurs parviennent à effectuer une recherche de 3000 patrons en 0.1s sur CPU dans une image VGA comme celle illustrée figure 2.13.

« **Scalabilité** » Afin d'accroître la résolution de l'estimation de pose, les dimensions de l'espace de recherche ou encore afin de considérer plusieurs objets simultanément, il est cependant nécessaire d'augmenter le nombre de patrons recherchés. Il devient alors essentiel que la recherche s'effectue avec une complexité sous-linéaire en le nombre de patrons.

Pour ce faire, Liu et al. (2012) décomposent leurs patrons en un ensemble de segments de contours, et évaluent la distance de ceux-ci aux contours observés dans un ordre permettant d'écarter rapidement les patrons présentant un écart trop important avec les données observées.

Ulrich et al. (2012) utilisent quant à eux une approche hiérarchique

7. Facteur 8 dans chaque dimension de l'image.

consistant à grouper les patrons représentant des points de vues proches d'un même objet et présentant un aspect similaire ensemble afin de réaliser la recherche avec un nombre initial limité de patrons puis et augmenter progressivement la résolution de recherche autour des zones où une instance d'objet a été détecté.

Cette idée de regrouper plusieurs patrons ensemble peut également être mise en œuvre de manière plus générale. [Rios-Cabrera et Tuytelaars \(2013\)](#) regroupent ainsi différents patrons d'un même objet en plusieurs ensembles, et procèdent à des tests sous forme de cascades de classifieurs afin d'écarter rapidement ceux dont aucun élément ne correspond à la fenêtre de l'image analysée. Les patrons des ensembles restants sont alors évalués de manière classique. Ce pré-filtrage permet selon les auteurs de réduire la durée d'exécution par 10 comparé à la méthode LINEMOD originale lorsqu'il s'agit de rechercher 15 objets simultanément, sans dégrader les performances de localisation d'objet. Leur approche reste cependant toujours linéaire en le nombre d'objets considérés, limitation que dépassent [Kehl et al. \(2015\)](#), au moyen d'une approche similaire ne testant qu'un nombre limité de patrons pour chaque fenêtre glissante, sélectionnés selon la sortie d'une fonction de hachage, au même titre que [Hodaň et al. \(2015\)](#).

[Cao et al. \(2016\)](#) proposent quant à eux de réduire la complexité de l'évaluation de la similarité d'une fenêtre de l'image avec un grand nombre de patrons simultanément. Ils considèrent une mesure de similarité par corrélation croisée, et expriment le calcul de celle-ci sous forme d'une multiplication matricielle entre l'ensemble des patrons et la fenêtre de l'image. La multiplication est effectuée sur GPU après décomposition de la matrice des patrons et réduction de la dimension de ses composantes par *Analyse en Composantes Principales*, de sorte de rendre ce calcul moins coûteux. Si l'estimation de la similarité pour chaque patron s'avère alors effectivement d'une complexité sous-linéaire en temps dans leurs expériences – voyant par exemple la durée d'exécution doubler lorsque le nombre d'objets recherchés passe de 1 à 15 –, cette performance est cependant quelque peu artificielle en ce qu'elle est avant tout due aux capacités de parallélisme du GPU utilisé. Ces dernières sont d'ailleurs limitées à la recherche d'environ 27000 patrons dans une image  $320 \times 240$  d'après les auteurs, pour des raisons de mémoire.

**Recherche 3D** La recherche 2D de patrons reste dépendante du point de vue et est dès lors sensible aux occultations même partielles de l'objet recherché, qui réduisent la similarité des données avec le patron correspondant. [Song et Xiao \(2014\)](#) proposent une approche intéressante concernant ce problème qui consiste non pas à effectuer la recherche de patrons dans une image mais au moyen d'une « boîte 3D glissante » dans l'ensemble de la scène représentée par un nuage de points 3D, de façon à évaluer si le contenu de cette boîte correspond ou non à une instance d'objet dans une pose donnée. Cette évaluation est réalisée au moyen d'un classifieur entraîné spécifiquement pour chaque patron de pose. Leur implémentation requière cependant plus de 20 minutes afin de rechercher un objet dans des données RGBD, tout en étant inscrite dans le contexte particulier d'objets reposant toujours sur un sol plan sur une même face, qui présente un es-

pace de recherche de dimension bien moindre que le cas général qui nous intéresse.

### 2.4.2 Détection et estimation de pose

Une autre approche potentielle du problème consiste à procéder en deux étapes, en localisant les zones représentant des instances d'objet dans les données (*détection*), et en estimant pour ces zones la classe ainsi que la pose de l'objet correspondant (*estimation de pose*).

#### 2.4.2.1 Détection d'instance

La stratégie de détection d'instances est une tâche relativement dépendante du type de scène considérée. Un scénario d'importance dans la littérature robotique consiste en le cas d'instances d'objets posées sur une table ou un plan horizontal de manière non contiguë, et en ce cas ces instances peuvent être segmentées relativement simplement dans des données 3D en procédant à une détection de plan (Rusu et al., 2010; Wang et al., 2013). Agrawal et al. (2010) segmentent quant à eux des instances d'objet posées sur une surface lisse dans une image à partir de la détection de ses contours au moyen d'un dispositif d'acquisition spécifique. Cette séparation entre détection et estimation de pose n'est cependant que rarement mise en œuvre dans la littérature pour des scènes plus complexes présentant un certain fouillis et où les instances sont occultées dans le cas d'objets rigides, car ces deux tâches sont intimement liées<sup>8</sup>. Rad et Lepetit (2017) emploient néanmoins une telle approche, et utilisent un réseau de neurones convolutif afin de segmenter l'objet d'intérêt dans une image RGB, pour estimer la pose de ce dernier dans un second temps. Afin d'apprendre à leur réseau le comportement attendu dans le cadre de telles scènes, ces derniers utilisent des données annotées, augmentées avec des images synthétiques. Gupta et al. (2015) procèdent de manière similaire pour classifier et détecter la pose de mobilier dans des images RGB à partir de la sortie d'un réseau de neurones de détection et de segmentation d'instances d'objet existant.

#### 2.4.2.2 Estimation de pose globale

Une fois une instance d'objet localisée sa position est approximativement connue. Aussi l'estimation de sa pose se limite-elle typiquement à l'estimation de son orientation, ainsi que sa classe dans le cas où plusieurs types d'objets seraient considérés.

**Classification par plus proche voisin** Une manière de réaliser cette estimation consiste à rechercher parmi une base de données la pose la plus similaire à la pose courante. Cette recherche de similarité peut être réalisée de manière exhaustive comme le font par exemple Heber et al. (2010) en comparant le contour de la silhouette de l'objet avec ceux d'une base

---

8. Des approches étagées sont cependant typiquement utilisées pour d'autres problèmes tels que la reconnaissance faciale, en procédant par détection des visages suivie d'une reconnaissance de celui-ci (par exemple par détection de certains points clés).

de données représentant différents points de vue, ou de manière similaire aux techniques de recherche de patrons décrites section 2.4.1. La détection préalable permet cependant d'envisager également d'autres approches.

**Descripteur global de pose** Une de ces approches consiste à synthétiser les données disponibles concernant l'instance localisée sous la forme d'un descripteur global, qui offre une représentation compacte permettant des calculs plus efficaces. En 1983, [Horn et Ikeuchi \(1983\)](#) décrivent ainsi un objet segmenté par l'histogramme d'orientation de ses normales, et recherchent parmi une base de données le descripteur présentant la corrélation la plus importante avec cet histogramme.

Les descripteurs utilisés sont généralement conçus sous la forme d'un vecteur de dimension fixe, et la similarité entre descripteurs est typiquement estimée à partir de la distance  $L_1$  ou  $L_2$  entre ceux-ci, pour lesquelles des outils de recherches de plus proches voisins efficaces ont été développés ([Muja et Lowe, 2009](#)). Ce passage à une représentation sous forme de descripteur a alors pour but de mettre en évidence une structure interne aux données, de sorte que des données correspondant à des poses similaires produisent des descripteurs similaires, tandis que des poses éloignées devraient idéalement présenter des descripteur éloignés. L'apprentissage d'une relation de passage adaptée entre données et descripteur constitue un champ important de la recherche en intelligence artificielle, connu sous le nom de *manifold learning*. Dans le cadre de notre sujet d'étude, [Wohlhart et Lepetit \(2015\)](#) ont notamment proposé d'apprendre un tel plongement d'une vue RGB d'un objet vers un descripteur compact de dimension 16 – illustré figure 2.14 – et obtiennent ainsi de meilleurs performances en classification d'objet et de pose qu'en estimant la similarité avec la méthode LINEMOD évoquée section 2.4.1. [Balntas et al. \(2017\)](#) ont montré qu'il est possible d'améliorer encore ces performances en introduisant lors de l'apprentissage un objectif plus explicite de préservation des propriétés métriques de l'espace d'origine – de sorte que la distance entre descripteurs corresponde à une mesure de similarité entre poses définie manuellement.

Avant l'avènement de l'apprentissage profond, un tel plongement était réalisé aux moyens de descripteurs globaux choisis à la main pour leurs caractéristiques. [Agrawal et al. \(2010\)](#) représentent ainsi la silhouette d'un objet par un descripteur de dimension 72<sup>9</sup>, correspondant aux moments principaux issus de sa projection sur la base des polynômes de Zernike. D'autres introduisent des descripteurs globaux de formes à partir de nuages de points et de normales tels que le *Viewpoint Feature Histogram* de [Rusu et al. \(2010\)](#), et sa variante exploitant une information de couleur additionnelle ([Wang et al., 2013](#)). L'absence d'apprentissage spécifique au moyen de données réelles rend cependant ces descripteurs globaux sensibles aux erreurs de segmentation ainsi qu'aux problèmes d'occultation, bien que certains auteurs comme [Aldoma et al. \(2011, 2012\)](#) proposent d'y introduire une certaine robustesse en procédant à une sur-segmentation.

**Classifieur générique** La détermination de la pose la plus vraisemblable vis à vis des données peut également être réalisée au moyen d'une méthode

9. 36 coefficients complexes pour être exact.

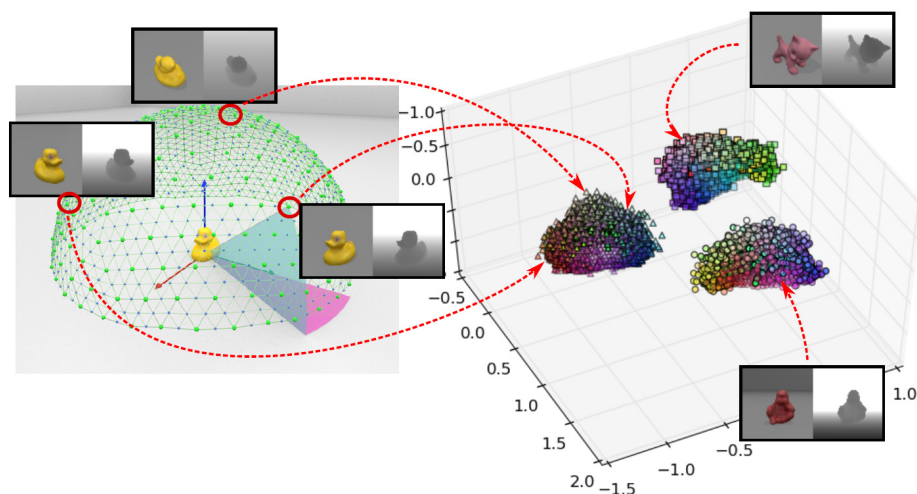


FIGURE 2.14 – Descripteurs globaux 3D de pose et de classe d’objet (représentés à droite) estimés à partir de données RGBD selon une méthode entraînée afin de dissocier les descripteurs d’objets distincts et préserver les propriétés topologiques de l’espace des poses, de sorte que des descripteurs soient d’autant plus proches qu’ils représentent des poses semblables. Extrait de (Wohlhart et Lepetit, 2015).

de classification générique telle qu’un réseau de neurones convolutif, et Gupta et al. (2015) procèdent de la sorte pour estimer grossièrement l’orientation 1D d’un meuble dans une photographie RGBD de pièce, parmi un ensemble de 20 classes. Le nombre de classes à considérer est cependant beaucoup plus important dans le cas général d’une orientation 3D – de l’ordre de 28000 avec une résolution angulaire de  $10^\circ$  ainsi que nous l’avons évoqué section 2.4.1. Il est toutefois possible de procéder avec un nombre de classes bien inférieur, à l’instar de Kehl et al. (2017) qui décomposent l’orientation 3D d’un objet vu dans une image en deux termes qu’il s’agit de recouvrer. Ces derniers classifient en effet d’une part le point de vue de l’objet, et d’autre part l’orientation de celui-ci dans le plan image de la caméra, ne nécessitant par là que 337 classes pour échantillonner les points de vue potentiels de l’objet et 19 classes pour son orientation dans le plan image dans le cas spécifique de leur application<sup>10</sup>. Ces chercheurs procèdent de plus de manière simultanée à la détection et à l’estimation de pose des instances d’objet dans une image, à l’aide d’un réseau de neurones convolutif estimant pour un ensemble de boîtes 2D « ancrées » dans l’image la classe des instances d’objets délimitées par celles-ci (pouvant également correspondre à une absence d’objet recherché) ainsi que leurs poses. Cette approche s’appuie sur les développements récents en détection et en reconnaissance d’objets dans des images et plus spécifiquement le *Single Shot multibox Detector* de Liu et al. (2016) illustré figure 2.15, qui réalise la détection et la reconnaissance de multiples instances d’objet en une seule passe et pour différentes échelles. La méthode de Kehl et al. (2017) s’exécute en moins

10. Pour laquelle  $1/8$  de l’espace des orientations 3D est admissible.



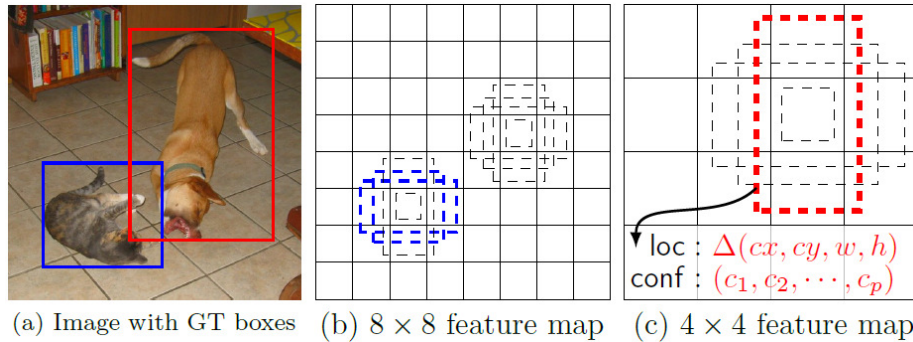


FIGURE 2.15 – Illustration de la méthode SSD, réalisant détection – sous forme de boîte englobante – et reconnaissance d’instances d’objet (ici chien et chat) à différentes échelles au moyen d’un même réseau de neurones entraîné de bout-en-bout. Extrait de (Liu et al., 2016).

de 100ms sur un GPU haut de gamme pour une image  $299 \times 299$  et présente de bons résultats sur les jeux de données de Hinterstoisser et al. (2012b) et Tejani et al. (2014) comparé à l’état de l’art, à partir d’un apprentissage avec des données synthétiques.

**Régression de pose** La notion de pose d’un objet 3D est une notion continue et présentant une topologie particulière, que l’on peut exprimer en munissant l’espace des poses d’une métrique (un état de l’art spécifique sur ce point est proposé chapitre 3). Cette structure permet d’aller au delà de la simple classification de pose évoquée précédemment, en permettant d’exprimer lors de l’apprentissage des objectifs tels que la préférence d’une faible erreur de pose à une erreur plus importante<sup>11</sup>. La continuité de l’espace de pose permet de plus d’envisager de *régresser* la pose d’une instance d’objet, c’est à dire d’en proposer une estimation plus précise qu’une simple relation d’appartenance à une classe. De manière usuelle, l’apprentissage d’une méthode de régression consiste à définir un paramétrage de l’espace de pose, et à apprendre à prédire à partir des données le paramétrage de la pose que celles-ci représentent. Kendall et al. (2015) prédisent ainsi la pose d’une caméra dans un environnement appris au préalable sous forme d’un vecteur de position 3D et d’une représentation de son orientation sous forme d’un quaternion unitaire, et Novotny et al. (2017) estiment de même l’orientation d’un objet présentant une certaine variabilité intraclasse telle qu’une voiture. La pose d’un objet non rigide présente d’avantage de degrés de libertés et en ce cas d’autres paramétrisations doivent être envisagées, la plus courante consistant à prédire la position 2D ou 3D de points clés de l’objet, tels que celle des articulations dans le cas d’une main (Oberweger et al., 2015).

**Régression probabiliste** La prédiction d’une pose unique ne permet cependant pas de modéliser les ambiguïtés et incertitudes vis à vis de celle-ci.

11. Propriété qui est exploitée dans les approches de *manifold learning* abordées précédemment.

Aussi, beaucoup d'approches de la littérature réalisent cette prédiction d'une manière *douce* sous forme d'une *régression probabiliste*, en prédisant non pas la pose de l'objet mais une distribution représentant la probabilité de chaque pose sur l'espace des poses admissibles. Il s'agit là d'un formalisme répandu et tout particulièrement dans le domaine de la classification où celle-ci est alors réalisée en prédisant une distribution de probabilité à  $n \in \mathbb{N}^*$  classes. Cette approche douce de la classification est selon nous plus favorable à l'apprentissage qu'une approche dure, en permettant d'exprimer un objectif d'une manière moins sensible aux optima locaux. [Bonde et al. \(2014\)](#) procèdent de la sorte suivant une approche de type boîte 3D glissante, prédisant pour chaque boîte 3D la probabilité que l'objet y soit localisé et la distribution de probabilité concernant son orientation, quantifiée en 16 classes, au moyen d'une forêt de décision.

Afin d'utiliser une résolution de pose plus importante tout en limitant le nombre de classes à considérer, la régression probabiliste n'est souvent réalisée que de manière partielle, en estimant individuellement la distribution de probabilité de composantes d'un paramétrage de pose. [Kehl et al. \(2017\)](#) estiment ainsi d'une part le point de vue sous lequel est observé un objet, et d'autre part l'orientation 2D de cette vue dans le plan image par classification douce. C'est également la démarche typiquement suivie lors de l'estimation de pose par prédiction de la position de points clés, à l'instar de [Cao et al. \(2016\)](#) qui estiment la pose d'un squelette humain en produisant une distribution 2D de probabilité pour chaque point clé (genou gauche, épaule droite, etc.). [Crivellaro et al. \(2015\)](#) puis [Rad et Lepetit \(2017\)](#) proposent d'utiliser une approche similaire pour estimer la pose 3D d'un objet rigide dans une image, en prédisant sous forme de distributions 2D de probabilité la position dans l'image des coins d'une boîte englobante 3D de l'objet, pour recouvrer par la suite la pose correspondante par résolution d'un problème de correspondance de points 2D-3D.

### 2.4.3 Fond et occultations

Ainsi que leur nom l'indique, les approches globales exploitent des données censées représenter l'intégralité de l'objet. Les instances d'objet sont cependant parfois partiellement occultées, et à moins d'une segmentation préalable, celles-ci ne sont de plus généralement pas isolées du reste de la scène, que l'on désignera ici sous le terme de *fond*. Pour ces raisons, les approches globales présentent intrinsèquement une sensibilité importante au fond et aux occultations. À titre d'exemple, la mesure de similarité estimée par la méthode LINEMOD ([Hinterstoisser et al., 2012a](#)) entre un patron et la vue d'une instance d'objet correspondante décroît linéairement avec l'augmentation du taux d'occultation de celle-ci, rendant la détection d'instances partiellement occultées difficile.

Afin de présenter une certaine robustesse, les méthodes basées sur des techniques d'apprentissage automatique doivent dès lors apprendre à présenter une certaine invariance vis-à-vis du fond et des occultations. [Bonde et al. \(2014\)](#), [Rad et Lepetit \(2017\)](#) et [Baltas et al. \(2017\)](#) utilisent pour ce faire des données d'apprentissage synthétiques ou réelles représentant divers types d'occultation et de fond.

En l'absence de tels exemples, [Tejani et al. \(2014\)](#) proposent quant à eux d'apprendre à distinguer entre patches 2D représentant l'objet et patches 2D représentant le fond, au moyen d'images réelles non annotées. Ils procèdent pour ce faire par *co-apprentissage*, au moyen de deux classifieurs dont la sortie est utilisée itérativement afin d'entraîner l'autre. Il s'agit là d'une approche non supervisée intéressante, qui repose cependant sur l'apriori d'un nombre limité d'instances d'objet visibles dans ces données.

## 2.5 Approches locales

Si les approches globales recherchent l'objet dans son intégralité au sein des données, les approches locales procèdent quant-à-elle en identifiant localement des zones de l'objet. Bien que ces dernières ne puissent exploiter autant d'information que les approches globales, les approches locales présentent implicitement une certaine robustesse aux problèmes de fond et d'occultation, puisque l'identification d'un nombre limité de zones d'une instance d'objet peut suffire à estimer sa pose. On distingue parmi ces méthodes locales deux catégories d'approches.

La première consiste à rechercher parmi ces zones identifiées des relations similaires à celles observées dans un modèle d'objet, de sorte à produire à partir de ces relations une hypothèse concernant la pose d'une instance. Cette approche a été largement développée jusque dans les années 1990 en travaillant à partir de primitives usuelles – point que nous abordons section 2.5.1. Elle a ensuite majoritairement évolué vers l'usage de zones d'intérêts plus génériques telles que des points d'intérêts, que nous abordons section 2.5.2.

La seconde classe d'approches consiste quant à elle à produire directement des hypothèses concernant la pose d'une instance d'objet à partir de la zone locale identifiée. Ces prédictions sont cependant généralement fortement bruitées, aussi ces dernières sont-elles agrégées de façon à identifier les poses pour lesquelles un consensus se crée. Cette démarche est présentée section 2.5.3.

### 2.5.1 Recherche de sous-graphes de primitives

Il est possible de décrire la géométrie d'un objet à partir d'un ensemble de primitives – surface plane ou sphérique, arrête droite ou courbe, etc.– positionnées les unes par rapport aux autres de manière fixe dans le cas d'instances d'objet rigide. Une approche potentielle de détection et d'estimation de pose d'instances d'un tel objet consiste donc à détecter de telles primitives au sein des données, et identifier parmi celles-ci des groupes de primitives respectant les mêmes relations que celles observées chez l'objet d'intérêt, de manière à produire des hypothèses de pose.

Cette approche a dominé le domaine jusqu'au milieu des années 1980 avec un nombre important de travaux sur le sujet. À titre d'exemple, [Bolles et Horaud \(1986\)](#) considèrent comme primitives des bords droits ou circulaires détectés dans une image de profondeur, et [Rahardja et Kosaka \(1996\)](#) des arrêtes circulaires. [Mundy et al. \(1994\)](#) détectent et localisent des polyèdres

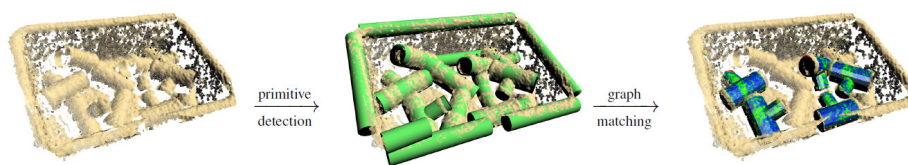


FIGURE 2.16 – Recherche d’instances d’objet par recherche d’un ensemble de primitives géométriques en relation les représentant (ici un assemblage de deux cylindres). Extrait de (Nieuwenhuisen et al., 2013).

et des surfaces de révolution au sein d’images 2D à partir des bords détectés. [Costa et Shapiro \(2000\)](#) recherchent de même des caractéristiques au sein d’une image 2D – segments, ellipses – ainsi que leurs relations pour estimer la pose d’objets. [Kak et Edwards \(1995\)](#) extraient en plus des surfaces planes, cylindriques, sphériques et coniques segmentées dans une carte de profondeur, et [Oshima et Shirai \(1983\)](#) procèdent par segmentation de surfaces planes ou courbes dans des données 3D. Plus récemment, [Nieuwenhuisen et al. \(2013\)](#); [Holz et al. \(2014\)](#) ont proposé une application de dévissage robotisé où des instances d’objet, vues comme assemblages de primitives telles que des cylindres, sont localisées dans un nuage de points 3D, et qui est illustrée figure 2.16. Une revue exhaustive serait fastidieuse et d’un intérêt limité pour notre propos, et le lecteur intéressé par les approches historiques est invité à se référer aux états de l’art de [Besl et Jain \(1985\)](#) et [Chin et Dyer \(1986\)](#).

Une fois des primitives détectées, la production d’une hypothèse de pose revient alors à chercher un sous-graphe au sein du graphe composé de l’ensemble de ces primitives et de leurs relations. Pour des raisons pratiques, cette recherche est usuellement réalisée de manière hiérarchique, en sélectionnant une primitive comme point de départ et en cherchant à grouper celle-ci avec d’autres primitives compatibles selon le modèle de l’objet. Il est alors possible de conjecturer la pose d’une instance d’objet à laquelle appartiendraient ces primitives, trois plans orthogonaux étant par exemple suffisants pour estimer la pose d’un cube.

Une limitation fondamentale de ces méthodes tient cependant au fait qu’elles ne sont applicables qu’à des objets pouvant être décrits au moyen des primitives considérées. Or, de nombreux objets ne peuvent pas être décrits simplement à partir de primitives simples telles que des plans, sphères, droites, etc. L’usage de primitives possédant un pouvoir descriptif plus grand telles que des NURBS est difficilement envisageable en pratique car il supposerait d’être en mesure de décomposer le modèle de l’objet ainsi que les données en un ensemble de telles primitives de manière répétable et fiable, ce qui constitue un challenge important, et explique que ces méthodes aient globalement été délaissées sauf cas particulier.

Les travaux de [Damen et al. \(2012\)](#) constituent cependant une exception intéressante, dans lesquels ces derniers détectent un objet dans une image 2D à partir de ses contours. Les contours de l’image 2D sont pour ce faire extraits et représentés sous forme d’un ensemble de petits segments appelés *edgelets*, et il s’agit d’identifier parmi ceux-ci un ensemble d’edgelets qui correspondent à ceux d’une vue d’un objet. Ces edgelets sont consi-

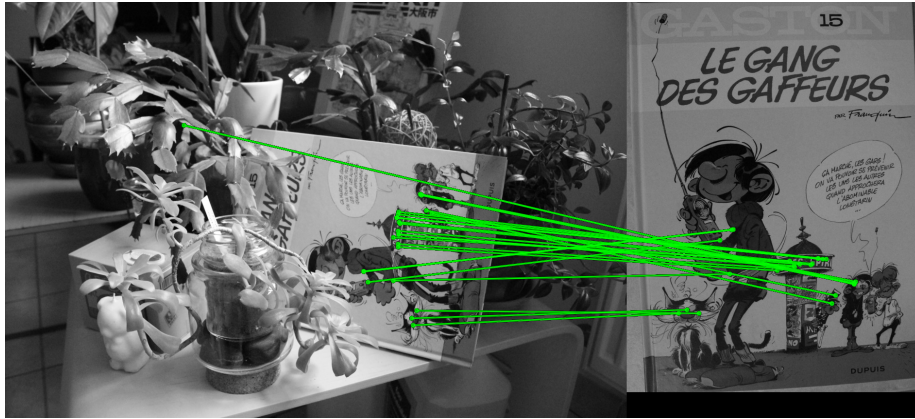


FIGURE 2.17 – Appariement des points d'intérêt extraits d'une scène (à gauche) et d'un modèle (à droite), ici au moyen de la méthode ORB (Rublec et al., 2011).

dérés en relations sous forme de *constellations*, représentant chacune une suites d'edgelets positionnés l'un par rapport à l'autre dans une position et une orientation donnée. Un ensemble de ces constellations sont apprises pour chacune des différentes vues des différents objets, et recherchées dans l'image de test afin de produire des hypothèses de détection. Au moyen d'une implémentation efficace, Damen *et al.* parviennent en moins de 200ms à obtenir des performances de détection convaincantes avec une base de données de 30 objets dans des images présentant un nombre limité de contours. Le succès de leur approche tient sans doute dans le grand nombre de primitives considérées – les contours étant découpés en nombreux edgelets de petites tailles – qui introduit une redondance importante des relations entre primitives, et facilite ainsi une détection relativement robuste malgré la répétabilité imparfaite de l'extraction des contours.

### 2.5.2 Appariement de points d'intérêt

Devant la difficulté de décomposer de manière non ambiguë tout type d'objet en un ensemble de primitives de haut niveau, les chercheurs se sont tournés vers l'utilisation de primitives plus simples et plus discriminantes, sous la forme de points d'intérêts associées à des descripteurs.

Cette approche – que nous qualifions d'*appariement de points d'intérêt* – consiste à identifier des régions locales d'intérêt dans les données et caractériser celles-ci de manière à identifier les zones correspondantes sur un modèle. A partir d'un ensemble de tels appariements illustrés figure 2.17, il est alors possible d'estimer une transformation permettant de recaler ces régions deux à deux, ce qui dans le cas d'une instance d'objet rigide peut définir sa pose.

**Point d'intérêt et descripteur 2D** Tant pour des raisons de performances que de robustesse, l'appariement de caractéristiques locales est généralement réalisé de manière parcimonieuse uniquement pour un ensemble de

points dits *d'intérêt* qu'il est possible d'extraire de manière relativement répétable dans les données. Ces points d'intérêt correspondent typiquement à des éléments saillants, tels que des coins dans le cas d'une image 2D. Divers détecteurs de points d'intérêts ont été conçus au fil des ans, tels que le détecteur de coin de Harris et Stephens (1988), SIFT (Lowe, 1999), ou encore SURF (Bay et al., 2006), avec pour objectif d'extraire de manière rapide des points caractéristiques de manière la plus robuste possibles aux variations potentielles de conditions : changements d'échelle, effets de perspective, changements de luminosité, flou, etc. Il est alors possible de caractériser un point d'intérêt au moyen d'un descripteur tel que ceux de SIFT, SURF ou encore le descripteur binaire ORB (Rublee et al., 2011) de manière à identifier les points d'intérêts similaires extraits du modèle, par recherche de plus proche voisins à l'instar de l'approche utilisée avec des descripteurs globaux, évoqués section 2.4.2.2. De même que dans les autres domaines, des chercheurs ont expérimenté avec succès l'apprentissage automatique afin d'identifier de manière robuste des points d'intérêts dans des images, à l'instar de Lepetit et al. (2005) et Ozuysal et al. (2010) qui procèdent par classification de patches au moyen respectivement d'une forêt de décision et de ferns. Holzer et al. (2012) apprennent quant à eux à détecter des points d'intérêt dans des images de profondeur d'une manière robuste aux mouvements de la caméra, à l'aide d'une forêt de régression.

Cette détection et identification de points 2D dans des images constitue une approche standard exploitée dans des domaines tels que la constitution d'images panoramiques ou encore la *Structure From Motion* où il s'agit d'estimer la pose d'une caméra en différentes acquisitions et de reconstruire un modèle de l'environnement. Elle nécessite cependant la présence d'une certaine texture dans la scène de sorte que des points d'intérêts visuellement saillants soient identifiables, or nous nous intéressons particulièrement dans cette thèse au cas de scènes d'objets non texturés.

**Descripteur 3D** En parallèle de ces méthodes 2D ont été développés des approches adaptées aux données 3D basées sur ce même principe d'appariement de points d'intérêt des données avec celles d'un modèle.

Les travaux de Johnson et Hebert (1997, 1999) ont dans ce domaine une importance historique. À partir de données 3D d'une scène sous forme de surfaces 3D maillées reconstruites, ces derniers estiment en un ensemble de points choisis aléatoirement dans la scène des descripteurs invariants aux transformations rigides appelés *spin-images*, basés sur un histogramme 2D de distances des points du voisinage au point d'intérêt. L'appariement des descripteurs extraits par recherche de plus proche voisin au sein d'une bibliothèque de descripteurs apprise sur des modèles d'objets permet alors de générer des hypothèses de reconnaissance d'objet et de pose, qui seront par la suite vérifiées. D'autres descripteurs ont depuis été proposés tels que le *Fast Point Feature Histogram* de Rusu et al. (2009) conçu pour être estimé de manière rapide et robuste aux changements d'échantillonnage, ou la *Signature of Histograms of Orientations* de Tombari et al. (2010) présentant une meilleure robustesse au bruit que les *spin-images*.

**Appariement automatique** Au lieu d'utiliser un descripteur de point, [Brachmann et al. \(2014\)](#) choisissent d'apprendre directement et de manière automatique à estimer pour un point des données (sous forme d'une image RGBD) la vraisemblance que celui-ci appartienne à une instance objet recherchée, ainsi que la position 3D de ce point dans le repère objet. Ils utilisent pour ce faire une forêt de décision, entraînée à partir d'images synthétiques générées à partir d'un modèle CAO et d'images de fonds génériques, et produisent par cette méthode un ensemble d'appariements de points entre la scène et un modèle d'objet, permettant de générer des hypothèses de pose. Leur approche est reprise dans les travaux ultérieurs de [Krull et al. \(2015\)](#); [Brachmann et al. \(2016\)](#).

**Génération d'hypothèses de pose** À partir d'un ensemble de correspondances entre des points d'une scène appartenant à une instance d'objet et les points 3D correspondants d'un modèle, il est possible d'estimer une transformation rigide permettant de recalculer ceux-ci deux à deux et ainsi estimer la pose de l'objet. Il est pour ce faire nécessaire de considérer a minima :

- 3 correspondances 3D-3D, pouvant être recalculées au sens des moindres carrés au moyen de la méthode décrite par [Umeyama \(1991\)](#).
- 4 correspondances 2D-3D entre points d'une image d'une caméra calibrée et points 3D. Le recalage peut alors être estimé de manière non ambiguë au moyen notamment de la méthode de [Lepetit et al. \(2009\)](#)<sup>12</sup>.

La détection et l'estimation de pose d'instances d'objets à partir de tels appariements n'en est cependant pas triviale. Ces appariements ne sont en effet pas toujours corrects du fait du pouvoir discriminant limité des descripteurs. De plus, on ignore quels points d'intérêts appartiennent à une même instance d'objet. Aussi, la solution usuellement mise en œuvre consiste-elle à procéder par essais et erreurs, à la manière du paradigme RANSAC (*RANdom SAmple Consensus*) introduit par [Fischler et Bolles \(1981\)](#) pour estimer un modèle de manière presque certaine en présence de valeurs aberrantes. Ce dernier consiste à sélectionner un petit nombre de correspondances supposées correctes et appartenant à une même instance d'objet pour produire une hypothèse de pose, vérifier cette hypothèse et itérer un nombre de fois suffisant afin d'accroître la probabilité de produire une hypothèse pertinente. La sélection des groupes de correspondances testés peut être réalisée selon leur cohérence géométrique ([Johnson et Hebert, 1997](#)) – et la probabilité que les points sélectionnés appartiennent à un même objet dans le cas de [Brachmann et al. \(2014\)](#) –, de manière à limiter la probabilité de sélectionner une paire incorrecte et ainsi le nombre de tests à réaliser.

Cette approche de génération d'hypothèses puis de vérification est également exploitée avec d'autres techniques, comme celles proposées par [Mian et al. \(2006\)](#) ou [Buchholz et al. \(2010\)](#) qui produisent des hypothèses de pose par mise en correspondance d'une paire de points 3D et leurs normales de

<sup>12</sup>. 3 correspondances 2D-3D sont en réalité suffisantes mais il n'y a alors en général pas unicité de la solution.

la scène avec une paire de points/normales d'un modèle, selon la similarité de descripteurs spécifiques.

### 2.5.3 Aggrégation d'hypothèses

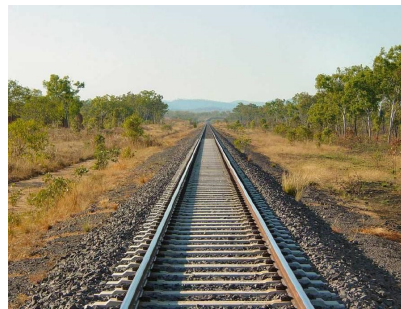
Les techniques vues précédemment génèrent différentes hypothèses de pose et testent celles-ci afin d'identifier celles pertinentes. Dès lors que ces hypothèses sont produites d'une manière au moins partiellement indépendante, il peut cependant être pertinent d'agréger ces hypothèses afin d'identifier si des consensus existent autour de certaines poses et n'évaluer que ces dernières, qui représentent des hypothèses plus probables que les hypothèses initiales.

**Accumulation d'hypothèses et transformée de Hough** On associe généralement le nom de Paul V. C. Hough aux approches d'accumulation d'hypothèses. Il y a près de 60 ans, celui-ci proposait un dispositif destiné à automatiser l'analyse des trajectoires de particules observées dans une photographie de chambre à bulles (Hough, 1959, 1962). Une trajectoire était localement modélisée dans une petite partie de l'image sous forme d'une droite, et l'invention de Hough consistait alors en un dispositif permettant d'estimer les paramètres de cette droite (ordonnée à l'origine et pente) afin de les donner en entrée à un ordinateur. Cette estimation s'effectuait au moyen d'une *transformée* consistant à accumuler pour chaque bulle détectée un ensemble de votes représentant l'ensemble des droites pouvant passer par cette bulle, dans un accumulateur défini sur l'espace des paramètres<sup>13</sup>. Les votes produits pour chaque bulle représentent des hypothèses extrêmement incertaines cependant ceux-ci doivent théoriquement s'accorder en les paramètres « vrais » de la droite, qu'il est alors possible de détecter sous forme d'un pic au sein de la transformée de Hough. La figure 2.18 propose une illustration de ce procédé de détection de droite utilisant une paramétrisation polaire, suggérée par Duda et Hart (1972). L'idée derrière cette transformée a depuis été généralisée, et on parle aujourd'hui de méthode de type Hough pour qualifier toute approche procédant à une accumulation de votes dans un espace de paramètres afin de détecter des formes en vision par ordinateur.

**Point Pair Features de Drost et al.** La méthode de détection et d'estimation de pose de Drost et al. (2010) constitue un bon exemple d'aggrégation d'hypothèse. À partir de données de la scène sous forme d'un nuage de points, les auteurs considèrent des paires de points et leurs normales associées. Ils en extraient des descripteurs simples synthétisant des informations géométriques telles que la distance entre ces deux points ou l'angle entre leurs normales, et cherchent des paires de points de descripteurs similaires parmi une base de données apprise au préalable sur un modèle d'objet afin de produire des hypothèses de pose. En cela, leur méthode est similaire à la méthode employée par Buchholz et al. (2010) évoquée précédemment. Les descripteurs utilisés ont cependant un pouvoir discriminant relativement

13. Accumulation réalisée à l'époque sur l'écran d'un tube cathodique.

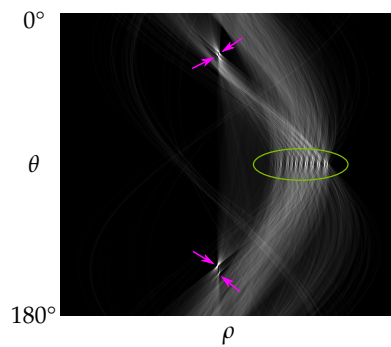
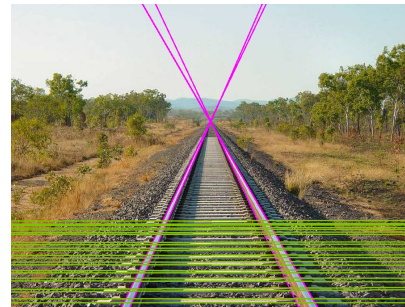




(a) Image d'origine.



(b) Pixels à partir desquels détecter des lignes.

(c) Transformée de Hough suivant une paramétrisation polaire de droite  $(\rho, \theta)$ .

(d) Lignes détectées, correspondant à des pics de la transformée.

FIGURE 2.18 – Détection de lignes dans une image par identification de pics dans la transformée de Hough.

faible aussi les hypothèses produites sont relativement bruitées. Les auteurs agrègent donc ces dernières afin d'obtenir un nombre limité d'hypothèses de pose plus pertinentes, dont la validité pourra être testée par la suite. Pour des raisons d'efficacité, cette agrégation est réalisée en deux étapes. L'ensemble des hypothèses produites par les paires de points partageant un même point de référence en commun sont dans un premier temps agrégées afin de prédire à quel point du modèle ce point de référence correspond et comment l'objet est orienté par rapport à celui-ci, au moyen d'un accumulateur 2D de type Hough. Ces prédictions locales sont ensuite traduites sous forme d'hypothèses de pose qui sont elles-même agrégées afin d'identifier des groupes d'hypothèses semblables, qui constituent alors les hypothèses finales qui seront testées et évaluées afin de produire le résultat.

Cette approche permet à leur méthode de dépasser les performances obtenues par [Mian et al. \(2006\)](#) qui utilisent pourtant un descripteur plus discriminant mais sans agrégation de votes, tout en permettant d'atteindre une durée d'exécution de l'ordre de la seconde sur un matériel grand public. [Drost et Ilic \(2012\)](#); [Choi et Christensen \(2012\)](#); [Choi et al. \(2012\)](#) ont étendu cette approche aux données RGB-D, afin d'incorporer des informations de couleur et de contours au détriment de l'invariance de leur descripteur aux transformations rigides et du temps de calcul. [Kim et Medioni \(2011\)](#) ont quant à eux proposé d'incorporer au descripteur une information quant à la visibilité du segment reliant les paires de points considérées et réalisent l'agrégation de votes en une seule étape, privilégiant ainsi la qualité des résultats à la rapidité. Plus récemment, [Birdal et Ilic \(2015\)](#) et [Hinterstoisser et al. \(2016\)](#) ont proposé diverses optimisations à la méthode d'origine, et montré que celle-ci est toujours compétitive vis-à-vis de l'état de l'art actuel.

**Classification de patches 2D par apprentissage** Si les variantes précédemment évoquées exploitent des descripteurs conçus manuellement, d'autres approches préfèrent optimiser ceux-ci aux objets considérés par apprentissage automatique. Ces approches ont typiquement été développées pour le traitement de données RGB et RGBD, et cherchent à prédire pour un patch local d'une image, si ce dernier appartient à une instance d'objet, et si oui quelle est sa pose. En pratique, cette prédiction prend la forme d'un ensemble de votes pour des hypothèses de poses, qui correspondent typiquement à des exemples rencontrés lors de l'apprentissage.

[Gall et al. \(2011\)](#) détectent et prédisent ainsi la pose 2D de piétons dans une image, en réalisant leurs prédictions au moyen d'une forêt de décision exploitant des différences de valeurs entre des paires de pixels observées dans un patch. Cette méthode est reprise par [Girshick et al. \(2011\)](#) afin de prédire la position de points clés du squelette humain dans des images de profondeur, puis par [Tejani et al. \(2014\)](#) pour prédire la pose d'un objet rigide à partir de données RGBD. [Doumanoglou et al. \(2016\)](#) propose une variation de cette dernière utilisant une forêt de décision exploitant les descripteurs appris par un réseau de neurones autoencodeur, tandis que [Kehl et al. \(2016\)](#) recourent directement à de tels descripteurs pour générer des votes, par recherche de plus proches voisins parmi des exemples d'apprentissage<sup>14</sup>. Enfin, [Rodrigues et al. \(2012\)](#) estiment quant à eux la pose d'un objet rigide

14. À l'instar des approches à base de descripteurs globaux évoquées section 2.4.2.2.

au moyen d'une forêt de fougères de décision<sup>15</sup>, travaillant à partir de patches RGB d'une scène éclairée de sorte à mettre en évidence les normales des surfaces.

Ces hypothèses *faibles* sont ensuite agrégées afin de produire des hypothèses plus *fortes*, correspondant aux maxima de densité de votes sur l'espace des poses admissibles. Cet espace est cependant généralement considéré comme de dimension trop importante pour pouvoir être discrétisé avec une résolution suffisante afin de mettre en place une détection au moyen d'un accumulateur de type Hough. Aussi, de même que Drost et al. (2010), la détections de maxima de densité dans la distribution des votes est typiquement réalisée par étapes. Rodrigues et al. (2012) commencent ainsi par détecter les maxima de probabilité d'une position 2D dans l'image au moyen d'un accumulateur de Hough, puis estiment la profondeur et l'orientation de l'objet en les pics 2D détectés au moyen d'un second accumulateur de Hough, tandis que Tejani et al. (2014); Doumanoglou et al. (2016); Kehl et al. (2016) réalisent cette seconde étape d'estimation d'orientation par Mean-Shift.

## 2.6 Raffinement et vérification d'hypothèses

L'ensemble des méthodes évoquées dans cet état de l'art produisent des hypothèses concernant la pose d'instances détectées dans les données. Ces hypothèses sont estimées de manière plus ou moins précises, et leurs poses doivent donc potentiellement être raffinées de manière à coller au mieux aux données, ainsi qu'être évaluées afin d'être infirmées ou confirmées.

**Raffinement d'hypothèse de pose** Le raffinement d'une hypothèse de pose peut être formulé comme un problème d'optimisation. L'approche usuelle pour se faire consiste à produire des données synthétiques de l'objet dans la pose supposée, et à ajuster cette pose de manière à faire correspondre autant que faire ce peut les données synthétiques aux données observées.

Dans le cas de données 3D, cette optimisation est classiquement réalisée au moyen de la méthode *Iterative Closest Point* de Besl et McKay (1992), conçue pour recalibrer un nuage de points  $\{\mathbf{x}_i\}_{i=1,\dots,n}$  – représentant par exemple les points d'un modèle de l'objet – à un autre nuage de points  $\{\mathbf{y}_i\}_{i=1,\dots,m}$  – représentant par exemple les données, avec  $n, m \in \mathbb{N}^*$  – au moyen d'une transformation rigide  $T \in SE(3)$ , de sorte à minimiser la distance entre ceux-ci

$$d(T) = \sum_{i=1}^n \left( \operatorname{argmin}_{j \in \llbracket 1, m \rrbracket} \|T(\mathbf{x}_i) - \mathbf{y}_j\|^2 \right). \quad (2.3)$$

Il s'agit d'une méthode itérative simple pour laquelle de nombreuses variantes existent, notamment celles de Granger et Pennec (2002); Jost et Hügli (2003) visant à accélérer la convergence ainsi qu'à limiter la présence de

15. *random ferns* dans la littérature en langue anglaise. Sorte de collections de tables de hachage apprises à la manière des arbres de décision et présentant un coût mémoire et un pouvoir discriminant moindre que ces derniers.

minima locaux, ou encore l'approche de [Yang et al. \(2013\)](#) garantissant de trouver le minimum global au prix d'un temps de calcul important. D'autres techniques d'optimisations permettent également de mettre en œuvre un tel recalage, et nous citerons notamment la méthode de [Fitzgibbon \(2003\)](#) basée sur un solveur de type Levenberg-Marcquardt et l'utilisation d'une grille de voxels, employée dans LINEMOD ([Hinterstoisser et al., 2012b](#)), ainsi que des techniques plus exotiques comme l'optimisation par essaim particulière, employée par [Hodaň et al. \(2015\)](#).

Le recalage peut également être réalisé de manière à aligner les contours de l'objet en sa pose supposée avec des contours perçus dans les données lorsque celles-ci sont sous forme d'image. Une formulation similaire à la précédente peut alors être utilisée pour le recalage 2D-3D, en exploitant une paramétrisation locale telle que celle formalisée par [Lowe \(1987\)](#); [Araújo et al. \(1998\)](#) dans le cas de l'utilisation d'un solveur du premier ordre.

Il convient dans tous les cas d'utiliser une méthode de recalage robuste aux valeurs aberrantes, car il y a nécessairement des zones du modèle non visibles dans les données du fait d'occultations ainsi que des zones des données n'appartenant pas à l'instance d'objet à recaler<sup>16</sup>. Cette robustesse peut être partiellement obtenue en écartant les résidus composés de paires de points trop distants dans le problème d'optimisation (2.3), ou encore en utilisant une formulation moins sensible aux valeurs aberrantes, à l'instar de la pondération Geman-McClure des résidus<sup>17</sup> utilisée par [Kehl et al. \(2017\)](#).

En lieu et place d'une formulation explicite de l'optimisation à mettre en œuvre, [Oberweger et al. \(2015\)](#) puis [Rad et Lepetit \(2017\)](#) ont récemment proposé de réaliser ce raffinement de pose au moyen d'un réseau de neurones convolutif entraîné pour prédire une modification de pose réduisant l'écart entre la pose supposée et celle observée dans les données. Une telle approche présente l'intérêt d'être potentiellement plus à même de gérer les problématiques de fond et d'occultation, pourvu d'un apprentissage adéquat.

**Vérification d'hypothèses** Afin de limiter au maximum la production de fausses détections, la majorité des approches de l'état de l'art procèdent à un filtrage des hypothèses de pose produites. Ce filtrage est souvent mis en place à partir de critères heuristiques tels que le pourcentage de caractéristiques de l'objet que l'on retrouve de manière cohérente dans les données étant donné sa pose supposée, pouvant s'agir de points 3D, de normales de la surface, de contours, de couleur, etc. ([Bolles et Horaud, 1986](#); [Park et al., 2010](#); [Drost et Ilic, 2012](#); [Hinterstoisser et al., 2012b](#); [Rodrigues et al., 2012](#); [Birdal et Ilic, 2015](#)). [Krull et al. \(2015\)](#) utilisent quant à eux un réseau de neurones convolutif afin d'estimer la vraisemblance d'une hypothèse de pose à partir d'un rendu synthétique de l'objet en la pose supposée, des données d'entrées et de la sortie d'un étage précédent de leur méthode.

16. En l'absence de segmentation préalable de ce dernier.

17. Le problème de minimisation de distance équation (2.3) défini au sens des moindres carré sous une forme  $\sum_i r_i^2$  peut être généralisé sous une forme  $\sum_i \phi(r_i)$ , avec  $\phi$  une fonction de pondération moins sensible aux valeurs aberrantes que la fonction carré. La pondération de Geman-McClure  $t \mapsto \mu t^2 / (\mu + t^2)$  de paramètre  $\mu > 0$  est une de ces fonctions.

La vérification des hypothèses de pose indépendamment les unes des autres n'est cependant pas la seule approche possible lorsqu'il s'agit de proposer une interprétation d'une scène où différentes instances d'objets sont présentes. [Barinova et al. \(2012\)](#) proposent ainsi un formalisme intéressant où la détection d'un nombre inconnu d'instances d'objet est traitée comme un problème d'optimisation probabiliste de type *Maximum A Posteriori*, qu'ils appliquent à la détection 2D de piétons dans une image. La validité de ce formalisme semble cependant fondamentalement liée à la possibilité de discrétiser l'espace, qui de plus ne peut être résolu efficacement que de manière gloutonne. [Aldoma et al. \(2016\)](#) proposent dans le même ordre d'idée un formalisme de vérification global, dont le but est de sélectionner parmi un ensemble d'hypothèses de pose d'objets un sous-ensemble cohérent qui constitue la « meilleure » interprétation globale de la scène.

## 2.7 Gestion des symétries

Les méthodes présentées dans cet état de l'art ont une vocation générale, c'est à dire à être applicable à des objets arbitraires. Une hypothèse implicite souvent réalisée réside cependant dans la paramétrisation d'une pose par une transformation rigide, qui entraîne certaines difficultés dans le cas d'objets symétriques.

Ces difficultés peuvent être contournées relativement facilement dans le cadre d'une approche procédant par classification de pose ou recherche de patrons, puisqu'il suffit alors de limiter le nombre de classes à considérer à un ensemble de poses véritablement distinctes.

La présence de symétries propres – c.-à-d. d'invariances par rotation – pour un objet entraîne cependant d'importantes difficultés pour l'apprentissage de méthodes de régression de pose. En effet, une même pose peut alors être représentée par plusieurs paramétrisations différentes. Les techniques d'apprentissage usuelles de régression vont dès lors tendre à prédire une paramétrisation moyenne qui minimise la distance aux paramétrisations possibles de la pose, malgré le fait qu'une telle sortie puisse ne pas correspondre du tout à la pose attendue. [Rad et Lepetit \(2017\)](#) présentent dans leur récente publication une technique permettant de contourner cette difficulté pour le cas précis de certaines symétries rencontrées dans leurs expérimentations<sup>18</sup>, mais il s'agit selon nous d'avantage d'un correctif *ad hoc* que d'une approche à généraliser.

Les approches locales souffrent également de ces questions de symétries si elles ne sont pas prises en compte. La présence de symétrie rend en effet l'appariement d'un point d'intérêt dans les données avec un unique point du modèle fondamentalement impossible, tandis que l'agrégation d'hypothèses de pose est rendue suboptimale lorsque ces hypothèses se retrouvent dispersées entre différentes paramétrisations. Ces questions sont cependant peu discutées dans la littérature et certains auteurs comme [Hinterstoisser et al. \(2016\)](#) choisissent de ne pas publier les résultats de leurs méthodes dans le cas d'objets présentant une symétrie de révolution.

18. Objets présentant une symétrie de révolution autour d'un axe d'un angle donné couplée à une symétrie plane passant par cet axe.

## 2.8 Synthèse et pistes de recherche

Ainsi que les sections précédentes l'ont montré, le problème de la détection et de l'estimation de pose a été abordé de manière relativement diverse dans la littérature scientifique au cours des 30 dernières années. Nous avons tâché de présenter un panorama cohérent de ces différentes approches, bâti autour de la distinction entre approches globales et locales. Bien que la classification proposée soit imparfaite, nous formulons néanmoins quelques enseignements généraux de cet état de l'art qui font l'objet de cette synthèse, et à partir desquels nous établissons les pistes de recherche suivies lors de cette thèse.

**Approche globale, approche locale** Les approches globales évaluent l'intégralité des données représentant une instance d'objet afin de réaliser leurs inférences. Elles disposent donc d'un maximum d'information pour réaliser leurs prédictions, cependant cette disponibilité les rend également relativement sensibles au contexte, qu'il s'agisse du fond entourant les instances d'objet ou encore des problèmes d'occultation partielles de ces dernières. La tendance actuelle est néanmoins d'orienter les recherches vers de telles approches globales, rendues robustes à ces changements de contexte par le biais d'un apprentissage adéquat typiquement mises en œuvre au moyen de réseaux de neurones convolutifs et d'importantes quantités de données d'apprentissage en contexte.

A contrario, les approches locales se basent sur l'identification de zones limitées des données afin de réaliser leurs inférences. Celles-ci présentent donc fondamentalement un pouvoir discriminant plus faible que les approches globales, qu'elles compensent par le biais de la fusion de ces différentes informations locales et l'utilisation de méthodes de génération d'hypothèses de pose robustes aux valeurs aberrantes. Ces techniques confèrent aux approches locales une certaine robustesse intrinsèque aux occultations, étant en mesure de prédire la pose d'instances d'objet à partir de l'identification de quelques zones limitées de l'objet uniquement.

**Facilité de mise en œuvre** Comme dans le reste des domaines de la vision par ordinateur, on observe parmi les techniques de détection et d'estimation de pose une tendance vers l'usage de plus en plus massif de techniques d'apprentissage automatique, dont l'apprentissage profond à l'aide de réseaux de neurones convolutifs constitue l'approche actuellement privilégiée dans la littérature.

Ces techniques, de par leur capacité à optimiser automatiquement leurs performances qualitatives pour un problème donné à partir d'exemples, présentent un intérêt évident. Celles-ci déplacent cependant la difficulté sur la question des données nécessaires à l'apprentissage, et par la même sur la facilité de mise en œuvre de ces approches. En effet, on observe globalement que les méthodes reposant sur un usage avancé d'apprentissage automatique requièrent généralement des données d'apprentissages plus délicates à obtenir que celles utilisées par des techniques plus basiques, comme l'illustre la figure 2.19. Ainsi, si des méthodes basées sur l'usage de descripteurs manuels comme celle de [Drost et al. \(2010\)](#) peuvent être

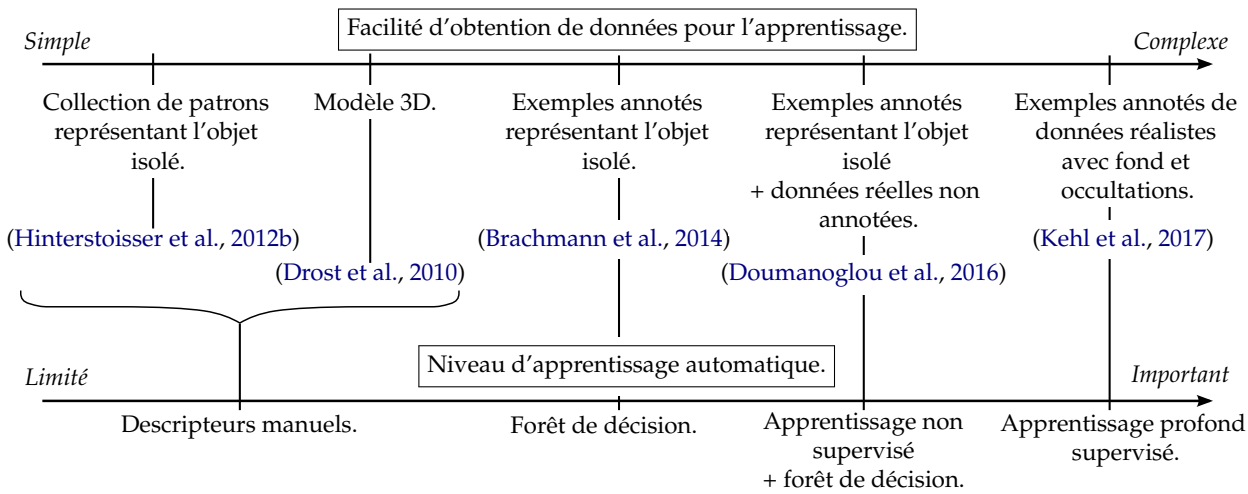


FIGURE 2.19 – Lien entre le niveau d'apprentissage automatique utilisé et l'utilisation de données d'apprentissage délicates à obtenir. **En haut** : Types de données exploitées pour entraîner une méthode de détection et d'estimation de pose, ordonnées par difficulté d'obtention de ces dernières. **Au centre** : exemples de travaux exploitant ces techniques. **En bas** : Techniques d'apprentissage automatique correspondantes.

mis en place relativement aisément à partir d'un modèle 3D de l'objet, les récentes approches de Kehl et al. (2017) et Rad et Lepetit (2017) basées sur un apprentissage profond nécessitent quant à elles d'importantes quantités de données d'apprentissage annotées et réalistes d'instances d'objet dans leur environnement. Ces données ne sont pas triviales à obtenir, et la mise en œuvre d'une telle approche pour un nouvel objet s'avère dès lors plus complexe que dans le cas d'une méthode simple.

Les quelques travaux explorant des techniques d'apprentissage partiellement non supervisées comme ceux de Doumanoglou et al. (2016) ou cherchant à apprendre des représentations intermédiaires à l'instar de ceux de Wohlhart et Lepetit (2015) permettent néanmoins de nuancer cette position, en laissant entrevoir la possibilité d'apprendre à généraliser facilement entre différents objets.

**Évaluation** Si certaines pistes de recherches telles que celles basées sur la détection de primitives comme des cercles ou des cylindres ont globalement été délaissées de par la difficulté de généraliser celles-ci à tout type d'objet, l'état de l'art présenté ici n'admet cependant pas de vainqueur clair qui supplanterait significativement l'ensemble des autres approches pour la majorité des applications.

Les approches basées sur une recherche de patrons dans une image ont en effet beau présenter un espace de recherche important<sup>19</sup>, Hinterstoisser et al. (2012b) ont prouvé sur leur jeu de données qu'il était possible d'effectuer cette recherche et localiser une instance d'objet en force brute en moins de

19. Typiquement constitué de la position, l'échelle et l'orientation du patron.

200ms. Ces approches présentent de plus l'avantage de nécessiter peu de données d'apprentissage, puisqu'un exemple par point de vue de l'objet est typiquement suffisant afin de définir un patron.

A contrario, si les approches exploitant des techniques d'apprentissage automatiques avancées telles que des arbres de décision ou de l'apprentissage profond affichent de meilleures performances qualitatives sur certains jeux de données, celles-ci nécessitent quant à elles des quantités de données d'apprentissage beaucoup plus importantes, ainsi qu'une phase d'apprentissage pouvant être relativement longue.

La récente publication de [Hinterstoisser et al. \(2016\)](#) remet même en question cette supériorité, en montrant qu'une méthode de 2010 basée sur des descripteurs géométriques simples pouvait obtenir des performances rivalisant avec l'état de l'art de l'apprentissage automatique une fois mis en œuvre un certain nombre d'heuristiques bien choisies. Cette dernière méthode présente pourtant des limitations fondamentales en pratique, en n'étant notamment pas adaptée au cas des objets plats.

Nous voyons dans ce statu quo apparent le symptôme d'une évaluation inadaptée de l'état de l'art, basée notamment sur un jeu de données de référence LINEMOD ([Hinterstoisser et al., 2012b](#)) peu représentatif de la variabilité des cas d'utilisation réels, et faisant l'objet d'un certain surapprentissage.

## Pistes de recherche

Cette thèse, de par son aspect industriel, présente un objectif applicatif certain. Il s'agit en effet d'aboutir à une méthode de détection et d'estimation de pose d'instances d'objet rigide en vrac qui soit à la fois :

- Efficiente, en ce que l'on souhaite être en mesure de détecter et estimer la pose de plusieurs instances d'objet dans des scènes de vrac de manière suffisamment précise pour permettre une manipulation robotisée, en une durée limitée (de l'ordre de la seconde, sur du matériel courant).
- Générique, en ce que qu'elle soit applicable à la majorité des objets rigides rencontrés dans une application de dévracage industriel.
- Simple de mise en place.

Au regard de l'état de l'art, l'objectif de précision des estimations de pose nous incite à exploiter des données 3D ou des acquisitions selon plusieurs points de vues. Nous privilégions l'usage d'une modalité de données 3D – que l'invariance aux changements d'illumination rend d'usage plus simple –, et choisissons d'explorer une approche de type *régression probabiliste* permettant une fusion simple de prédictions faites suivant différents points de vue si besoin. Plus précisément nous choisissons de travailler à partir d'*images de profondeur*, car ces dernières préservent d'avantage d'information qu'un nuage de points, notamment vis à vis de la notion de surface et d'occupation de l'espace.

Par mesure de simplicité, nous préférons de plus que l'entraînement à la reconnaissance d'un objet soit indépendant de l'environnement dans lequel les instances de ce dernier devront être localisées. Nous privilégions



pour cela une *approche locale* – qui présente intrinsèquement une certaine robustesse aux occultations et au fouillis – et que l'on entraîne à partir d'un modèle 3D de l'objet, facilement disponible dans le cadre d'applications industrielles. Nous explorons la mise en œuvre de celle-ci au moyen d'une *forêt de décision*.

Au moment de nos choix d'orientations en 2014, cette technique avait montré un certain succès avec des problématiques similaires à la nôtre comme l'estimation de pose humaine (Shotton et al., 2013a). L'usage de réseaux de neurones convolutifs pour la détection et la reconnaissance de multiples objets dans une image était alors encore à ses balbutiements avec notamment le développement de *R-CNN* par Girshick et al. (2014), dont la durée d'exécution était trop importante pour notre cas d'application<sup>20</sup>. Le développement d'algorithmes plus efficaces tels que *Faster R-CNN* (Ren et al., 2015), *YOLO* (Redmon et al., 2016) ou *SSD* (Liu et al., 2016), associé aux progrès matériels a depuis changé la donne, et fait de l'apprentissage profond la piste privilégiée dans la majorité des publications scientifiques récentes. Notre approche conserve néanmoins sa pertinence dans le cadre d'une application industrielle. La production de données annotées réalistes et représentatives du domaine d'utilisation afin d'entraîner un réseau de neurones n'est en effet pas triviale, et l'usage d'une approche locale associée à une forêt de décision permet d'introduire certains apriori qui simplifient cet apprentissage, et ainsi sa mise en œuvre.

**Contributions majeures** Si nous abordons certains des aspects précédents de manière novatrice, il s'agit principalement d'innovations incrémentales, pouvant être considérées comme relevant de l'ingénierie.

Dans cette thèse, nous choisissons également d'explorer deux domaines ayant été relativement laissés de côté dans la littérature, et nous paraissant d'importance sur le plan scientifique :

- La question des objets présentant des symétries a été relativement laissée de côté dans la littérature ainsi qu'on l'a vu, cependant de nombreux types d'objets manufacturés présentent de telles propriétés. Nous explorons donc les notions de symétrie et de pose d'objet dans le détail – en y consacrant l'entièreté du chapitre 3 –, afin d'être en mesure de gérer convenablement les objets présentant des symétries.
- Les évaluations réalisées dans l'état de l'art ne nous semblent de plus que peu satisfaisantes afin de juger des performances d'une méthode de détection et d'estimation de pose en conditions réelles, or il s'agit d'un point essentiel au progrès du domaine. Nous tâchons donc d'améliorer ce point en proposant un protocole expérimental plus adapté, qui fait l'objet du chapitre 5.

---

20. 13s sur GPU, 53s sur CPU pour la détection et la reconnaissance d'objets sur le jeu PASCAL VOC par *R-CNN* en 2014.

## Chapitre 3

# Notion de pose d'objet, et distance associée

*Computer science is not about machines in the same way that astronomy is not about telescopes. There is an essential unity of mathematics and computer science.*

– Michael R. Fellows, *Computer science and mathematics in the elementary schools.*

---

3.1	Introduction . . . . .	56
3.2	Définition de la pose . . . . .	58
3.3	État de l'art relatif aux distances sur l'espace de pose . . . . .	60
3.4	Distance proposée . . . . .	64
3.5	Calculs efficaces de distance . . . . .	68
3.6	Symétries des représentants . . . . .	77
3.7	Projection sur l'espace de pose . . . . .	80
3.8	Moyenne de poses . . . . .	82
3.9	Propriétés locales de la distance proposée . . . . .	90
3.10	Exemple applicatif . . . . .	92
3.11	Synthèse . . . . .	98

---

L'estimation de pose d'un objet rigide est basée comme son nom l'indique sur la notion de pose d'objet. La pose d'un objet rigide est habituellement considérée comme une transformation rigide, décrite par six degrés de liberté : trois degrés de translation, trois de rotation. Cependant, assimiler l'espace de pose à l'espace des transformations rigides est en règle général abusif et ne permet pas de prendre en compte convenablement les objets présentant des symétries propres. Or ceux-ci sont nombreux parmi les objets fabriqués par l'homme.

Dans ce chapitre nous proposons de dépasser cette limitation en définissant la pose d'un objet comme un état statique distinguable de ce dernier, et montrons qu'une pose peut alors être assimilée à un ensemble de transformations rigides. En nous appuyant uniquement sur des considérations géométriques, nous proposons une distance objective sur l'espace des poses, valable pour n'importe quel objet rigide physique et ne requérant pas de

paramétrage arbitraire. Nous montrons comment cette distance peut être évaluée de manière efficace à l'aide d'une représentation des poses dans un espace euclidien de dimension au plus 12 suivant les symétries de l'objet considéré. Cette propriété permet de réaliser de manière efficace des requêtes de voisinage parmi un grand nombre de poses, telles que la recherche de poses dans une boule de rayon donné autour d'une pose de référence, ou la recherche de ses  $k$  plus proches voisins, à l'aide d'outils sur étagère. Elle rend également possible de calculer facilement la moyenne de poses au sens de cette métrique, à l'aide d'une fonction de projection de l'espace euclidien sur l'espace de pose.

L'intérêt pratique de ces développements théoriques est illustré sur un exemple d'estimation de pose d'instances d'objet rigide 3D à partir d'un nuage de points, à l'aide d'une procédure de Mean Shift. Ceux-ci sont également à la base de l'approche que nous développons au chapitre 4.

Ce chapitre consiste principalement en une adaptation d'un article publié dans le journal IJCV (Brégier et al., 2017b).

### 3.1 Introduction

Le modèle d'objet rigide joue un rôle important dans de nombreux domaines scientifiques et techniques, tels que les sciences physiques, la mécanique, la vision par ordinateur ou encore l'animation 3D. Sous hypothèse de rigidité, l'état statique d'un objet est qualifié de *pose*, et est souvent décrit en termes de *position* et d'*orientation*.

Les poses d'un objet 3D rigide sont traditionnellement assimilées à des transformations rigides, et l'ensemble des poses d'un objet identifié au groupe des transformations rigides  $SE(3)$ , le groupe spécial euclidien. La structure de groupe de Lie de  $SE(3)$  permet de mettre en évidence le déplacement relatif de l'objet entre deux poses, et de définir une distance entre celles-ci comme la longueur du plus petit mouvement réalisant ce déplacement. Cette identification est en cela particulièrement adaptée aux applications où le mouvement d'un corps rigide est considéré, telles que la planification de trajectoire (Sucan et al., 2012) ou le suivi d'objet (Tjaden et al., 2016). Malgré les nombreux travaux existants relatifs aux métriques de  $SE(3)$ , il reste cependant toujours difficile de déterminer la *bonne* manière de traiter les poses d'un objet. La question de « comment régler l'importance de l'orientation relativement à la position » demeure notamment, même à l'ère de l'apprentissage profond (Kendall et al., 2015).

Il existe de plus des applications où les considérations de mouvement ne sont pas pertinentes, et pour lesquelles seule une notion de *similarité* entre poses est nécessaire. L'estimation de pose d'instances d'un objet rigide à partir d'un ensemble de votes bruités en est un bon exemple. Alors que les applications basées sur la notion de mouvement s'appuient sur les propriétés locales de l'espace de pose, qui ont fait l'objet de recherches conséquentes, les applications basées sur la notion de similarité présentent des besoins différents et ont typiquement à gérer de nombreuses poses simultanément, afin de réaliser des opérations telles que des requêtes de voisinage, consistant à identifier des poses similaires à une pose donnée au sein d'un ensemble, ou encore à moyenniser de poses, et n'ont pas fait l'objet d'un

tel intérêt. Par conséquent, des mesures de similarité souffrant de défauts majeurs continuent à être utilisées. À titre d'exemple, [Tejani et al. \(2014\)](#); [Doumanoglou et al. \(2016\)](#) utilisent, suivant l'approche de [Fanelli et al. \(2011\)](#), une procédure de Mean Shift basée sur la distance euclidienne entre paramétrisations de poses au moyen d'angles d'Euler dans leur récente méthode de détection et d'estimation de pose. Bien qu'une telle mesure puisse être estimée de manière rapide et permette l'usage d'outils efficaces de recherche de voisinage développés pour les espaces euclidiens, il ne s'agit pas d'une distance. En effet, la paramétrisation d'une rotation sous forme d'angles d'Euler souffre notablement de problèmes tels que des effets de bords et des singularités, et est de plus dépendante du choix de repère. Il est possible que ces défauts n'aient que des effets limités sur les résultats expérimentaux présentés par ces auteurs, grâce à un choix approprié de système de coordonnées et à la variabilité limitée des orientations des objets au sein de leurs jeux de données. Cependant, ces problèmes ne sauraient être évités dans le cas général d'objets ayant des orientations arbitraires. Un tel exemple illustre le manque d'outils adaptés à la gestion efficace de larges ensembles de poses.

Il existe enfin des objets rigides dont les poses ne peuvent pas être identifiées à une unique transformation rigide et dès lors pour lesquels les résultats existants ne peuvent s'appliquer. Cette situation se rencontre chez des objets présentant certaines propriétés de symétrie ; tels que des objets de révolution ou des pavés ; et est relativement courante parmi les objets produits par l'homme. La littérature existante en estimation de pose d'objets ne développe généralement pas comment traiter de tels objets, et la méthode de validation la plus répandue pour ceux-ci ([Hinterstoisser et al., 2012b](#)) consiste en une mesure de similarité qui ne permet pas de distinguer des cas tels qu'une boîte cylindrique (p. ex. une boîte de conserve) à l'endroit ou à l'envers.

**Organisation du chapitre** Nous tâchons dans ce chapitre de dépasser ces difficultés en développant un cadre général permettant de traiter, dans des applications pratiques, le cas de n'importe quel objet rigide. Pour se faire, nous proposons une définition de la notion de pose valide pour tout objet rigide physiquement admissible, selon laquelle une pose peut être identifiée avec un ensemble de transformations rigides (section 3.2). Nous proposons alors une distance physiquement pertinente sur l'espace des poses (section 3.4), et montrons comment les poses peuvent être représentées dans un espace euclidien de manière à permettre des calculs de distance et des requêtes de voisinage efficaces (section 3.5). Nous exposons comment le problème de l'estimation de la moyenne de poses peut également être résolu de manière efficace (section 3.8) pour cette distance, à l'aide d'une méthode de projection présentée section 3.7. Enfin nous proposons section 3.10 un exemple d'application concernant l'estimation de poses d'instances d'un objet rigide à partir d'un ensemble de votes, et synthétisons les principaux résultats de ce chapitre section 3.11.

## 3.2 Définition de la pose

Bien que la notion de pose soit largement répandue, p. ex. en robotique ou en vision par ordinateur, nous n'avons pas trouvé dans la littérature une définition générale de celle-ci. Aussi, nous proposons la suivante :

**Définition 1.** Une pose d'un objet rigide est un état statique distinguable de cet objet.

Nous désignons par *espace de pose* l'ensemble des poses possibles, que nous notons  $\mathcal{C}$  pour des raisons de cohérence avec la notion d'*espace de configuration* parfois utilisée dans la littérature robotique.

### 3.2.1 Lien entre l'espace de pose et $SE(3)$

La notion d'espace de pose est étroitement liée au groupe des transformations rigides  $SE(3)$ . Considérons en effet un objet rigide, et  $\mathcal{P}_0 \in \mathcal{C}$  un pose arbitraire de référence de ce dernier.

Une transformation rigide appliquée à l'objet en sa pose de référence définit un état statique de l'objet, c.-à-d. une pose. Réciproquement, une pose  $\mathcal{P} \in \mathcal{C}$  de l'objet peut être atteinte au moyen d'un déplacement rigide à partir de la pose de référence  $\mathcal{P}_0$ , aussi  $\mathcal{P}$  peut-elle être complètement définie par la transformation rigide correspondant à ce déplacement.

Nous notons  $\mathcal{P} \in \mathcal{C}$  et  $T = (R, t) \in SE(3)$  un tel couple de pose et de transformation rigide – avec  $R \in SO(3)$  une matrice de rotation et  $t \in \mathbb{R}^3$  un vecteur de translation. La transformation considérée ici est telle que tout point  $x \in \mathbb{R}^3$  lié à une instance d'objet dans la pose de référence  $\mathcal{P}_0$  est transformé par  $T$  en le point correspondant  $T(x)$  d'une instance en la pose  $\mathcal{P}$  comme suit, et ainsi qu'illustré figure 3.3 :

$$T(x) = Rx + t. \quad (3.1)$$

La transformation rigide correspondant à une pose donnée n'est néanmoins pas nécessairement unique et dès lors l'identification de  $SE(3)$  et de l'espace de pose est dans le cas général incorrecte. Certains objets – et en particulier ceux produits industriellement – peuvent en effet présenter des propriétés de symétrie propre les rendant invariant à certaines transformations rigides.

### 3.2.2 Pose comme classe d'équivalence de $SE(3)$

Soit  $M \subset SE(3)$  l'ensemble des transformations rigides représentant la même pose qu'une transformation rigide  $T$ . Dans le cas de l'objet *lapin* figure 3.2d,  $M$  consiste typiquement en le singleton  $\{T\}$ . Mais  $M$  peut également contenir un continuum de poses dans le cas d'un objet de révolution tel que le chandelier figure 3.2a, ou même un ensemble discret, tel que pour la fusée illustrée figure 3.2e pour laquelle une même pose peut être représentée par trois transformations.

Par définition de  $M$ ,  $G \triangleq \{T^{-1} \circ M, M \in M\}$  représente l'ensemble des transformations rigides n'ayant pas d'effet sur l'état statique de l'objet. Cet ensemble ne dépend donc pas de la transformation arbitraire  $T$  considérée.

$G$  est de plus un sous-groupe de  $SE(3)$ . En effet, des compositions et inversions de telles transformations peuvent être appliquées à l'objet tout en laissant ce dernier inchangé, et la transformation identité n'a évidemment pas d'effet sur la pose de l'objet. Nous désignons par *symétries propres* de l'objet les éléments de ce groupe, et désignons  $G \subset SE(3)$  comme le groupe des symétries propres de l'objet.

Étant donné une transformation rigide  $T$  définissant une pose  $\mathcal{P}$ , nous pouvons donc identifier  $\mathcal{P}$  à la classe d'équivalence  $[T] \subset SE(3)$  suivante, consistant en la composition de  $T$  avec n'importe quelle transformation rigide n'ayant pas d'effet sur la pose de l'objet :

$$\mathcal{P} = [T] \triangleq \{T \circ G, G \in G\}. \quad (3.2)$$

### 3.2.3 Le groupe des symétries propres

Nous proposons dans cette section une classification des groupes de symétries propres admissibles pour un objet rigide borné. Bien que des modèles d'objets infinis soient parfois utilisés, par exemple lors de la détection de plan en analyse de scène 3D, nous ne considérons pas ceux-ci dans le cadre de nos travaux en ce qu'ils ne correspondent pas à des objets ayant une réalité physique, et que la définition d'une métrique sur l'espace de pose d'un tel objet est typiquement dépendante de l'application considérée. Les résultats pratiques développés plus loin dans ce chapitre découlent de cette classification.

L'ensemble des symétries propres d'un objet borné partagent nécessairement un point fixe commun, aussi il est possible de considérer le groupe des symétries propres comme un sous-groupe du groupe des rotations  $SO(3)$  en choisissant un tel point comme origine du repère objet. Les sous-groupes de  $SO(3)$  ont notamment été étudiés dans le contexte de la cristallographie, et le lecteur intéressé est renvoyé à l'ouvrage de [Vainsthein \(1994\)](#) pour de plus amples informations relatives à la théorie de la symétrie.

En ignorant le cas pathologique des sous-groupes infinis de  $SO(3)$  non fermés suivant la topologie usuelle – en ce que ces derniers ne font pas sens physiquement –, les groupes de symétrie propre admissibles par un objet borné peuvent être classifiés en quelques catégories.

Dans le cas 2D, un objet borné présente soit une symétrie circulaire – c'est à dire une invariance par n'importe quelle rotation 2D – soit une symétrie cyclique d'ordre  $n \in \mathbb{N}^*$  – c.-à-d. une invariance par rotation de  $1/n$  tours, le cas particulier  $n = 1$  correspondant à un objet sans symétrie propre (hormis l'identité). Le tableau 3.1 présente des exemples de ces symétries.

De manière similaire, nous distinguons dans le cas 3D cinq classes de groupes de symétrie propre, synthétisées dans le tableau 3.2. Un objet 3D borné peut en effet présenter une symétrie sphérique – c.-à-d. une invariance par n'importe quelle rotation – ou une symétrie de révolution – c.-à-d. une invariance par rotation d'angle arbitraire autour d'un axe donné. Cette dernière classe peut être scindée en deux, suivant que l'objet soit ou non également invariant par réflexion suivant un plan orthogonal à l'axe de révolution. Nous désignons respectivement ces classes par symétrie de révolution avec ou sans invariance par rotoréflexion. Il convient également de

TABLE 3.1 – Classification des groupes de symétrie propre admissibles par un objet physique 2D borné.









		
Sans symétrie propre	Symétrie cyclique (non triviale)	Symétrie circulaire

TABLE 3.2 – Classification des groupes de symétrie propre admissibles par un objet physique 3D.

Groupes infinis		Groupes finis		
Symétrie de révolution				
				
(a) Sans invariance par rotoréflexion	(b) Avec invariance par rotoréflexion	(c) Symétrie sphérique	(d) Sans symétrie propre	(e) Fini non trivial

considérer les groupes finis de symétrie propre. Il en existe une infinité, aussi ceux-ci sont considérés de manière générale. Nous distinguons néanmoins le cas d'un objet sans symétrie propre (c.-à-d. pour lequel  $G$  consiste en le singleton identité) des autres, car ce dernier est essentiel à nos développements. Notons que les symétries indirectes potentielles de l'objet ne sont pas prise en compte ici. Cela est du au fait que l'on considère un espace orienté – par exemple par la règle de la main droite – dans lequel les réflexions ne sont pas réalisables au moyen de déplacements rigides. La symétrie de révolution avec invariance par rotoréflexion est néanmoins considérée, car il s'agit d'un groupe de symétrie propre : en effet, la symétrie de réflexion peut être obtenue par l'introduction d'une invariance par rotation de  $180^\circ$  suivant un axe arbitraire, orthogonal à l'axe de révolution.

### 3.3 État de l'art relatif aux distances sur l'espace de pose

Nous proposons dans cette section un état de l'art succinct des travaux récents concernant la définition d'une distance sur l'espace de pose d'un objet rigide. Nous ne considérons ici que des distances au sens mathématique du terme, c'est à dire des applications symétriques définies positives de  $\mathcal{C} \times \mathcal{C}$  vers  $\mathbb{R}^+$  vérifiant l'inégalité triangulaire. À notre connaissance, la littérature existante ne prend pas en compte les symétries potentielles de l'objet, et dès lors dans cet état de l'art, l'espace de pose peut être identifié au groupe des transformations rigides  $SE(3)$ .

### 3.3.1 Objectivité

L'identification de l'espace de pose à  $SE(3)$  repose sur le choix de deux repères arbitraires : un repère lié à l'objet que l'on désigne par *repère objet*, et un repère fixe désigné comme *repère inertiel*, qui sont définis de telle manière que le repère objet coïncide avec le repère inertiel lorsque l'objet est dans sa pose de référence  $\mathcal{P}_0$ . Pour qu'une distance soit bien définie, elle ne doit pas dépendre du choix arbitraire de ces repères. Lin et Burdick (2000) formalisent cette notion sous le terme d'*objectivité* ou encore d'invariance en le choix de repère.

Parmi l'ensemble des distances envisageables sur  $SE(3)$ , les distances géodésiques focalisent certainement la majeure partie de l'intérêt et sont typiquement étudiées dans le cadre de la géométrie riemannienne sur le groupe de Lie  $SE(3)$ . Celles-ci sont en effet parfaitement adaptées pour les applications traitant du mouvement d'objet en ce qu'elles représentent la longueur minimale d'un mouvement permettant d'amener l'objet d'une pose à une autre. Park (1995) a montré qu'il n'existe pas de métrique riemannienne bi-invariante sur  $SE(3)$  – c'est à dire invariante par changement de repère inertiel (invariance à gauche) et changement de repère objet (invariance à droite). Chirikjian (2015) a récemment étudié cette question plus en avant et a montré qu'il existe néanmoins des métriques continues invariantes à gauche qui sont invariantes par transformation à droite par des rotations pures.

### 3.3.2 Approximation par des hyper-rotations

Néanmoins, plusieurs auteurs ont travaillé au développement de métriques « approximativement bi-invariantes » (Purwar et Ge, 2009) pour  $SE(3)$  au travers d'une application de l'espace des transformations rigides vers celui des hyper-rotations  $SO(4)$ , et l'usage d'une métrique bi-invariante sur cet espace. Différentes approches permettant de réaliser une telle application ont été proposées, basées notamment sur une représentation sous forme de biquaternion (Etzet et McCarthy, 1996) ou une décomposition polaire (Larochelle et al., 2007). Malheureusement, une telle transformation nécessite le choix d'un facteur d'échelle concernant la partie translation de la transformation rigide, et celle-ci doit être choisie empiriquement suivant l'application (Angeles, 2006).

### 3.3.3 Décomposition en position et orientation

Fort heureusement, bien que l'invariance en le choix du repère inertiel soit nécessaire pour l'*objectivité* d'une métrique, l'invariance en le choix du repère objet ne l'est pas. Lin et Burdick (2000) ont en effet montré qu'une métrique est objective si et seulement si celle-ci est indépendante du choix du repère inertiel et est modifiée par transformation à droite en réponse à un changement de repère objet. Aussi, une méthode simple pour définir une métrique objective consiste à définir une distance de manière invariante à gauche, à choisir un repère objet et à toujours considérer ce dernier, de manière à ne pas avoir à transformer l'expression de la distance.



Étant donné un choix de repère objet, une approche fréquente consiste à séparer une pose en une position et une orientation, de manière à définir une distance sur  $SE(3)$  basée sur des métriques invariantes aux changements de repères de l'espace des positions  $\mathbb{R}^3$  et du groupe des rotations  $SO(3)$ . Ces métriques peuvent alors être fusionnées sous forme d'une moyenne pondérée généralisée, ici écrite étant donné deux facteurs scalaires strictement positif  $a$  and  $b$  et un exposant  $p \in [1, \infty]$  :

$$d(\mathbf{T}_1, \mathbf{T}_2) = \sqrt[p]{a^p d_{\text{rot}}(\mathbf{R}_1, \mathbf{R}_2)^p + b^p d_{\text{trans}}(\mathbf{t}_1, \mathbf{t}_2)^p}. \quad (3.3)$$

La distance euclidienne est un choix usuel afin de quantifier la distance entre différentes positions. En combinant celle-ci avec la distance riemannienne usuelle sur  $SO(3)$ , il est possible d'obtenir une distance riemannienne sur  $SE(3)$  (Park, 1995) :

$$d(\mathbf{T}_1, \mathbf{T}_2) = \sqrt{a^2 \|\log(\mathbf{R}_1^{-1} \mathbf{R}_2)\|^2 + b^2 \|\mathbf{t}_2 - \mathbf{t}_1\|^2}. \quad (3.4)$$

Celle-ci est particulièrement intéressante en ce que la distance  $\|\log(\mathbf{R}_1^{-1} \mathbf{R}_2)\|$  entre deux orientations correspond à l'angle  $\alpha$  de la rotation relative entre celles-ci. Ce dernier peut être évalué simplement, par exemple à partir des relations suivantes s'appuyant sur des représentations matricielles  $\mathbf{R}_1, \mathbf{R}_2 \in SO(3)$  ou sous forme de quaternions unitaires  $\mathbf{q}_1, \mathbf{q}_2$  des orientations, et respectivement les opérateurs trace et produit scalaire :

$$\text{Tr}(\mathbf{R}_1^{-1} \mathbf{R}_2) = 2 \cos(\alpha) + 1 \quad (3.5)$$

$$1 - \langle \mathbf{q}_1 | \mathbf{q}_2 \rangle^2 = \frac{1}{2} (1 - \cos(\alpha)). \quad (3.6)$$

De nombreuses autres distances objectives peuvent être envisagées dès lors que l'on supprime la contrainte riemannienne. Gupta (1997) propose notamment d'utiliser la distance de Frobenius entre représentations matricielles des orientations. Celle-ci ne dépend également que de l'angle de la rotation relative entre les orientations, suivant la relation

$$\|\mathbf{R}_2 - \mathbf{R}_1\|_F = 2\sqrt{2} |\sin(\alpha/2)|. \quad (3.7)$$

La distance euclidienne entre des représentations d'orientation sous forme de paires antipodales de quaternions unitaires  $\mathbf{q}_1$  and  $\mathbf{q}_2$  présente des propriétés similaires est peut également être utilisée :

$$\min \|\mathbf{q}_2 \pm \mathbf{q}_1\| = 2 |\sin(\alpha/4)|. \quad (3.8)$$

Fusionner des distances de position et d'orientation en une distance sur  $SE(3)$  requière néanmoins de choisir les pondérations relatives de  $a$  et  $b$ . Le choix de ces dernières demeure un problème heuristique, et les travaux récents de Kendall et al. (2015) en régression de pose de caméra à partir d'un réseau de neurones profond ont notamment mis en évidence le fait que ce réglage puisse avoir un impact important sur les performances de

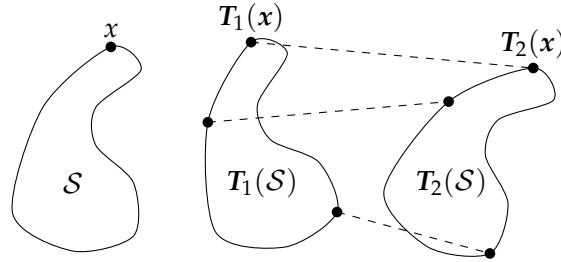


FIGURE 3.3 – Représentation de paires de points en correspondance, entre des instances d'un objet rigide sans symétrie propre en des poses distinctes.

l'application visée. Dans le cas de pose d'objet rigide, un choix raisonnable consiste à fixer  $b$  à 1 et à choisir la pondération d'orientation  $a$  comme le rayon maximal de l'objet (Di Gregorio, 2008) en supposant le repère objet au centre de ce dernier, de manière à obtenir avec la distance (3.4) une borne supérieure du déplacement des points de l'objet entre deux poses.

### 3.3.4 Approches géométriques

Certaines distances s'appuient sur les propriétés géométriques de l'objet de manière à éviter de recourir au choix de pondérations arbitraires. Une approche particulièrement intéressante consiste à définir une métrique reposant sur la distance entre points 3D en correspondance d'instances de l'objet en les poses  $T_1$  et  $T_2$ , ainsi qu'illustré figure 3.3. En notant  $\mu$  une distribution de densité définie sur le volume de l'objet, et  $V = \int \mu(x) dv$  l'intégrale de celle-ci sur l'objet complet, on peut définir une telle distance sous forme d'une norme  $L^p$  (avec  $p \geq 1$ ) ainsi :

$$d(T_1, T_2) = \frac{1}{V} \left( \int \mu(x) \|T_2(x) - T_1(x)\|^p dv \right)^{\frac{1}{p}}. \quad (3.9)$$

Un telle distance possède un sens physique fort. Elle est de plus objective puisque sa définition ne dépend pas du choix d'un repère particulier, et prend en compte la forme de l'objet. Martinez et Duffy (1995) suggèrent de considérer le déplacement maximal des points de l'objet ( $p = \infty$ ), tandis qu'Hinterstoisser et al. (2012b) proposent un critère basé sur le déplacement moyen ( $p = 1$ ) pour évaluer les algorithmes d'estimation de pose. Pour des raisons pratiques, ces auteurs n'évaluent l'intégrale (3.9) qu'à partir d'un nombre limité de points de l'objet. Kazerounian et Rastegar (1992) considèrent au contraire l'intégrale des carrés des déplacements ( $p = 2$ ) sur l'objet complet, et montrent que celle-ci peut être évaluée de manière efficace à partir du tenseur d'inertie de l'objet. Chirikjian et Zhou (1998) étendent cette distance aux transformations affines arbitraires, et montrent qu'elle peut être interprétée comme une norme de Frobenius pondérée.

Zefran et Kumar (1996) et Lin et Burdick (2000) proposent également un tenseur riemannien lié à la notion d'énergie cinétique. Celui-ci peut être vu comme un équivalent local de la distance de Kazerounian et Rastegar (1992).

Cependant, il n'existe pas, à notre connaissance, d'expression analytique de la distance géodésique résultante dans le cas général.

### 3.3.5 Paramétrisation locale

Enfin, d'autres approches s'appuient sur une paramétrisation locale de l'espace de pose, et considèrent la distance euclidienne dans l'espace des paramètres. Parmi ces paramétrisations, on retrouve notamment la représentation de rotation sous forme d'angles d'Euler, ou encore une projection stéréographique de l'espace de pose identifié à la *quadrique de Study* – une hypersurface de dimension 6 plongée dans  $\mathbb{R}^7$  – proposée par [Eberharter et Ravani \(2004\)](#). Une discussion plus poussée de ce sujet sort néanmoins de notre cadre de recherche, qui se focalise sur les distances globales et non les métriques locales.

## 3.4 Distance proposée

Dans cette section, nous proposons une distance sur l'espace de pose d'un objet rigide borné dont l'expression est valable même pour les objets symétriques, et abordons quelques unes de ses propriétés. Cette distance peut être vue comme une extension des travaux de [Kazerounian et Rastegar \(1992\)](#) et [Chirikjian et Zhou \(1998\)](#) au cas d'objets bornés arbitraires.

### 3.4.1 Définition de la distance entre poses

Soit  $\mathcal{S}$  l'ensemble des points de l'objet dans une pose de référence  $\mathcal{P}_0 \in \mathcal{C}$ , et  $\mu$  une distribution de densité positive définie sur  $\mathcal{S}$ . L'ensemble des points de l'objet ainsi que la distribution de densité associée sont supposés être compatibles avec les propriétés de symétrie de l'objet, et posséder les mêmes propriétés de symétrie. De manière formelle, nous considérons que ceux-ci vérifient  $G(\mathcal{S}) = \mathcal{S}$  et  $(\mu \circ G) = \mu$  pour toute symétrie propre  $G \in G$ .

**Définition 2.** Soient  $\mathcal{P}_1, \mathcal{P}_2 \in \mathcal{C}$  deux poses et  $T_1, T_2 \in SE(3)$  deux transformations rigides dont les classes d'équivalence sont respectivement identifiées à  $\mathcal{P}_1$  et  $\mathcal{P}_2$  étant donné la pose de référence, suivant la définition (3.2). Nous définissons la distance entre  $\mathcal{P}_1$  et  $\mathcal{P}_2$  comme suit :

$$d(\mathcal{P}_1, \mathcal{P}_2) \triangleq \min_{G_1, G_2 \in G} d_{\text{no\_sym}}(T_1 \circ G_1, T_2 \circ G_2),$$

avec

$$d_{\text{no\_sym}}(T_1, T_2) \triangleq \sqrt{\frac{1}{S} \int_{\mathcal{S}} \mu(x) \|T_2(x) - T_1(x)\|^2 ds} \quad (3.10)$$

$$\text{où } S \triangleq \int_{\mathcal{S}} \mu(x) ds.$$

L'expression ci-dessus est bien définie : le minimum de la définition (3.10) est atteint du fait de la compacité du groupe des symétries propres  $G$  – en tant que sous-groupe fermé de  $SO(3)$  qui est lui-même compact – et de la

continuité de  $d_{\text{no\_sym}}$ . Cette définition est de plus par construction indépendante du choix des transformations rigides  $T_1, T_2$  représentant les poses considérées. On vérifie aisément qu'elle satisfait les critères de définition d'une distance :  $d$  est symétrique, définie positive, et vérifie l'inégalité triangulaire. Cette dernière propriété découle de l'inégalité triangulaire vérifiée par  $d_{\text{no\_sym}}$ , par conséquence directe de l'inégalité de Minkowski. Une formulation équivalente de cette distance n'impliquant qu'une minimisation simple (et non double) sur  $G$  est introduite plus loin proposition 3.

Dans le cadre d'une application d'estimation de pose, l'objectif attendu correspond typiquement au bon positionnement de la surface de l'objet. Aussi, nous considérons dans le cadre de nos expériences la surface de l'objet comme ensemble  $\mathcal{S}$ . La fonction de densité  $\mu$  peut être utilisée afin de moduler l'importance de positionnement de certaines régions spécifiques de celle-ci, mais faute d'information additionnelle il est pertinent de considérer une distribution uniforme  $\mu = 1$ .

### 3.4.2 Objectivité

La distance proposée est par construction indépendante du choix de repères arbitraires, en ce qu'elle admet une interprétation purement géométrique (nous discutons ce point sous-section 3.4.3).

La définition 2 ne repose en effet sur aucune hypothèse concernant le choix d'un repère objet, et l'utilisation d'une pose de référence dans notre formulation – c.-à-d. d'un repère inertiel – n'a pour but que de faciliter l'écriture de celle-ci. En effet, la distance euclidienne entre points 3D est invariante par application d'isométries (par définition de celles-ci), et en particulier est invariante à toute transformation rigide  $T_3^{-1} \in SE(3)$  :

$$\forall x, y \in \mathbb{R}^3, \|x - y\| = \|T_3^{-1}(x) - T_3^{-1}(y)\|. \quad (3.11)$$

Aussi, une pose arbitraire  $\mathcal{P}_3 \in \mathcal{C}$  peut-elle être considérée comme nouvelle pose de référence sans pour autant avoir d'effet sur les propriétés métriques de l'espace. En notant  $T_3$  une transformation rigide identifiée à  $\mathcal{P}_3$  relativement à l'ancienne pose de référence  $\mathcal{P}_0$ , nous vérifions l'indépendance de  $d_{\text{no\_sym}}$  par rapport au choix de la pose de référence

$$d_{\text{no\_sym}}(T_1, T_2) = d_{\text{no\_sym}}(T_3^{-1}T_1, T_3^{-1}T_2), \quad (3.12)$$

et par la même l'indépendance de notre distance :

$$d([T_1], [T_2]) = d([T_3^{-1}T_1], [T_3^{-1}T_2]). \quad (3.13)$$

### 3.4.3 Interprétation géométrique

Un croquis valant mieux qu'un long discours, les explications développées dans cette section sont illustrées figure 3.4 sur l'exemple d'un objet 2D présentant une symétrie cyclique d'ordre 3 : une fleur à 3 pétales.

Ainsi que nous l'avons abordé section 3.2, une pose  $\mathcal{P}_i \in \mathcal{C}$  peut être identifiée à un ensemble de transformations rigides  $\{T_i \circ G, G \in G\}$ . Chacune de ces transformations peut elle-même être identifiée à la pose d'un

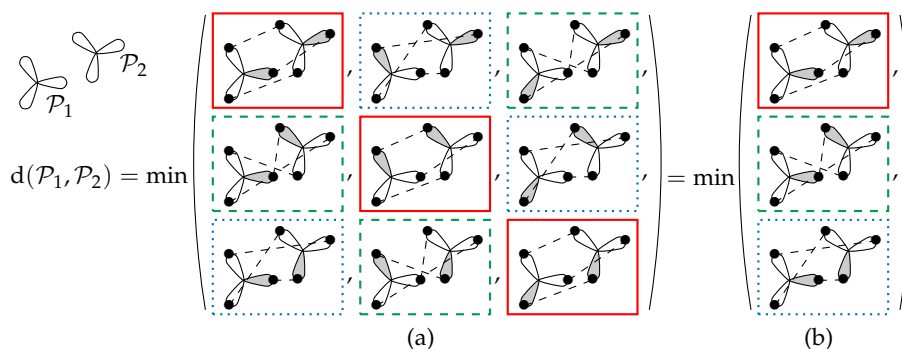


FIGURE 3.4 – Illustration de la distance proposée pour un objet 2D présentant une symétrie cyclique d'ordre 3. (a) Nous définissons la distance entre deux poses comme la distance minimale entre deux poses d'un objet équivalent ne présentant pas de propriétés de symétrie. Ici, à chaque pose de l'objet original sont associées 3 poses de l'objet équivalent. La distance entre poses d'un objet sans symétrie propre correspond à la distance RMS entre les points de l'objet en correspondance (segments pointillés). (b) De manière équivalente, la distance proposée peut être vue comme une mesure du plus petit déplacement d'une pose à une autre. Ici, il existe 3 déplacements différents entre ces dernières (encadrés en traits pleins, tirets et pointillés).

objet de caractéristiques identiques à l'objet considéré, mais ne présentant pas de propriété de symétrie propre. Nous désignons par *objet équivalent* un tel objet, que nous représentons figure 3.4 avec un pétale grisé de manière à briser la symétrie de l'objet initial. Une pose de l'objet initial peut ainsi être vue comme un ensemble de poses de l'objet équivalent (3 dans notre exemple). Puisque ce dernier ne présente pas de symétrie propre, les points de l'objet équivalent peuvent être mis en correspondance de manière non ambiguë entre différentes poses; correspondances que nous représentons par des segments pointillés sur la figure. Il est dès lors légitime de définir une distance entre poses de l'objet équivalent à partir des distances entre de tels points en correspondance, et nous considérons plus spécifiquement la distance  $d_{\text{no\_sym}}$ , car celle-ci permet des calculs efficaces (voir section 3.5) :

$$d_{\text{no\_sym}}^2(T_1, T_2) = \frac{1}{S} \int_S \mu(x) \|T_2(x) - T_1(x)\|^2 ds. \quad (3.14)$$

La distance entre deux poses de l'objet initial peut alors être définie comme la distance minimale entre chaque paire potentielle de poses de l'objet équivalent ( $3 \times 3$  combinaisons dans notre exemple<sup>1</sup>) :

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{G_1, G_2 \in G} d_{\text{no\_sym}}(T_1 \circ G_1, T_2 \circ G_2). \quad (3.15)$$

Une autre interprétation plus intuitive consiste à considérer notre distance comme une mesure du plus petit déplacement d'une pose à l'autre.

1. Le nombre de combinaisons à envisager est infini dans le cas de certains groupes de symétrie propre, notamment dans le cas d'un objet de révolution.

Un déplacement d'une pose  $\mathcal{P}_1$  à une pose  $\mathcal{P}_2$  est une transformation rigide relative d'une pose de l'objet équivalent correspondant à  $\mathcal{P}_1$  vers une pose de l'objet équivalent correspondant à  $\mathcal{P}_2$ , et la longueur d'un déplacement est mesurée via  $d_{\text{no\_sym}}$ . Différentes paires de poses de l'objet équivalent sont en fait liées par le même déplacement ainsi qu'illustré figure 3.4 où celles-ci sont mises en évidence par des encadrements identiques. En réalité, l'ensemble des déplacements d'une pose  $\mathcal{P}_1$  à une pose  $\mathcal{P}_2$  sont envisagés en choisissant arbitrairement une pose  $T_1$  de l'objet équivalent pour  $\mathcal{P}_1$ , et en considérant les transformations rigides de  $T_1$  aux poses de l'objet équivalent correspondant à  $\mathcal{P}_2$  (figure 3.4b). Grâce à cela, la distance entre deux poses peut être évaluée en ne tenant en compte des symétries que pour l'une des deux :

**Proposition 3.** Pour toute poses  $\mathcal{P}_1, \mathcal{P}_2 \in \mathcal{C}$ , et  $T_1, T_2 \in SE(3)$  deux transformations rigides dont les classes d'équivalences sont respectivement associées à  $\mathcal{P}_1$  et  $\mathcal{P}_2$  étant donné la pose de référence,

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{G \in G} d_{\text{no\_sym}}(T_1, T_2 \circ G). \quad (3.16)$$

Cette formulation est plus simple que celle de la définition 2, cependant elle ne met pas en évidence la symétrie de rôles de  $\mathcal{P}_1$  et  $\mathcal{P}_2$ .

**Démonstration.** De manière formelle, l'expression (3.16) peut être déduite de la définition 2 comme suit. Étant donné deux symétries propres  $G_1, G_2 \in G$ , il est possible d'effectuer les changements de variables  $x \leftarrow G_1(x)$  et  $G \leftarrow G_2 \circ G_1^{-1}$  afin d'écrire l'égalité suivante :

$$\begin{aligned} & d_{\text{no\_sym}}^2(T_1 \circ G_1, T_2 \circ G_2) \\ &= \frac{1}{S} \int_{\mathcal{S}} \mu(x) \|T_2 \circ G_2(x) - T_1 \circ G_1(x)\|^2 ds \\ &= \frac{1}{S} \int_{G_1(\mathcal{S})} \mu(G_1^{-1}(x)) \|T_2 \circ G(x) - T_1(x)\|^2 ds \end{aligned} \quad (3.17)$$

La symétrie de l'ensemble des points de l'objet ainsi que de sa densité assure que  $G_1(\mathcal{S}) = \mathcal{S}$  et  $\mu \circ G_1^{-1} = \mu$ , ce qui conduit au résultat suivant à partir duquel la conclusion est directe :

$$\begin{aligned} & d_{\text{no\_sym}}^2(T_1 \circ G_1, T_2 \circ G_2) \\ &= \frac{1}{S} \int_{\mathcal{S}} \mu(x) \|T_2 \circ G(x) - T_1(x)\|^2 ds \\ &= d_{\text{no\_sym}}^2(T_1, T_2 \circ G). \end{aligned} \quad (3.18)$$

□

### 3.4.4 Anisotropie de rotation

Dans le cas d'un déplacement correspondant à une pure rotation autour du centre de masse d'un objet ne présentant pas de symétrie propre, les

métriques usuelles dépendent uniquement de l'angle de la rotation relative entre les deux poses.  $d_{\text{no\_sym}}$  en revanche, ainsi que la distance proposée, prend en compte la géométrie de l'objet et dès lors dépend également de l'axe de rotation considéré.

De manière plus précise, la distance entre deux poses liées par un tel déplacement est fonction de l'angle  $\theta$  et du moment d'inertie  $I_k$  le long de l'axe  $k$  de la rotation relative entre les deux poses, ainsi que suit :

$$d_{\text{no\_sym}}(T_1, T_2) = 2\sqrt{I_k} \sin\left(\frac{\theta}{2}\right) \quad (3.19)$$

où  $I_k = \frac{1}{S} \int \mu(x) \|k \times x\|^2 ds.$

Ce résultat peut être obtenu en injectant la formule de rotation de Rodrigues

$$R\mathbf{x} = \mathbf{x} + (1 - \cos(\theta))(k \times (k \times \mathbf{x})) + \sin(\theta)(k \times \mathbf{x}) \quad (3.20)$$

dans l'expression de la distance proposée. La figure 3.5 illustre cette propriété dans le cas d'un objet présentant une anisotropie prononcée consistant en un modèle de tour Eiffel taille réelle, pour deux couples de poses liées par un plus petit déplacement correspondant à une rotation de  $15^\circ$  autour de deux axes distincts. Malgré le fait que l'angle de la rotation relative soit identique dans les deux cas, le déplacement des points de la surface de l'objet est significativement différent et on tend visuellement à considérer les poses dans la configuration (b) plus distantes entre elles que celles dans la configuration (a). Notre distance formalise cette intuition, résultant d'une distance entre les poses dans la configuration (b) environ 2.1 fois plus grande que dans la configuration (a).

### 3.5 Calculs efficaces de distance

La définition 2 et même la proposition 3 plus simple sont d'une utilité limitée en pratique pour évaluer des distances entre poses, car celles-ci font figurer une intégration sur l'ensemble des points de l'objet et une minimisation sur le groupe des symétries propres, et ces deux ensembles sont potentiellement infinis.

Aussi dans cette section, nous montrons comment la distance proposée peut être évaluée de manière efficace. Dans ce but, nous proposons une représentation d'une pose  $\mathcal{P}$  sous forme d'un ensemble fini de points  $\mathcal{R}(\mathcal{P})$  d'un espace euclidien  $\mathbb{R}^N$  d'au plus 12 dimensions (selon les symétries de l'objet). Nous désignons par *représentant* un élément de  $\mathcal{R}(\mathcal{P})$ , car il définit complètement une pose (voir section 3.7).

À l'aide de ce formalisme, il devient possible d'exprimer la distance entre deux poses  $\mathcal{P}_1, \mathcal{P}_2$  comme le minimum de la distance euclidienne entre leurs représentants respectifs

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{p_1 \in \mathcal{R}(\mathcal{P}_1), p_2 \in \mathcal{R}(\mathcal{P}_2)} \|p_2 - p_1\| \quad (3.21)$$

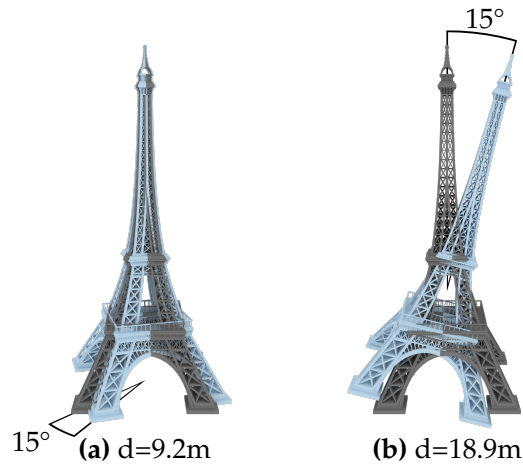


FIGURE 3.5 – Anisotropie de rotation : les métriques usuelles considèrent les distances entre les deux poses des configurations (a) et (b) égales, car ces poses sont liées dans les deux cas par une rotation de même angle autour du centre de masse de l’objet. Notre distance prend en compte la géométrie de l’objet et discrimine ces deux configurations.

ou, de manière équivalente, comme le minimum de la distance euclidienne entre un représentant d’une pose et les représentants de l’autre, suivant un raisonnement similaire à celui développé dans la démonstration de la proposition 3 :

$$\forall p_1 \in \mathcal{R}(\mathcal{P}_1), d(\mathcal{P}_1, \mathcal{P}_2) = \min_{p_2 \in \mathcal{R}(\mathcal{P}_2)} \|p_2 - p_1\|. \quad (3.22)$$

Le cardinal de  $\mathcal{R}(\mathcal{P})$  est indépendant de la pose considéré et dépend seulement de la classe de symétrie propre de l’objet, aussi nous notons celui-ci  $|\mathcal{R}(\bullet)|$ . Pour la plupart des classes d’objet – objets sans symétrie propre, objets sphériques ou objets de révolution sans invariance par rotoréflexion – une pose admet un unique représentant et on désigne dans ce cas ce dernier par  $\mathcal{R}(\mathcal{P})$  par abus de notation.  $\mathcal{R}$  représente alors un isométrie de  $\mathcal{C}$  vers  $\mathbb{R}^N$ , et la distance entre deux poses correspond alors simplement à la distance euclidienne entre leurs représentants respectifs.

Les expressions de ces représentants de pose pour les différentes classes d’objet seront déduites des développements présentés dans la suite de cette section, et sont synthétisées dans les tableaux 3.6 et 3.7.

### 3.5.1 Recherche de voisinage

Le caractère « quasi-euclidien » de l’expression (3.22) de notre distance présente un grand intérêt pratique, et permet notamment de réaliser efficacement des requêtes de voisinage parmi de larges ensembles de poses en utilisant n’importe quel algorithme adapté aux espaces euclidiens. La recherche de voisinage – notamment la recherche de k-plus proches voisins



TABLE 3.6 – Expression des représentants proposés pour une pose  $\mathcal{P} = [(\mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^3)]$  d'un objet 3D suivant sa classe de symétrie propre.

<b>Convention</b>		
Centre de masse de l'objet choisi comme origine du repère objet.		
Axe de révolution choisi comme axe $e_z$ du repère objet dans le cas d'un objet de révolution.		
<b>Notations</b>		
$\mathbf{\Lambda} \triangleq \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \mathbf{x} \mathbf{x}^\top ds \right)^{1/2}$		
et $\lambda \triangleq \sqrt{\lambda_r^2 + \lambda_z^2}$ pour les objets de révolution où $\mathbf{\Lambda} = \text{diag}(\lambda_r, \lambda_r, \lambda_z)$ .		
Classe de symétrie	Groupe de symétrie propre $G$	Représentants de pose $\mathcal{R}(\mathcal{P})$
Sphérique	$SO(3)$	$\mathbf{t} \in \mathbb{R}^3$
Révolution sans invariance par rotoréflexion	$\{\mathbf{R}_z^\alpha \mid \alpha \in \mathbb{R}\}$	$(\lambda(\mathbf{R}e_z)^\top, \mathbf{t}^\top)^\top \in \mathbb{R}^6$
Révolution avec invariance par rotoréflexion	$\{\mathbf{R}_x^\delta \mathbf{R}_z^\alpha \mid \delta \in \{0, \pi\}, \alpha \in \mathbb{R}\}$	$\{(\pm \lambda(\mathbf{R}e_z)^\top, \mathbf{t}^\top)^\top\} \subset \mathbb{R}^6$
Sans symétrie propre	$\{\mathbf{I}\}$	$(\text{vec}(\mathbf{R}\mathbf{\Lambda})^\top, \mathbf{t}^\top)^\top \in \mathbb{R}^{12}$
Fini non trivial	Fini	$\{(\text{vec}(\mathbf{R}\mathbf{G}\mathbf{\Lambda})^\top, \mathbf{t}^\top)^\top \mid \mathbf{G} \in G\} \subset \mathbb{R}^{12}$

TABLE 3.7 – Expression des représentants proposés pour une pose  $\mathcal{P} = [(\theta \in \mathbb{R}, \mathbf{t} \in \mathbb{R}^2)]$  d'un objet 2D suivant sa classe de symétrie propre.

<b>Convention</b>		
Centre de masse de l'objet choisi comme origine du repère objet.		
<b>Notations</b>		
$\forall \alpha \in \mathbb{R}, e^{i\alpha} = (\cos(\alpha), \sin(\alpha))$ , et $\lambda \triangleq \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \ \mathbf{x}\ ^2 ds \right)^{1/2}$ .		
Classe de symétrie	Groupe de symétries propre $G$	Représentants de pose $\mathcal{R}(\mathcal{P})$
Circulaire	$SO(2)$	$\mathbf{t} \in \mathbb{R}^2$
Sans symétrie propre	$\{\mathbf{I}\}$	$(\lambda e^{i\theta}, \mathbf{t}^\top)^\top \in \mathbb{R}^4$
Cyclique (d'ordre $n \in \mathbb{N}^*$ )	$\{\mathbf{R}^{2k\pi/n} \mid k \in \llbracket 0, n \rrbracket\}$	$\{(\lambda e^{i(\theta+2k\pi/n)}, \mathbf{t}^\top)^\top \mid k \in \llbracket 0, n \rrbracket\} \subset \mathbb{R}^4$

(exacte ou approximative) ou la recherche des voisins dans un rayon donné – est une opération que l’on retrouve dans de nombreux problèmes, et la section 3.10 présente un exemple où la recherche de poses dans un rayon donné est intensivement utilisée.

Les méthodes existantes adaptées à des espaces vectoriels munis d’une norme  $L^1$  ou  $L^2$  permettent d’accélérer ces recherches comparé à l’approche brutale consistant à estimer la distance du point courant à chacun des points de l’ensemble. Elles s’appuient pour se faire sur des structures de recherche particulières adaptées aux propriétés de l’espace métrique, ainsi qu’à la distribution de l’ensemble de points dans lequel réaliser des requêtes. Un état de l’art exhaustif de ces méthodes sort du cadre de ces travaux, nous citerons néanmoins les approches à bases de grilles ou de *kD-trees* adaptées aux espaces de dimension limitée, et les méthodes de type *Locality Sensitive Hashing* notamment utilisées pour des recherches approximatives dans des espaces de grande dimension. Le lecteur intéressé par ce sujet est renvoyé vers la bibliothèque FLANN (Muja et Lowe, 2009) comme point de départ.

Soit  $S$  un ensemble fini de poses. Nous désignons par  $R$  l’ensemble de points correspondant à l’agrégation de l’ensemble des représentants des poses de  $S$  :

$$R = \bigcup_{\mathcal{P} \in S} \mathcal{R}(\mathcal{P}). \quad (3.23)$$

D’après l’égalité (3.22) et étant donné une pose  $Q$  et  $q \in \mathcal{R}(Q)$  un de ses représentants, les poses dans  $S$  distantes de  $Q$  de moins d’un seuil  $\delta \in \mathbb{R}^+$  sont les poses ayant un représentant distant de  $q$  de moins que  $\delta$  suivant la distance euclidienne de  $\mathbb{R}^N$ . L’ensemble de ces représentants peut être obtenu au moyen d’une opération standard de recherche dans  $R \subset \mathbb{R}^N$  des voisins dans un rayon donné autour de  $q \in \mathbb{R}^N$ , à l’aide d’une des méthodes adaptées aux espaces euclidiens évoquée plus haut. La recherche du plus proche voisin d’une pose peut être réalisée de manière similaire.

Il convient cependant de prendre garde aux doublons potentiels lors de ces opérations, car plusieurs représentants dans  $R$  peuvent correspondre à la même pose suivant les symétries de l’objet. L’absence de doublons est néanmoins garantie localement autour du point de requête  $q$  dans une boule ouverte de rayon  $T/2$ , où  $T$  est une constante définie ainsi :

**Définition 4 (Distance minimale entre représentants).** Nous notons  $T$  la distance minimale entre différents représentants d’une même pose, avec la convention  $T = +\infty$  dans le cas où une pose n’admet qu’un seul représentant :

$$\forall \mathcal{P} \in \mathcal{C}, \forall \mathcal{p} \in \mathcal{R}(\mathcal{P}), T \triangleq \min_{q \in \mathcal{R}(\mathcal{P}), q \neq \mathcal{p}} \|q - \mathcal{p}\|. \quad (3.24)$$

$T$  peut être estimé en considérant une pose arbitraire  $\mathcal{P}$  du fait de l’invariance de notre distance relativement au choix d’une pose de référence (cf. section 3.4.2), et en considérant un représentant arbitraire de celle-ci  $\mathcal{p} \in \mathcal{R}(\mathcal{P})$  du fait des propriétés de symétrie des représentants qui seront abordées section 3.6.

### 3.5.2 Décomposition en translation et orientation

À partir de ce point, nous considérons un repère direct orthonormé  $(O, e_x, e_y, e_z)$  de manière à pouvoir exprimer les coordonnées de points 3D. De même que dans la section 3.2.3 sur le groupe des symétries propres, nous supposons que l'origine  $O$  du repère objet est un point invariant de l'objet, relativement aux symétries de celui-ci.  $O$  est par exemple choisi au centre de l'objet si celui-ci est sphérique, et sur l'axe de révolution dans le cas d'un objet de révolution. Ce faisant, le groupe des symétries propres peut être assimilé à un groupe de rotations autour de l'origine, et nous représentons donc les symétries propres sous forme de matrices de rotation. Nous utilisons cette propriété pour développer le terme intérieur de la distance (3.10) au carré en :

$$\begin{aligned} & \| (T_2 \circ G_2)(x) - (T_1 \circ G_1)(x) \|^2 \\ &= \| R_2 G_2 x + t_2 - (R_1 G_1 x + t_1) \|^2 \\ &= \| R_2 G_2 x - R_1 G_1 x \|^2 + \| t_2 - t_1 \|^2 \\ &\quad + 2 \underbrace{(t_2 - t_1)^\top (R_2 G_2 - R_1 G_1) x}_{(\#)}. \end{aligned} \quad (3.25)$$

Nous ajoutons également la contrainte supplémentaire que l'origine  $O$  du repère objet soit choisie au centre de masse de la surface de l'objet, c.-à-d.  $\int_S \mu(x) x ds = \mathbf{0}$ . Cette contrainte est compatible avec la précédente car le centre de masse est unique, aussi celui-ci doit nécessairement être invariant aux symétries propres de l'objet. Grâce à ce choix, le terme (#) de l'égalité (3.25) est d'intégrale nulle, et le carré de la distance (3.10) peut alors être décomposé en une partie liée à la position et une partie  $d_{\text{rot}}$  liée à l'orientation de l'objet :

$$d^2(\mathcal{P}_1, \mathcal{P}_2) = \| t_2 - t_1 \|^2 + \underbrace{\min_{G_1, G_2 \in G} \frac{1}{S} \int_S \mu(x) \| R_2 G_2 x - R_1 G_1 x \|^2 ds}_{d_{\text{rot}}^2(R_1, R_2)}. \quad (3.26)$$

Dans les sous-sections suivantes, nous montrons comment simplifier ce terme d'orientation, et comment cela nous amène à la notion de représentant de pose.

### 3.5.3 Objet sans symétrie propre

Considérons tout d'abord le cas d'un objet sans symétrie propre. Le groupe de symétrie propre d'un tel objet est réduit à la rotation identité, aussi le terme d'orientation introduit dans l'égalité (3.26) peut être exprimé comme suit :

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = \int_S \mu(x) \| R_2 x - R_1 x \|^2 ds. \quad (3.27)$$

En réécrivant la partie intérieure de (3.27) au moyen de l'opérateur trace,

$$\| R_2 x - R_1 x \|^2 = \text{Tr} \left( (R_2 - R_1) x x^\top (R_2 - R_1)^\top \right), \quad (3.28)$$

il est possible d'exprimer le terme d'orientation de la distance sous forme d'une distance de Frobenius pondérée entre les matrices représentant les orientations des deux poses :

$$\begin{aligned} d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) &= \text{Tr} \left( (\mathbf{R}_2 - \mathbf{R}_1) \mathbf{\Lambda}^2 (\mathbf{R}_2 - \mathbf{R}_1)^\top \right) \\ &= \|\mathbf{R}_2 \mathbf{\Lambda} - \mathbf{R}_1 \mathbf{\Lambda}\|_F^2, \end{aligned} \quad (3.29)$$

où  $\mathbf{\Lambda}$  représente la racine carré symétrique semi-définie positive de la matrice de covariance de l'ensemble des points de l'objet :

$$\mathbf{\Lambda} \triangleq \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \mathbf{x} \mathbf{x}^\top ds \right)^{1/2}. \quad (3.30)$$

$\mathbf{\Lambda}$  ne dépend pas de la pose considérée et peut dès lors être estimée une fois pour toute pour un objet donné. Nous présentons notamment dans l'annexe A un formulaire permettant d'estimer cette dernière dans le cas où  $S$  est définie sous forme d'un ensemble de triangles, ce qui est une représentation usuelle de la surface d'un objet 3D.

On dispose alors d'une expression analytique permettant d'estimer la distance entre poses, et en notant  $\text{vec}(\bullet)$  l'opérateur vectorisant par colonne une matrice de dimension  $(n, m) \in (\mathbb{N}^*)^2$  en un vecteur de dimension  $n \cdot m$ , il est possible de définir comme suit une isométrie  $\mathcal{R}$  de l'espace de pose vers  $\mathbb{R}^{12}$  muni de la distance euclidienne

$$\begin{aligned} &\text{Objet sans symétrie propre :} \\ d^2(\mathcal{P}_1, \mathcal{P}_2) &= \|\mathbf{R}_2 \mathbf{\Lambda} - \mathbf{R}_1 \mathbf{\Lambda}\|_F^2 + \|\mathbf{t}_2 - \mathbf{t}_1\|^2 \\ &= \|\mathcal{R}(\mathcal{P}_2) - \mathcal{R}(\mathcal{P}_1)\|^2 \\ &\text{avec } \mathcal{R}(\mathcal{P}) \triangleq \left( \text{vec}(\mathbf{R} \mathbf{\Lambda})^\top, \mathbf{t}^\top \right)^\top \in \mathbb{R}^{12}. \end{aligned} \quad (3.31)$$

Le passage d'une représentation de pose sous forme d'une matrice de rotation  $\mathbf{R}$  et d'un vecteur de translation  $\mathbf{t}$  à son représentant dans  $\mathbb{R}^{12}$  est direct puisqu'il s'agit d'une simple opération linéaire. Dans le cas où le repère objet est choisi aligné avec les axes principaux de l'objet,  $\mathbf{\Lambda}$  est diagonale, ce qui rend cette opération encore moins coûteuse en terme de calcul.

### 3.5.4 Objet de révolution sans invariance par rotoréflexion

Considérons maintenant le cas d'un objet de révolution sans invariance par rotoréflexion. Comme convenu dans la section 3.5.2, nous supposons que l'origine du repère objet est confondue avec le centre de masse de ce dernier. Sans perte de généralité, nous supposons de plus que l'axe  $e_z$  du repère objet est aligné avec l'axe de révolution. Une pose  $\mathcal{P}$  d'un tel objet est donc définie à une rotation près  $\mathbf{R}_z^\phi$  autour de l'axe  $e_z$ , où  $\phi$  représente l'angle de la rotation considérée. Le groupe de symétrie propre de l'objet peut donc s'exprimer sous la forme  $G = \left\{ \mathbf{R}_z^\phi \mid \phi \in \mathbb{R} \right\}$ .

La simplification réalisée section 3.5.3 afin de se débarrasser de l'intégrale sur l'ensemble des points de l'objet grâce à l'introduction de la matrice  $\Lambda$  est également valide ici, ce qui permet d'écrire ainsi le terme d'orientation de la distance proposée :

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = \min_{\phi_1, \phi_2} \|\mathbf{R}_2 \mathbf{R}_z^{\phi_2} \Lambda - \mathbf{R}_1 \mathbf{R}_z^{\phi_1} \Lambda\|_F^2. \quad (3.32)$$

Du fait que  $(\mathbf{O}, \mathbf{e}_z)$  corresponde à l'axe de révolution de l'objet,  $\Lambda$  est de plus nécessairement diagonale et de la forme

$$\Lambda = \begin{pmatrix} \lambda_r & 0 & 0 \\ 0 & \lambda_r & 0 \\ 0 & 0 & \lambda_z \end{pmatrix} \quad (3.33)$$

avec  $\lambda_r, \lambda_z \in \mathbb{R}^+$ . Cet a priori permet de simplifier encore le terme d'orientation (3.32) et d'exprimer ce dernier sous forme d'une distance entre les axes de révolution de l'objet dans les deux poses,  $\mathbf{R}_1 \mathbf{e}_z$  et  $\mathbf{R}_2 \mathbf{e}_z$  :

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = (\lambda_r^2 + \lambda_z^2) \|\mathbf{R}_2 \mathbf{e}_z - \mathbf{R}_1 \mathbf{e}_z\|^2. \quad (3.34)$$

Le lecteur est renvoyé à l'annexe B pour une preuve de ce résultat.

Aussi et de manière similaire à ce que nous avons proposé pour un objet sans symétrie propre, il est possible de définir une isométrie simple  $\mathcal{R}$  qui associe à une pose d'un objet de révolution sans invariance par rotoréflexion un vecteur 6D, consistant en la concaténation des coordonnées de son axe de révolution (à un facteur d'échelle près) et de sa position, afin de pouvoir estimer de manière efficace des distances :

<p>Objet de révolution sans invariance par rotoréflexion :</p> $d^2(\mathcal{P}_1, \mathcal{P}_2) = \ \mathbf{t}_2 - \mathbf{t}_1\ ^2 + \lambda^2 \ \mathbf{R}_2 \mathbf{e}_z - \mathbf{R}_1 \mathbf{e}_z\ ^2$ $= \ \mathcal{R}(\mathcal{P}_2) - \mathcal{R}(\mathcal{P}_1)\ ^2$ <p>avec <math>\mathcal{R}(\mathcal{P}) \triangleq \left( \lambda (\mathbf{R} \mathbf{e}_z)^\top, \mathbf{t}^\top \right)^\top \in \mathbb{R}^6</math></p> <p>où <math>\lambda = \sqrt{\lambda_r^2 + \lambda_z^2}</math>.</p>	(3.35)
---	--------

### 3.5.5 Objet sphérique

Nous traitons ici le cas d'un objet à symétrie sphérique. En choisissant pour origine du repère objet le centre de ce dernier, le groupe des symétries propres de l'objet consiste en l'ensemble du groupe des rotations  $SO(3)$ , et le terme d'orientation de notre distance peut donc être exprimé par

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = \min_{\mathbf{R}_1, \mathbf{R}_2} \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \|\mathbf{R}_2 \mathbf{x} - \mathbf{R}_1 \mathbf{x}\|^2 ds \right). \quad (3.36)$$

Ce terme est nul, puisqu'il est minoré par 0 et s'annule lorsque  $\mathbf{R}_1 = \mathbf{R}_2$ . Aussi, on conclut que l'espace de pose peut être plongé de manière isométrique dans  $\mathbb{R}^3$  en représentant une pose par la position de son centre :

<p>Objet sphérique :</p> $d^2(\mathcal{P}_1, \mathcal{P}_2) = \ \mathbf{t}_2 - \mathbf{t}_1\ ^2$ $= \ \mathcal{R}(\mathcal{P}_2) - \mathcal{R}(\mathcal{P}_1)\ ^2$ <p>avec <math>\mathcal{R}(\mathcal{P}) = \mathbf{t} \in \mathbb{R}^3</math>.</p>	(3.37)
---	--------

### 3.5.6 Objet de révolution avec invariance par rotoréflexion

Considérons maintenant le cas d'un objet de révolution avec invariance par rotoréflexion, c'est à dire présentant un plan de symétrie orthogonal à l'axe de révolution. En appliquant les mêmes conventions concernant le choix du repère objet que dans le cas d'un objet de révolution sans invariance par rotoréflexion, le groupe des symétries propres de l'objet peut être exprimé ainsi :

$$G = \left\{ \mathbf{R}_x^\delta \mathbf{R}_z^\alpha \mid \alpha \in \mathbb{R}, \delta \in \{0, \pi\} \right\}. \quad (3.38)$$

La distance entre deux poses  $\mathcal{P}_1, \mathcal{P}_2 \in \mathcal{C}$  d'un tel objet correspond alors au minimum

$$\min_{(\delta_1, \delta_2) \in \{0, \pi\}^2, (\phi_1, \phi_2) \in \mathbb{R}^2} d_{\text{no\_sym}} \left( (\mathbf{R}_1 \mathbf{R}_x^{\delta_1} \mathbf{R}_z^{\phi_1}, \mathbf{t}_1), (\mathbf{R}_2 \mathbf{R}_x^{\delta_2} \mathbf{R}_z^{\phi_2}, \mathbf{t}_2) \right). \quad (3.39)$$

Nous avons montré section 3.5.4 comment évaluer une telle expression relativement à la symétrie axiale  $\{\mathbf{R}_z^\phi\}_{\phi \in \mathbb{R}}$ . En exploitant le résultat (3.35), il est possible d'exprimer la distance entre poses comme le minimum de la distance euclidienne entre deux paires de points 6D, une paire correspondant à chaque pose :

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{\delta_1, \delta_2 \in \{0, \pi\}} \|\mathbf{p}_2^{\delta_2} - \mathbf{p}_1^{\delta_1}\| \quad (3.40)$$

avec  $\mathbf{p}_i^\delta = (\lambda(\mathbf{R}_i \mathbf{R}_x^\delta \mathbf{e}_z)^\top, \mathbf{t}^\top)^\top \in \mathbb{R}^6$  les représentants de la pose  $\mathcal{P}_i$ , pour  $\delta = 0, \pi$  et  $i = 1, 2$ .

En simplifiant légèrement ces définitions sachant que  $\mathbf{R}_x^0 \mathbf{e}_z = \mathbf{e}_z$  et  $\mathbf{R}_x^\pi \mathbf{e}_z = -\mathbf{e}_z$ , on conclut qu'une pose d'un objet de révolution avec invariance par rotoréflexion peut être représentée par deux vecteurs 6D correspondant à la concaténation des coordonnées de l'axe de révolution de l'objet avec les coordonnées de la position de son centre de masse, chacune des deux orientations potentielles de l'axe étant prise en compte par un représentant :

<p>Objet de révolution avec invariance par rotoréflexion :</p> $d(\mathcal{P}_1, \mathcal{P}_2) = \min_{\mathbf{p}_1 \in \mathcal{R}(\mathcal{P}_1), \mathbf{p}_2 \in \mathcal{R}(\mathcal{P}_2)} \ \mathbf{p}_2 - \mathbf{p}_1\ $ <p>avec <math>\mathcal{R}(\mathcal{P}) \triangleq \left\{ (\pm \lambda(\mathbf{R} \mathbf{e}_z)^\top, \mathbf{t}^\top)^\top \right\} \subset \mathbb{R}^6</math>.</p>	(3.41)
--	--------

### 3.5.7 Objet présentant un nombre fini de symétries propres

Le dernier type d'objet 3D à considérer est celui d'un objet présentant un groupe de symétrie propre  $G$  fini et différent de l'identité, tel que celui de la fusée représentée tableau 3.2e. La distance entre deux poses d'un tel objet est par définition égale à

$$\min_{G_1, G_2 \in G} d_{\text{no\_sym}}((R_1 G_1, t_1), (R_2 G_2, t_2)). \quad (3.42)$$

Nous avons montré dans la section 3.5.3 qu'il est possible de représenter la pose d'un objet sans symétrie propre sous forme d'un point 12D de telle sorte que la distance  $d_{\text{no\_sym}}$  entre deux poses d'un tel objet corresponde à la distance euclidienne entre les points respectifs les représentant. Il est donc dès lors direct de conclure que la pose d'un objet ayant un groupe de symétrie propre fini peut être représenté par un ensemble fini de points 12D, de telle sorte que la distance entre deux poses corresponde au minimum de la distance euclidienne entre leurs représentant de pose respectifs :

Object avec un groupe fini de symétrie propre :

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{p_1 \in \mathcal{R}(\mathcal{P}_1), p_2 \in \mathcal{R}(\mathcal{P}_2)} \|p_2 - p_1\| \quad (3.43)$$

avec  $\mathcal{R}(\mathcal{P}) \triangleq \left\{ \left( \text{vec}(\mathbf{R}\mathbf{G}\mathbf{\Lambda})^\top, \mathbf{t}^\top \right)^\top \mid \mathbf{G} \in G \right\} \subset \mathbb{R}^{12}$ .

### 3.5.8 Objet 2D

La notion de représentant de pose définie ici peut également être étendue aux objets 2D. Pour des raisons de concision, nous ne traitons ici que du cas d'un objet 2D sans symétrie propre, car le raisonnement développé est très similaire au cas 3D. La liste complète des expressions de représentants de pose proposées est fournie tableau 3.7 pour l'ensemble des classes de symétries potentielles.

La décomposition du carré de notre distance en un terme de position et un terme d'orientation (3.26), ainsi que l'expression du terme d'orientation sous forme d'une norme de Frobenius (3.29) sont toujours valides dans le cas 2D, mais il est possible d'aller plus loin dans la simplification.

En effet, une matrice de rotation 2D peut être paramétrée par un angle  $\theta$  ainsi :

$$\mathbf{R}^\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}. \quad (3.44)$$

Aussi, en introduisant les éléments de la matrice de covariance

$$\mathbf{\Lambda}^2 = \begin{pmatrix} \lambda_{xx}^2 & \lambda_{xy}^2 \\ \lambda_{xy}^2 & \lambda_{yy}^2 \end{pmatrix}, \quad (3.45)$$

le terme d'orientation de notre distance peut être simplifié en

$$\begin{aligned} d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) &= \text{Tr} \left( (\mathbf{R}^{\theta_2} - \mathbf{R}^{\theta_1}) \mathbf{\Lambda}^2 (\mathbf{R}^{\theta_2} - \mathbf{R}^{\theta_1})^\top \right) \\ &= (\lambda_{xx}^2 + \lambda_{yy}^2) \|e^{i\theta_2} - e^{i\theta_1}\|^2 \end{aligned} \quad (3.46)$$

où on note  $e^{i\theta} \triangleq (\cos(\theta), \sin(\theta))^\top$ . Il est donc possible d'introduire dans notre cadre théorique le cas d'un objet 2D sans symétrie propre en représentant une pose d'un tel objet par un vecteur 4D, consistant en la concaténation de la représentation dans le plan complexe de son orientation et des coordonnées de son centre :

$$\begin{aligned}
 &\text{Objet 2D sans symétrie propre :} \\
 &d(\mathcal{P}_1, \mathcal{P}_2) = \min_{p_1 \in \mathcal{R}(\mathcal{P}_1), p_2 \in \mathcal{R}(\mathcal{P}_2)} \|p_2 - p_1\| \\
 &\text{avec } \mathcal{R}(\mathcal{P}) \triangleq \left( \lambda e^{i\theta}, \mathbf{t}^\top \right)^\top \in \mathbb{R}^4 \quad (3.47) \\
 &\text{où } \lambda = \left( \frac{1}{S} \int_S \mu(x) \|x\|^2 ds \right)^{1/2}.
 \end{aligned}$$

### 3.6 Symétries des représentants

Les objets dotés d'un groupe de symétrie propre fini non trivial, ainsi que les objets de révolution avec invariance par rotoréflexion admettent plusieurs représentants par pose. Cette multiplicité de représentants traduit les symétries propres de l'objet non prises en compte dans l'expression d'un représentant, et conduit à certaines propriétés de symétrie dans l'ensemble des représentants d'une pose lui-même. Ces propriétés peuvent être formalisées sous la forme d'un groupe de symétrie  $G_{\mathcal{R}}$  défini sur l'espace ambiant  $\mathbb{R}^N$ . Ce groupe est restreint au singleton identité dans le cas d'un objet n'admettant qu'un représentant par pose, et est défini tableau 3.8 pour les autres classes d'objet. Nous discutons dans cette section quelques unes des propriétés de ce groupe. Celles-ci nous seront utiles dans la section 3.8.2 afin de proposer une méthode permettant d'estimer la moyenne d'un ensemble de poses.

Nous commençons tout d'abord par vérifier que le groupe proposé est bien défini :

**Proposition 5.**  $G_{\mathcal{R}}$  est un groupe pour l'opération de composition.

*Démonstration.* Ce résultat découle directement des propriétés de groupe de  $G$ ,  $\{1, -1\}$  et  $\{e^{i2k\pi/n} | k \in \llbracket 0, n \rrbracket\}$  pour les opérations de multiplication.  $\square$

Nous introduisons ensuite le lemme suivant, qui exprime dans une certaine mesure le fait que la géométrie de l'objet est cohérente avec les symétries de ce dernier :

**Lemme 6.** Pour toute symétrie propre  $G \in G$ ,  $G$  et  $\Lambda$  commutent, c.-à-d.  $G\Lambda = \Lambda G$ .

*Démonstration.* Soit  $G \in G$ . Par définition de  $\Lambda^2$ ,

$$G\Lambda^2 = \frac{1}{S} \int_S \mu(x) Gxx^\top ds. \quad (3.48)$$



TABLE 3.8 – Expressions proposées des symétries définies sur l'espace ambiant pour les classes d'objets admettant strictement plus d'un représentant par pose.

<b>Notations</b>			
On décompose un point de l'espace ambiant $\mathbb{R}^N$ en deux parties définies comme suit, suivant la dimension $N$ de l'espace :			
— $(\text{vec}(\mathbf{M})^\top, \mathbf{t}^\top)^\top$ pour un espace 12D, avec $\mathbf{M} \in \mathcal{M}_{3,3}(\mathbb{R})$ , et $\mathbf{t} \in \mathbb{R}^3$ .			
— $(\mathbf{a}^\top, \mathbf{t}^\top)^\top$ pour un espace 6D, avec $\mathbf{a}, \mathbf{t} \in \mathbb{R}^3$ .			
— $(\mathbf{a}^\top, \mathbf{t}^\top)^\top$ pour un espace 4D, avec $\mathbf{a}, \mathbf{t} \in \mathbb{R}^2$ .			
Nous notons « $\cdot$ » la multiplication complexe dans le cas 4D, en assimilant $\mathbf{a}$ à un nombre complexe.			
Type d'objet	Classe de symétrie propre	Groupe de symétrie $G_{\mathcal{R}}$	Définition d'une symétrie
3D	Fini	$\{s_G   G \in G\}$	$s_G : \mathbb{R}^{12} \rightarrow \mathbb{R}^{12}$ $(\text{vec}(\mathbf{M})^\top, \mathbf{t}^\top)^\top \mapsto (\text{vec}(\mathbf{M}\mathbf{G})^\top, \mathbf{t}^\top)^\top$
	Révolution avec invariance par rotoréflexion	$\{s_{\text{rev},\delta}   \delta = \pm 1\}$	$s_{\text{rev},\delta} : \mathbb{R}^6 \rightarrow \mathbb{R}^6$ $(\mathbf{a}^\top, \mathbf{t}^\top)^\top \mapsto (\delta \mathbf{a}^\top, \mathbf{t}^\top)^\top$
2D	Cyclique (ordre $n \in \mathbb{N}^*$ )	$\{s_{2D,n,k}   k \in \llbracket 0, n \rrbracket\}$	$s_{2D,n,k} : \mathbb{R}^4 \rightarrow \mathbb{R}^4$ $(\mathbf{a}^\top, \mathbf{t}^\top)^\top \mapsto (e^{i2k\pi/n} \cdot \mathbf{a}, \mathbf{t}^\top)^\top$ .

Réaliser le changement de variable  $\mathbf{x} \leftarrow \mathbf{G}\mathbf{x}$  permet de réécrire cette dernière égalité en :

$$\frac{1}{S} \int_{G(S)} \mu(\mathbf{G}^{-1}\mathbf{x}) \mathbf{x} (\mathbf{G}^{-1}\mathbf{x})^\top ds. \quad (3.49)$$

Grâce à l'invariance de  $S$  Et  $\mu$  aux symétries propres de l'objet, on exhibe de nouveau  $\Lambda^2$  comme suit :

$$\begin{aligned} \mathbf{G}\Lambda^2 &= \frac{1}{S} \int_S \mu(\mathbf{x}) \mathbf{x} \mathbf{x}^\top \mathbf{G}^{-\top} ds \\ &= \Lambda^2 \mathbf{G}^{-\top}. \end{aligned} \quad (3.50)$$

$\mathbf{G}$  étant une rotation,  $\mathbf{G}^{-\top} = \mathbf{G}$ , aussi  $\mathbf{G}$  et  $\Lambda^2$  commutent, c.-à-d.

$$\mathbf{G}\Lambda^2 = \Lambda^2 \mathbf{G}. \quad (3.51)$$

En tant que matrice symétrique semi-définie positive,  $\Lambda^2$  admet une décomposition en valeurs propres  $\Lambda^2 = \mathbf{U}\mathbf{D}\mathbf{U}^\top$ , où  $\mathbf{U} \in SO(3)$  et où  $\mathbf{D}$  est une matrice diagonale semi-définie positive. En injectant cette décomposition dans le terme de droite de l'égalité 3.51, on observe que  $\mathbf{G}^\top \mathbf{U}$  est également une base de vecteurs propres de  $\Lambda^2$  :

$$\Lambda^2 = (\mathbf{G}^\top \mathbf{U}) \mathbf{D} (\mathbf{G}^\top \mathbf{U})^\top. \quad (3.52)$$

$\Lambda$  étant la racine carré principale de  $\Lambda^2$ , toutes deux partagent le même espace propre, aussi

$$\begin{cases} \Lambda = \mathbf{U}\mathbf{D}^{1/2}\mathbf{U}^\top \\ \Lambda = (\mathbf{G}^\top \mathbf{U}) \mathbf{D}^{1/2} (\mathbf{G}^\top \mathbf{U})^\top. \end{cases} \quad (3.53)$$

En injectant la première égalité dans le terme de droite de la seconde, on montre donc que

$$\Lambda = \mathbf{G}^\top \Lambda \mathbf{G}, \quad (3.54)$$

c.-à-d. que  $\mathbf{G}$  et  $\Lambda$  commutent :  $\mathbf{G}\Lambda = \Lambda\mathbf{G}$ .  $\square$

Grâce à ce lemme, il nous est maintenant possible de mettre en évidence les trois propriétés suivantes des symétries définies sur l'espace ambiant :

**Proposition 7.**  $G_{\mathcal{R}}$  contient  $|\mathcal{R}(\bullet)|$  éléments, et étant donné une pose  $\mathcal{P}$  et un représentant  $\mathbf{p} \in \mathcal{R}(\mathcal{P})$ , l'ensemble des symétries de  $\mathbf{p}$  ( $\mathbf{p}$  inclus) est l'ensemble des représentants de la pose  $\mathcal{P}$ , c.-à-d.

$$\{s(\mathbf{p}) | s \in G_{\mathcal{R}}\} = \mathcal{R}(\mathcal{P}). \quad (3.55)$$

*Démonstration.* Cette proposition se vérifie aisément à partir des définitions des représentants de pose tableaux 3.6 et 3.7. La seule subtilité réside dans le cas d'un objet 3D au groupe de symétrie propre fini. Dans ce cas pour tout  $\mathbf{G} \in G$ , le symétrique par  $s_{\mathbf{G}}$  d'un représentant de pose  $(\text{vec}(\mathbf{R}\Lambda)^\top, \mathbf{t}^\top)^\top$ , où  $\mathbf{R} \in \mathcal{M}_{3,3}(\mathbb{R})$  et  $\mathbf{t} \in \mathbb{R}^3$ , admet pour expression

$$s_{\mathbf{G}} \left( (\text{vec}(\mathbf{R}\Lambda)^\top, \mathbf{t}^\top)^\top \right) = (\text{vec}(\mathbf{R}\Lambda\mathbf{G})^\top, \mathbf{t}^\top)^\top, \quad (3.56)$$

ce qui selon le lemme 6 est égal à  $(\text{vec}(\mathbf{R}\mathbf{G}\Lambda)^\top, \mathbf{t}^\top)^\top$ . Par définition des représentants d'un tel objet, c'est donc un représentant de la pose  $\mathcal{P}$ .  $\square$

**Proposition 8.** Les éléments de  $G_{\mathcal{R}}$  sont des transformations linéaires de l'espace ambiant, c.-à-d. pour tout  $s \in G_{\mathcal{R}}$ ,  $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^N$ , et  $\alpha \in \mathbb{R}$ ,

$$s(\mathbf{x}_1 + \alpha\mathbf{x}_2) = s(\mathbf{x}_1) + \alpha s(\mathbf{x}_2). \quad (3.57)$$

*Démonstration.* Cette proposition est une conséquence directe de la définition des symétries de  $G_{\mathcal{R}}$  tableau 3.8.  $\square$

**Proposition 9.** Les éléments de  $G_{\mathcal{R}}$  sont des automorphismes de l'espace ambiant : pour tout  $s \in G_{\mathcal{R}}$ ,  $s$  est bijective et pour tout  $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^N$ ,

$$\|s(\mathbf{x}_2) - s(\mathbf{x}_1)\| = \|\mathbf{x}_2 - \mathbf{x}_1\|. \quad (3.58)$$

*Démonstration.* La bijectivité est directe à démontrer car on peut exhiber un inverse :

- $(s_{\mathbf{G}})^{-1} = s_{\mathbf{G}^{-1}}$ , pour tout  $\mathbf{G} \in G$ .
- $(s_{\text{rev},\delta})^{-1} = s_{\text{rev},\delta}$ , pour tout  $\delta \in \{-1, 1\}$ .
- $(s_{2D,n,k})^{-1} = s_{2D,n,-k}$ , pour tout  $k \in \mathbb{N}$ .

La propriété de morphisme découle de la linéarité des opérations de symétrie (proposition 8) et du fait que celles-ci préservent la norme, car les éléments de  $G$ ,  $\{1, -1\}$  et  $\{e^{i2k\pi/n} | k \in \llbracket 0, n \rrbracket\}$  sont eux-même de norme 1.  $\square$

### 3.7 Projection sur l'espace de pose

Nous avons discuté section 3.5 la manière dont il est possible d'identifier une pose  $\mathcal{P}$  à un ensemble fini de points  $\mathcal{R}(\mathcal{P})$  d'un espace euclidien  $\mathbb{R}^N$  de dimension fini, et comment les éléments de  $\mathcal{R}(\mathcal{P})$  pouvaient être calculés simplement à partir d'une transformation rigide  $(\mathbf{R}, \mathbf{t}) \in SE(3)$  associée à la pose. L'application réciproque existe, et à partir de tout élément de  $\mathcal{R}(\mathcal{P})$  il est possible d'estimer une transformation rigide décrivant totalement la pose  $\mathcal{P}$ . Aussi, nous considérons un élément de  $\mathcal{R}(\mathcal{P})$  comme un *représentant* de  $\mathcal{P}$ . Cette transformation est relativement immédiate étant donné les expressions des représentants de pose (voir tableaux 3.6 et 3.7), aussi nous discutons cette assertion dans le cadre plus général de la projection sur l'espace de pose. Étant donné un vecteur  $N$ -D arbitraire  $\mathbf{x}$ , il s'agit de trouver la pose dont le représentant est le plus similaire à  $\mathbf{x}$ . Les résultats de cette section nous seront notamment utiles dans la section 3.8 afin de proposer une méthode permettant de moyenniser des poses.

**Définition 10.** Nous définissons comme projections de  $\mathbf{x} \in \mathbb{R}^N$  l'ensemble des poses

$$\text{proj}(\mathbf{x}) \triangleq \underset{\mathcal{P}}{\text{argmin}} \min_{\mathbf{p} \in \mathcal{R}(\mathcal{P})} \|\mathbf{p} - \mathbf{x}\|^2. \quad (3.59)$$

La projection est unique dans les cas non pathologiques, et nous proposons dans les sous-sections suivantes l'expression de celle-ci selon la classe de symétrie de l'objet.

#### 3.7.1 Objet sphérique

La projection est triviale à estimer dans le cas d'un objet sphérique car alors tout point de  $\mathbb{R}^3$  constitue un représentant de pose valide. Un point  $\mathbf{x} \in \mathbb{R}^3$  se projette donc sur la pose admettant  $\mathbf{x}$  comme représentant, où autrement dit la pose dans laquelle le centre de l'objet admet  $\mathbf{x}$  pour coordonnées 3D.

#### 3.7.2 Objet de révolution

Dans le cas d'un objet de révolution sans invariance par rotoréflexion, la position du centre de masse ainsi que l'axe de révolution orienté de l'objet sont bien définis pour une pose donnée. Réciproquement, une pose peut être définie par la position de son centre masse  $\mathbf{t} \in \mathbb{R}^3$  et son axe de révolution orienté, que nous représentons par un vecteur normalisé  $\mathbf{a} \in \mathbb{R}^3$ . L'unique représentant d'une telle pose est  $(\lambda \mathbf{a}^\top, \mathbf{t}^\top)^\top$ , ainsi que défini section 3.5.4.

Soit  $\mathbf{x} \in \mathbb{R}^6$  un point à projeter sur l'espace de pose. Sans perte de généralité,  $\mathbf{x}$  peut être scindé en deux termes :  $\mathbf{x} = (\mathbf{x}_r^\top, \mathbf{x}_t^\top)^\top$ , où  $\mathbf{x}_r, \mathbf{x}_t \in \mathbb{R}^3$ . Le projeté de  $\mathbf{x}$  peut alors être exprimé comme suit :

$$\begin{aligned} \text{proj}(\mathbf{x}) &= \underset{\mathcal{P}}{\text{argmin}} \|\mathbf{x} - \mathcal{R}(\mathcal{P})\|^2 \\ &= \underset{\mathbf{a}, \mathbf{t} \in \mathbb{R}^3 / \|\mathbf{a}\|=1}{\text{argmin}} \left( \|\mathbf{x}_r - \lambda \mathbf{a}\|^2 + \|\mathbf{x}_t - \mathbf{t}\|^2 \right). \end{aligned} \quad (3.60)$$

Ce problème de minimisation admet une unique solution si et seulement si  $x_r \neq \mathbf{0}$ . Celle-ci correspond dans ce cas à la pose de centre de masse  $\hat{\mathbf{t}} = x_t$  et d'axe orienté  $\hat{\mathbf{a}} = x_r / \|x_r\|$ . Ce résultat demeure vrai dans le cas d'un objet de révolution présentant une invariance par rotoréflexion, puisque  $(\lambda \hat{\mathbf{a}}^\top, \hat{\mathbf{t}}^\top)^\top$  est alors le représentant de pose valide le plus proche de  $x$ , au sens de la norme  $L^2$ .

### 3.7.3 Objet présentant un nombre fini de symétries propres

Le représentant d'une pose d'un objet sans symétrie propre est un vecteur 12D dont les 9 premières dimensions représentent l'orientation de l'objet sous forme d'une matrice de rotation mise à l'échelle anisotropiquement et vectorisée, et les 3 autres la position de l'objet. Aussi étant donné un point  $x \in \mathbb{R}^{12}$  à projeter, nous scindons ce dernier en deux de manière à mettre en évidence ces deux composantes :  $x = (\text{vec}(\mathbf{X}_r)^\top, x_t^\top)^\top$  où  $x_t \in \mathbb{R}^3$  et  $\mathbf{X}_r \in \mathcal{M}_{3,3}(\mathbb{R})$ . La projection de  $x$  dans le cas d'un objet sans symétrie propre s'exprime alors :

$$\begin{aligned} \text{proj}(x) &= \underset{\mathcal{P}}{\text{argmin}} \|x - \mathcal{R}(\mathcal{P})\|^2 \\ &= \underset{\mathbf{R}, t}{\text{argmin}} \left( \|\mathbf{X}_r - \mathbf{R}\mathbf{\Lambda}\|_F^2 + \|x_t - t\|^2 \right) \end{aligned} \quad (3.61)$$

Les deux termes étant indépendants, on conclut une fois encore que la position du centre de masse d'une projection de  $x$  a nécessairement  $\hat{\mathbf{t}} = x_t$  pour coordonnées. Le terme d'orientation correspond quant-à lui à un problème de minimisation parfois référé sous le nom de *problème de Procruste contraint orthogonal* (Schönemann, 1966; Umeyama, 1991). Il admet pour solution  $\hat{\mathbf{R}} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$ , où  $\mathbf{U}\mathbf{D}\mathbf{V}^\top$  est une décomposition en valeur singulière de  $\mathbf{X}_r\mathbf{\Lambda}$  telle que  $\mathbf{U}, \mathbf{V} \in O(3)$  et

$$\mathbf{D} = \text{diag}(\alpha_1, \alpha_2, \alpha_3), \quad (3.62)$$

avec  $\alpha_1 \geq \alpha_2 \geq \alpha_3 \geq 0$ , et où  $\mathbf{S}$  est définie par

$$\mathbf{S} = \begin{cases} \mathbf{I} & \text{si } \det(\mathbf{U}) \det(\mathbf{V}) > 0 \\ \text{diag}(1, 1, -1) & \text{sinon.} \end{cases} \quad (3.63)$$

La solution est unique dans le cas où  $\mathbf{X}_r\mathbf{\Lambda}^\top$  est de rang supérieur ou égal à 2 (Umeyama, 1991), une condition qui est remplie dans la majorité des cas pratiques. Ce résultat vaut également dans le cas général d'un objet ayant un groupe de symétries propres fini, car  $(\text{vec}(\hat{\mathbf{R}}\mathbf{\Lambda})^\top, \hat{\mathbf{t}}^\top)^\top$  est alors le plus proche représentant d'une pose valide de  $x$ .

### 3.7.4 Objet 2D

Le problème de projection dans le cas d'un objet 2D est similaire au cas 3D :

- Dans le cas d'un objet circulaire, tout point  $x \in \mathbb{R}^2$  est le représentant d'une unique pose valide, et se projette donc sur cette pose, de centre ayant pour coordonnées  $x$ .
- Dans le cas d'un objet à symétrie cyclique d'ordre  $n \in \mathbb{N}^*$ , on conclut suivant le même raisonnement que dans le cas d'un objet de révolution 3D qu'un vecteur  $x = (\mathbf{a}^\top, \mathbf{t}^\top)^\top \in \mathbb{R}^4$ , où  $\mathbf{a}, \mathbf{t} \in \mathbb{R}^2$ , admet une unique projection si et seulement si  $\|\mathbf{a}\| \neq 0$ . Le projeté admet un représentant  $(\lambda/\|\mathbf{a}\| \cdot \mathbf{a}^\top, \mathbf{t}^\top)^\top$ , et correspond à la pose définie par la position  $\mathbf{t}$  et l'orientation 2D d'angle  $\arg(\mathbf{a})$  (modulo  $2\pi/n$ ), où  $\arg(\mathbf{a})$  est l'argument de  $\mathbf{a}$  vu comme un nombre complexe.

### 3.8 Moyenne de poses

La capacité de moyennner des poses peut être d'une grande importance dans des applications telles que le débruitage, la détection de modes ou encore l'interpolation. La définition de la moyenne n'est pas évidente dans des espaces non vectoriels tels que notre espace de pose, aussi nous considérons ici une généralisation de la notion de moyenne adaptée à des espaces métriques arbitraires, connue sous le nom de *moyenne de Fréchet*.

Étant donné un ensemble fini de poses  $\{\mathcal{P}_i\}_{i=1..n}$  et un ensemble de poids strictement positifs  $\{w_i\}_{i=1..n}$  assignés à chacune d'elles, la moyenne pondérée des poses  $\{\mathcal{P}_i\}_{i=1..n}$  est définie comme la pose minimisant la *variance de Fréchet*  $\Phi$  :

$$\text{moyenne}((\mathcal{P}_i, w_i)_{i=1..n}) \triangleq \underset{\mathcal{P} \in \mathcal{C}}{\text{argmin}} \Phi(\mathcal{P}), \quad (3.64)$$

cette variance étant définie pour une pose  $\mathcal{P} \in \mathcal{C}$  comme la somme des carrés des distances aux poses de  $S$  :

$$\Phi(\mathcal{P}) \triangleq \sum_{i=1}^n w_i d^2(\mathcal{P}_i, \mathcal{P}). \quad (3.65)$$

La moyenne de Fréchet n'est pas toujours bien définie du fait que le minimum de la variance  $\Phi$  n'est pas nécessairement atteint en une pose unique. Heureusement, ce cas ne se produit cependant typiquement que dans des configurations où la moyenne est dénuée de sens, par exemple en cherchant à moyennner deux poses d'axes opposés pour un objet de révolution sans invariance par rotoréflexion.

La question de l'estimation de la moyenne de poses a déjà été étudiée pour les objets ne présentant pas de symétrie propre et [Sharf et al. \(2010\)](#) comparent notamment différentes techniques de moyennage relatives à l'orientation d'une pose, pour des distances usuelles. Bien qu'il n'existe pas de solution analytique connue pour la métrique Riemannienne (3.4), il est possible d'en estimer une de manière itérative, et il existe des approximations analytiques « suffisamment bonnes » pour des applications pratiques ([Gramkow, 2001](#)). Une de ces approximation lorsque l'on traite avec plus de deux poses est basée sur le calcul de la moyenne arithmétique

des matrices de rotation associées à chaque pose, et correspond à la moyenne de Fréchet exacte pour la distance (3.7) (Curtis et al., 1993).

Dans le cas de notre distance, l'expression de la variance de Fréchet peut être développée grâce à l'introduction de la notion de représentant de pose en

$$\Phi(\mathcal{P}) = \sum_{i=1}^n w_i \min_{\mathbf{p}_i \in \mathcal{R}(\mathcal{P}_i), \mathbf{p} \in \mathcal{R}(\mathcal{P})} \|\mathbf{p}_i - \mathbf{p}\|^2. \quad (3.66)$$

Étant donné un n-uplet  $P = (\mathbf{p}_i)_{i=1..n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  de représentants des poses à moyenner, la somme pondérée des carrés des distances d'un représentant de pose  $\mathbf{p}$  aux éléments de  $P$  peut être scindée en deux termes, en introduisant la moyenne arithmétique  $\mathbf{m}_P$  des éléments de  $P$  :

$$\begin{aligned} \sum_{i=1}^n w_i \|\mathbf{p}_i - \mathbf{p}\|^2 &= \sum_i w_i \|\mathbf{p}_i - \mathbf{m}_P\|^2 \\ &+ \left( \sum_i w_i \right) \|\mathbf{p} - \mathbf{m}_P\|^2, \end{aligned} \quad (3.67)$$

avec la moyenne arithmétique

$$\mathbf{m}_P \triangleq \frac{\sum_i w_i \mathbf{p}_i}{\sum_i w_i}. \quad (3.68)$$

Le premier terme de l'égalité (3.67) est indépendant de  $\mathbf{p}$ , aussi minimiser cette expression pour un n-uplet  $P$  donné revient à trouver la pose  $\mathcal{P}$  qui minimise

$$\min_{\mathbf{p} \in \mathcal{R}(\mathcal{P})} \|\mathbf{p} - \mathbf{m}_P\|^2. \quad (3.69)$$

Il s'agit là du problème de projection que nous avons discuté et résolu section 3.7. La pose moyenne, si elle est bien définie, correspond donc à la projection de la moyenne arithmétique d'une combinaison de représentants des poses à moyenner, ou de manière plus formelle :

$$\text{moyenne}((\mathcal{P}_i, w_i)_{i=1..n}) = \underset{\mathcal{P} \in \mathcal{A}}{\text{argmin}} \Phi(\mathcal{P}), \quad (3.70)$$

avec

$$\mathcal{A} \triangleq \left\{ \text{proj}(\mathbf{m}_P) \mid P \in \prod_i \mathcal{R}(\mathcal{P}_i) \right\}. \quad (3.71)$$

### 3.8.1 Objets admettant un unique représentant par pose

La projection étant unique dans les cas non pathologiques, il est facile de conclure pour les objets n'admettant qu'un unique représentant par pose (objet sphérique, de révolution sans invariance par rotoréflexion, ou sans symétrie propre) car il n'y a alors qu'une seule combinaison de représentants  $P = (\mathcal{R}(\mathcal{P}_i))_{i=1..n}$  à envisager. La moyenne de poses d'un de ces objets

existe donc et correspond simplement à la projection sur l'espace de pose de la moyenne arithmétique des représentants de ces poses :

$$\boxed{\begin{array}{l} \text{Objet admettant un unique représentant par pose :} \\ \text{moyenne}((\mathcal{P}_i, w_i)_{i=1..n}) = \text{proj} \left( \frac{\sum_i w_i \mathcal{R}(\mathcal{P}_i)}{\sum_i w_i} \right). \end{array}} \quad (3.72)$$

### 3.8.2 Objects admettant plusieurs représentants par pose

Dès lors que les poses d'un objet admettent chacune plus d'un représentant, il convient de considérer les différentes combinaisons de représentants possibles afin d'estimer exactement la moyenne (en supposant son existence et son unicité) comme énoncé équation (3.70). Le nombre de combinaisons étant exponentiel en le nombre de poses considéré, ce calcul peut rapidement devenir coûteux.

Il ne s'agit là pas d'une spécificité de notre approche, et on retrouve notamment une difficulté similaire lorsque l'on cherche à approximer la moyenne d'orientations 3D au moyen de la moyenne arithmétique de représentations sous forme de quaternions unitaires, car alors chaque orientation admet deux quaternions antipodaux pour représentants. Une pratique courante pour contourner cette difficulté dans le cas de quaternions consiste à estimer la moyenne arithmétique d'une combinaison « cohérente » de représentants et considérer son projeté<sup>2</sup> pour moyenne. Une telle combinaison est généralement construite en sélectionnant un représentant d'une pose initiale arbitraire, et en sélectionnant pour chaque pose le représentant le plus proche du représentant initial (Gramkow, 2001). Cette heuristique est simple mais s'avère cependant mal définie, car la combinaison ainsi sélectionnée – et par la même l'estimation de la moyenne – est dans le cas général dépendante du choix initial. Nous représentons figure 3.9abc un exemple d'un tel cas, où 3 choix différents de pose initiale conduisent à la sélection de 3 combinaisons « cohérentes » distinctes, et par la même 3 estimations différentes de la moyenne.

Dans cette sous-section, nous proposons une définition plus stricte de la notion de *cohérence* d'une combinaison de représentants, et montrons que celle-ci permet une estimation non ambiguë de la moyenne.

**Définition 11 (Cohérence).** Un  $n$ -uplet de représentants  $(\mathbf{p}_i)_{i=1..n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  est dit *cohérent* si et seulement si

$$\forall (i, j) \in \llbracket 1, n \rrbracket^2, \forall \mathbf{q}_j \in \mathcal{R}(\mathcal{P}_j) \setminus \{\mathbf{p}_j\}, \quad \|\mathbf{p}_j - \mathbf{p}_i\| < \|\mathbf{q}_j - \mathbf{p}_i\|. \quad (3.73)$$

En d'autres termes, un  $n$ -uplet cohérent est un ensemble de représentants de poses plus proches les uns des autres que d'aucun autre représentants de ces poses.

2. c.-à-d. l'orientation correspondant au quaternion moyen normalisé.

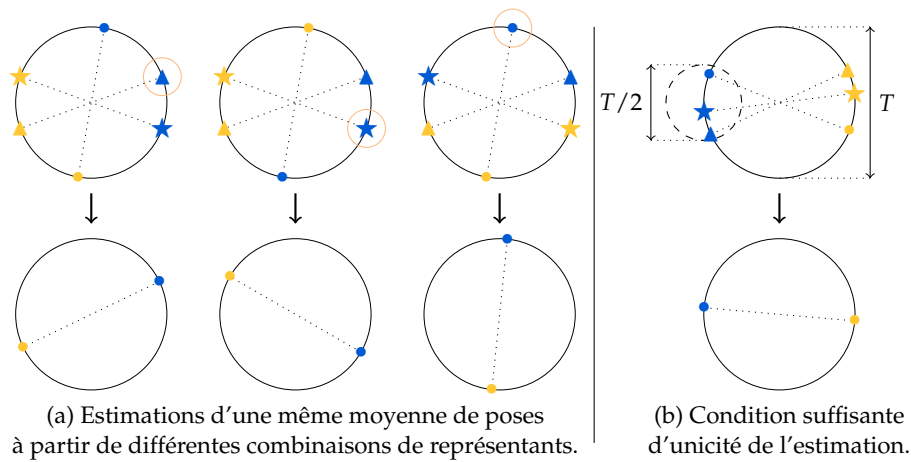


FIGURE 3.9 – Estimation de la moyenne de poses admettant chacune plusieurs représentants : illustration avec l'orientation d'un objet 2D présentant une invariance par rotation de  $180^\circ$ , qui peut être représentée par un point sur un cercle ou le point antipodal. Nous considérons 3 poses à moyenner (disque, triangle et étoile, première ligne). (a) Le choix d'une combinaison de représentants « cohérente » au sens de (Gramkow, 2001) (ensemble bleu, première ligne) est dépendant du choix de la pose initiale (entourée), ce qui conduit à des estimations différentes de la pose moyenne (seconde ligne). (b) Nous proposons une définition plus stricte de la cohérence d'une combinaison de représentants qui est en particulier satisfaite lorsque les représentants de pose considérés sont suffisamment proches les uns des autres et qui garantit une estimation non ambiguë de la pose moyenne.



**Proposition 12 (Unicité d'un n-uplet cohérent, aux symétries près).** Si  $(\mathbf{p}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  est cohérent, alors l'ensemble des n-uplets cohérents de  $\prod_i \mathcal{R}(\mathcal{P}_i)$  est l'ensemble composé de  $(\mathbf{p}_i)_{i=1\dots n}$  et de ses symétries

$$\{(s(\mathbf{p}_i))_{i=1\dots n} | s \in G_{\mathcal{R}}\}. \quad (3.74)$$

*Démonstration.* Soit  $(\mathbf{p}_i)_{i=1\dots n}, (\mathbf{q}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  deux n-uplets cohérents distincts. Il existe un  $j \in \llbracket 1, n \rrbracket$  tel que  $\mathbf{p}_j \neq \mathbf{q}_j$ , et nous savons par définition de la cohérence de  $(\mathbf{p}_i)_{i=1\dots n}$  et  $(\mathbf{q}_i)_{i=1\dots n}$  que

$$\forall i \in \llbracket 1, n \rrbracket, \begin{cases} \|\mathbf{p}_j - \mathbf{p}_i\| < \|\mathbf{q}_j - \mathbf{p}_i\| \\ \|\mathbf{q}_j - \mathbf{q}_i\| < \|\mathbf{p}_j - \mathbf{q}_i\|. \end{cases} \quad (3.75)$$

En supposant qu'il existe  $i \in \llbracket 1, n \rrbracket$  tel que  $\mathbf{p}_i = \mathbf{q}_i$ , on aboutit à l'inégalité toujours fautive  $\|\mathbf{p}_j - \mathbf{p}_i\| < \|\mathbf{p}_j - \mathbf{p}_i\|$ . On montre ainsi par l'absurde que les n-uplets  $(\mathbf{p}_i)_{i=1\dots n}$  et  $(\mathbf{q}_i)_{i=1\dots n}$  sont disjoints, et il existe donc au plus  $|\mathcal{R}(\bullet)|$  n-uplets cohérents.

Or il existe exactement  $|\mathcal{R}(\bullet)|$  combinaisons de représentants distinctes symétriques de  $(\mathbf{p}_i)_{i=1\dots n}$ , en comptant cette dernière (proposition 7) :

$$\{(s(\mathbf{p}_i))_{i=1\dots n} | s \in G_{\mathcal{R}}\}. \quad (3.76)$$

Ces combinaisons sont aussi cohérentes que  $(\mathbf{p}_i)_{i=1\dots n}$ , puisque les applications de symétrie de  $G_{\mathcal{R}}$  sont des automorphismes (proposition 9). On en conclut l'unicité – aux symétries près – d'un n-uplet cohérent de représentants de pose.  $\square$

**Proposition 13 (Invariance de la projection aux symétries des représentants).** Soit  $\mathbf{x} \in \mathbb{R}^N$  un point de l'espace ambiant et  $s \in G_{\mathcal{R}}$ . Les projections de  $\mathbf{x}$  et de son symétrique  $s(\mathbf{x})$  correspondent à une même pose :

$$\text{proj}(s(\mathbf{x})) = \text{proj}(\mathbf{x}). \quad (3.77)$$

*Démonstration.* Ce résultat se vérifie sans difficulté dans le cas d'un objet de révolution avec invariance par rotoréflexion ou d'un objet 2D à symétrie cyclique, aussi nous ne discutons ici que le cas d'un objet présentant un groupe de symétrie propre fini.

Soit  $\mathbf{x} \in \mathbb{R}^{12}$  un point de l'espace ambiant et  $s_G \in G_{\mathcal{R}}$  une symétrie de cet espace, avec  $G \in G$ . Nous scindons  $\mathbf{x}$  en deux termes  $\mathbf{M} \in \mathcal{M}_{3,3}(\mathbb{R})$  et  $\mathbf{t} \in \mathbb{R}^3$  de telle sorte que

$$\mathbf{x} = (\text{vec}(\mathbf{M})^\top, \mathbf{t}^\top)^\top. \quad (3.78)$$

Le symétrique de  $\mathbf{x}$  peut par définition de  $s_G$  être exprimé

$$s_G(\mathbf{x}) = (\text{vec}(\mathbf{M}\mathbf{G})^\top, \mathbf{t}^\top)^\top. \quad (3.79)$$

La projection de  $\mathbf{x}$  sur l'espace de pose consiste en la pose  $[\hat{\mathbf{R}}, \mathbf{t}]$ , où  $\hat{\mathbf{R}} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$  étant donné une décomposition en valeur singulières  $\mathbf{M}\mathbf{\Lambda} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$

suivant les mêmes conventions pour  $U, V, S$  et  $D$  que celles utilisées dans la section 3.7.3 où nous avons introduit ce résultat.

La projection de  $s_G(x)$  peut de manière similaire être déduite d'une décomposition en valeurs singulières de  $MGA$ . Nous savons grâce au lemme 6 qu'il est possible de réarranger ce dernier terme en  $MAG$ . Aussi, en injectant la décomposition en valeurs singulières  $UDV^\top$  dans cette dernière expression, il est possible de mettre en évidence une décomposition en valeur singulières de  $MGA$  :

$$\begin{aligned} MGA &= MAG \\ &= UDV^\top G \\ &= UD\tilde{V}^\top \end{aligned} \quad (3.80)$$

avec  $\tilde{V} = G^\top V \in O(3)$ . De plus,

$$\begin{aligned} \det(\tilde{V}) &= \det(G) \det(V) \\ &= \det(V) \end{aligned} \quad (3.81)$$

puisque  $G$  est une matrice de rotation et a donc un déterminant de 1. Aussi d'après le résultat de la section 3.7.3, la projection de  $s_G(x)$  est la pose

$$\begin{aligned} \text{proj}(s_G(x)) &= [U\tilde{S}\tilde{V}^\top, t] \\ &= [\hat{R}G, t]. \end{aligned} \quad (3.82)$$

Comme  $G$  est une symétrie propre de l'objet, celle-ci laisse par définition la pose inchangée c.-à-d.

$$[\hat{R}G, t] = [\hat{R}, t] \quad (3.83)$$

ce qui conclut cette démonstration.  $\square$

À partir de ces propriétés, il est maintenant possible de définir une estimation non ambiguë de la moyenne.

**Définition 14 (Estimation de la moyenne).** *Étant donné un  $n$ -uplet  $(\mathcal{P}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  cohérent de représentants des poses  $\{\mathcal{P}_i\}_{i=1\dots n}$ , on définit comme estimation de la moyenne de ces poses pondérées par  $\{w_i\}_{i=1\dots n}$*

$$\widehat{\text{moyenne}}((\mathcal{P}_i, w_i)_{i=1\dots n}) \triangleq \text{proj} \left( \frac{\sum_i w_i \mathcal{P}_i}{\sum_i w_i} \right). \quad (3.84)$$

Cette estimation correspond vraisemblablement à la moyenne exacte (3.64), cependant nous n'avons pas de preuves de cette conjecture.

**Démonstration.** Montrons néanmoins que cette notion est bien définie, c.-à-d. qu'elle ne dépend pas du  $n$ -uplet cohérent de représentants choisi. Soit  $(\mathcal{P}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  un  $n$ -uplet cohérent de représentants. L'ensemble des  $n$ -uplets cohérents de  $\prod_i \mathcal{R}(\mathcal{P}_i)$  correspond à l'ensemble des  $n$ -uplets symétriques à  $(\mathcal{P}_i)_{i=1\dots n}$  (proposition 12) :

$$\{(s(\mathcal{P}_i))_{i=1\dots n} \mid s \in G_{\mathcal{R}}\}. \quad (3.85)$$

Considérons donc un n-uplet cohérent arbitraire  $(s(\mathbf{p}_i))_{i=1\dots n}$ , avec  $s \in G_{\mathcal{R}}$  et montrons que ce dernier conduit à la même estimation  $\mathcal{M}$  de la moyenne que celle réalisée en considérant  $(\mathbf{p}_i)_{i=1\dots n}$ .

Par définition,

$$\mathcal{M} = \text{proj} \left( \frac{\sum_i w_i s(\mathbf{p}_i)}{\sum_i w_i} \right). \quad (3.86)$$

Du fait de la linéarité de la symétrie  $s$  (proposition 8), la moyenne arithmétique des  $(s(\mathbf{p}_i))_{i=1\dots n}$  correspond au symétrique de la moyenne arithmétique des  $(\mathbf{p}_i)_{i=1\dots n}$  :

$$\frac{\sum_i w_i s(\mathbf{p}_i)}{\sum_i w_i} = s \left( \frac{\sum_i w_i \mathbf{p}_i}{\sum_i w_i} \right), \quad (3.87)$$

d'où l'égalité

$$\mathcal{M} = \text{proj} \left( s \left( \frac{\sum_i w_i \mathbf{p}_i}{\sum_i w_i} \right) \right). \quad (3.88)$$

L'invariance de la projection aux symétries de l'espace ambiant (proposition 13) permet de conclure cette démonstration avec l'égalité

$$\mathcal{M} = \text{proj} \left( \frac{\sum_i w_i \mathbf{p}_i}{\sum_i w_i} \right). \quad (3.89)$$

□

### 3.8.3 Conditions suffisantes de cohérence d'un n-uplet

Nous avons montré comment la moyenne d'un ensemble de poses pouvait facilement être estimée à partir d'un n-uplet cohérent de représentants de celles-ci.

La notion de cohérence proposée n'est néanmoins pas triviale à établir, et il peut arriver qu'aucun n-uplet cohérent n'existe, comme c'est le cas dans la configuration illustrée figure 3.9abc. Il peut alors se révéler nécessaire pour calculer la moyenne de l'ensemble des poses d'évaluer de manière exhaustive la variance de Fréchet pour chaque n-uplet potentiel, afin de sélectionner celui produisant l'estimation de moyenne de variance minimale.

Heureusement, ce type de configuration présente un intérêt pratique limité, et nous présentons ici des conditions suffisantes simples garantissant la cohérence d'un n-uplet de représentants, et garantissant par là même la possibilité d'estimer facilement la moyenne de poses.

Nous montrons que la cohérence d'un n-uplet de représentants est en particulier satisfaite lorsque ceux-ci sont *suffisamment proches* les uns des autres, relativement à la distance  $T$  entre représentants d'une même pose (définition 4) :

**Proposition 15 (Représentants proches).** Soit  $(\mathbf{p}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  un  $n$ -uplet de représentants de poses. Si les éléments de  $(\mathbf{p}_i)_{i=1\dots n}$  sont distants entre eux de moins de  $T/2$ , c.-à-d. si

$$\forall (i, j) \in \llbracket 1, n \rrbracket^2, \|\mathbf{p}_i - \mathbf{p}_j\| < T/2, \quad (3.90)$$

alors ce  $n$ -uplet est cohérent.

**Démonstration.** Soit  $(\mathbf{p}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  un  $n$ -uplet satisfaisant la condition (3.90). Pour tout  $(i, j) \in \llbracket 1, n \rrbracket^2$  et  $\mathbf{q}_j \in \mathcal{R}(\mathcal{P}_j) \setminus \{\mathbf{p}_j\}$ , les inégalités suivantes sont vérifiées :

$$\begin{cases} \|\mathbf{q}_j - \mathbf{p}_j\| \leq \|\mathbf{p}_j - \mathbf{p}_i\| + \|\mathbf{q}_j - \mathbf{p}_i\| & \text{(inégalité triangulaire)} \\ \|\mathbf{q}_j - \mathbf{p}_j\| \geq T & \text{(définition 4)} \\ \|\mathbf{p}_j - \mathbf{p}_i\| < T/2. & \text{(condition (3.90))} \end{cases} \quad (3.91)$$

De ces inégalités, on déduit que

$$\|\mathbf{p}_j - \mathbf{p}_i\| < \|\mathbf{q}_j - \mathbf{p}_i\|, \quad (3.92)$$

d'où la cohérence de  $(\mathbf{p}_i)_{i=1\dots n}$ , d'après la définition de cette notion.  $\square$

Il est aisément possible d'obtenir une condition suffisante moins stricte en n'imposant seulement que les parties orientation des différents représentants de pose soient suffisamment proches les unes des autres. Il s'agit là d'une conséquence directe du fait que l'espace de pose peut être décomposé en le produit cartésien d'un espace de position et d'un espace d'orientation, et du fait que les symétries n'affectent que la partie orientation dans le cas d'un objet borné. La condition énoncée présente néanmoins l'avantage de s'abstraire de cette décomposition d'une pose en deux termes, aussi nous privilégions cette dernière. Il est également possible de considérer dans certains cas une borne supérieure plus grande que  $T/2$  selon la classe de symétrie de l'objet considérée, cependant il s'agit de la plus grande valeur satisfaisant à l'ensemble de celles-ci (proportionnellement à  $T$ ). Le lecteur est renvoyé à l'annexe C concernant ce point.

Un cas particulier d'intérêt pratique de ce critère est celui où l'ensemble des représentants considérés peuvent être inclus dans une boule de rayon suffisamment faible. Celui-ci est illustré figure 3.9d, et est exploité dans le cadre de notre application d'estimation de pose présentée section 3.10.

**Proposition 16 (Représentants inclus dans une boule).** Soit  $(\mathbf{p}_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  un  $n$ -uplet de représentants de pose. Si l'ensemble de ces représentants est inclus dans une boule de rayon  $T/4$ , c.-à-d. si

$$\exists \mathbf{c} \in \mathbb{R}^N / \forall i \in \llbracket 1, n \rrbracket^2, \|\mathbf{p}_i - \mathbf{c}\| < T/4, \quad (3.93)$$

alors ce  $n$ -uplet est cohérent.

**Démonstration.** Un  $n$ -uplet satisfaisant cette condition satisfait également celle de la proposition 15 d'après l'inégalité triangulaire, car pour tout

$$(i, j) \in \llbracket 1, n \rrbracket^2, \quad \begin{aligned} \|p_i - p_j\| &\leq \|p_i - c\| + \|p_j - c\| \\ &< T/4 + T/4. \end{aligned} \quad (3.94)$$

□

### 3.9 Propriétés locales de la distance proposée

Nous nous concentrons dans le cadre de nos travaux sur les propriétés globales de distances qui permettent de quantifier la similarité entre poses. Cependant, il est possible de démontrer que notre distance est localement équivalente à une métrique riemannienne définie sur la variété de l'espace de pose, aussi nous abordons brièvement dans cette section ces aspects locaux.

**Object présentant un groupe fini de symétrie propre** L'espace de pose d'un objet présentant un groupe de symétrie propre fini peut être vu comme une variété riemannienne de dimension 6 (3 dimensions liées à la position, 3 à l'orientation). En effet, considérons deux poses d'un tel objet, associées aux transformations rigides  $T_1$  et  $T_2 \in SE(3)$  choisies telles que

$$d([T_1], [T_2]) = d_{\text{no\_sym}}(T_1, T_2), \quad (3.95)$$

c.-à-d. telles que  $T_1^{-1} \circ T_2$  soit un plus petit déplacement de la pose  $[T_1]$  à la pose  $[T_2]$ . Pourvu que l'angle  $\theta$  de la rotation relative entre  $T_1$  et  $T_2$  soit suffisamment faible, le déplacement d'un point  $x \in \mathbb{R}^3$  de l'objet entre ces deux poses peut être approximé comme suit en introduisant le vecteur de déplacement  $v \in \mathbb{R}^3$  et le vecteur de rotation  $\omega \in \mathbb{R}^3$  entre  $T_1$  et  $T_2$  :

$$(T_1^{-1} \circ T_2)(x) \underset{\theta \rightarrow 0}{\sim} x + \omega \times x + v. \quad (3.96)$$

La distance entre les deux poses peut alors être approximée par

$$d([T_1], [T_2]) \underset{\theta \rightarrow 0}{\sim} \frac{1}{S} \int_S \mu(x) \|\omega \times x + v\|^2 ds. \quad (3.97)$$

En considérant un déplacement infinitésimal entre  $T_1$  et  $T_2$ , et en assimilant alors  $v$  et  $\omega$  à des vitesses respectivement translationnelles et angulaires, cette dernière expression correspond à la notion d'énergie cinétique (à un facteur  $2/S$  près). Il s'agit d'une forme quadratique, aussi notre distance est-elle localement équivalente à la distance riemannienne associée au tenseur métrique  $g$  suivant, défini pour toute paire de vecteurs tangents  $(v_1^\top, \omega_1^\top)^\top, (v_2^\top, \omega_2^\top)^\top \in \mathbb{R}^3 \times \mathbb{R}^3$  par

$$\begin{aligned} g\left((v_1^\top, \omega_1^\top)^\top, (v_2^\top, \omega_2^\top)^\top\right) \\ \triangleq \frac{1}{S} \int_S \mu(x) (\omega_1 \times x + v_1)^\top (\omega_2 \times x + v_2) ds. \end{aligned} \quad (3.98)$$

Cette métrique riemannienne a déjà été évoquée dans la littérature (Zefran et Kumar, 1996; Lin et Burdick, 2000), et Belta et Kumar (2002) suggèrent notamment l'usage de celle-ci à des fins d'interpolation sur  $SE(3)$ . Dans le cas où la covariance de l'objet est isotrope – c.-à-d.  $\Lambda = \lambda I$ , avec  $\lambda \in \mathbb{R}^{+*}$ , par exemple dans le cas d'un objet de géométrie sphérique ou cubique –, notre distance est de plus localement équivalente à la distance riemannienne usuelle (3.4) sur  $SE(3)$ .

**Autres classes d'objet** De manière similaire, l'espace de pose d'un objet de révolution peut être envisagé comme une variété à 5 dimensions. Notre distance est en effet pour un tel objet localement équivalente à une distance riemannienne sur  $S^2 \times \mathbb{R}^3$  induite pour la partie orientation par le plongement de  $S^2$  en tant que sphère d'un espace 3D euclidien, et en considérant la distance euclidienne usuelle pour la partie translation.

Les poses d'un objet 2D présentant un groupe de symétrie propre fini reposent de même sur une variété de dimension 3, et la distance proposée est localement équivalente à une distance riemannienne sur  $S^1 \times \mathbb{R}^2$  induite par le plongement de  $S^1$  en tant que cercle d'un espace euclidien à 2 dimensions, c.-à-d. en considérant pour distance entre deux orientations infiniment proches l'angle de la rotation relative entre les deux (à un facteur d'échelle près).

Enfin, les espaces de poses d'un objet 3D sphérique et d'un objet 2D circulaire sont respectivement équivalents à  $\mathbb{R}^3$  et  $\mathbb{R}^2$ , associés avec la distance euclidienne.

**Remarques** Malgré ces équivalences locales, les espaces de pose proposés pour les différentes classes d'objet peuvent néanmoins présenter une topologie différente des variétés évoquées plus haut, du fait des symétries discrètes des objets. À titre d'exemple, l'espace de pose d'un objet de révolution avec invariance par rotoréflexion n'est pas homéomorphe avec  $S^2 \times \mathbb{R}^3$ , mais plutôt avec  $\mathbb{R}P^2 \times \mathbb{R}^3$ , où  $\mathbb{R}P^2$  représente le plan réel projectif – c'est à dire une sphère dont les points antipodaux sont associés.

De plus, à l'exception des objets 3D sphériques ou 2D circulaires, notre distance est globalement distincte des distances géodésiques associées à ces métriques riemanniennes, ces dernières présentant certains inconvénients nous amenant à privilégier la distance proposée dans le cadre de notre visée applicative.

Ces distances géodésiques sont en effet plus coûteuses à estimer en ce qu'elles requièrent des calculs trigonométriques (p. ex. pour la distance (3.4)), et il n'existe pas de solution analytique connue pour la distance géodésique associée à la métrique (3.98) dans le cas d'un objet 3D arbitraire de groupe de symétrie propre fini. Celles-ci ne bénéficient de plus pas des sympathiques propriétés de calcul de notre distance notamment en ce qui concerne l'estimation de la moyenne de pose, pour laquelle elles peuvent nécessiter l'usage d'approches itératives (Pennec, 1998). Mais plus fondamentalement et ainsi qu'évoqué dans l'introduction, notre distance a pour but de quantifier la *similarité* entre deux poses d'un point de vue global, et la notion

de *mouvement* entre deux poses qui est exprimée au travers d'une distance géodésique n'est pas pertinente dans ce contexte.

### 3.10 Exemple applicatif

Dans cette section, nous illustrons l'intérêt de la distance proposée dans le cadre du problème de la détection et de l'estimation de pose d'instances d'objet, à partir d'un nuage de points. Afin d'illustrer la versatilité de notre approche, nous expérimentons avec trois objets présentant trois classes de symétrie distinctes parmi ceux illustrés tableau 3.2 :

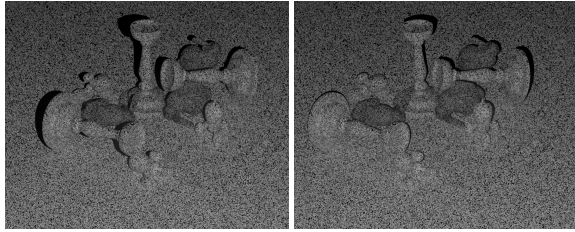
- le lapin de Stanford, un objet sans symétrie propre.
- un chandelier, assimilé à un objet de révolution sans invariance par rotoréflexion.
- une fusée stylisée, présentant une symétrie cyclique d'ordre 3, c.-à-d. une invariance par rotation de  $120^\circ$  autour d'un axe.

Pour des raisons pratiques, notre exemple s'appuie sur des données 3D synthétiques représentées figure 3.10b. Celles-ci ont été produites en simulant le fonctionnement d'un capteur stéréoscopique actif (figure 3.10a), suivant une approche décrite plus en détail section 5.1.2.2.

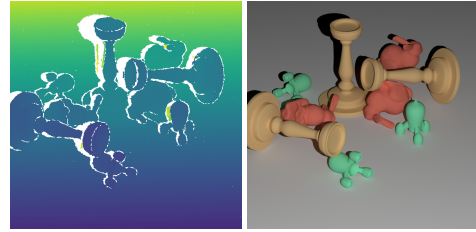
#### 3.10.1 Détection et estimation de poses via l'algorithme Mean Shift

Parmi les approches existantes prenant en entrée des données de profondeur, certaines approches (Drost et al., 2010; Fanelli et al., 2011; Tejani et al., 2014) procèdent suivant une approche « bottom-up » en produisant un ensemble de votes pour des poses candidates et, après avoir identifié ces votes à un échantillonnage d'une distribution, recherchent les modes principaux de celle-ci, qui correspondent idéalement aux poses des instances d'objet de la scène. Nous nous plaçons dans ce cadre de recherche de modes.

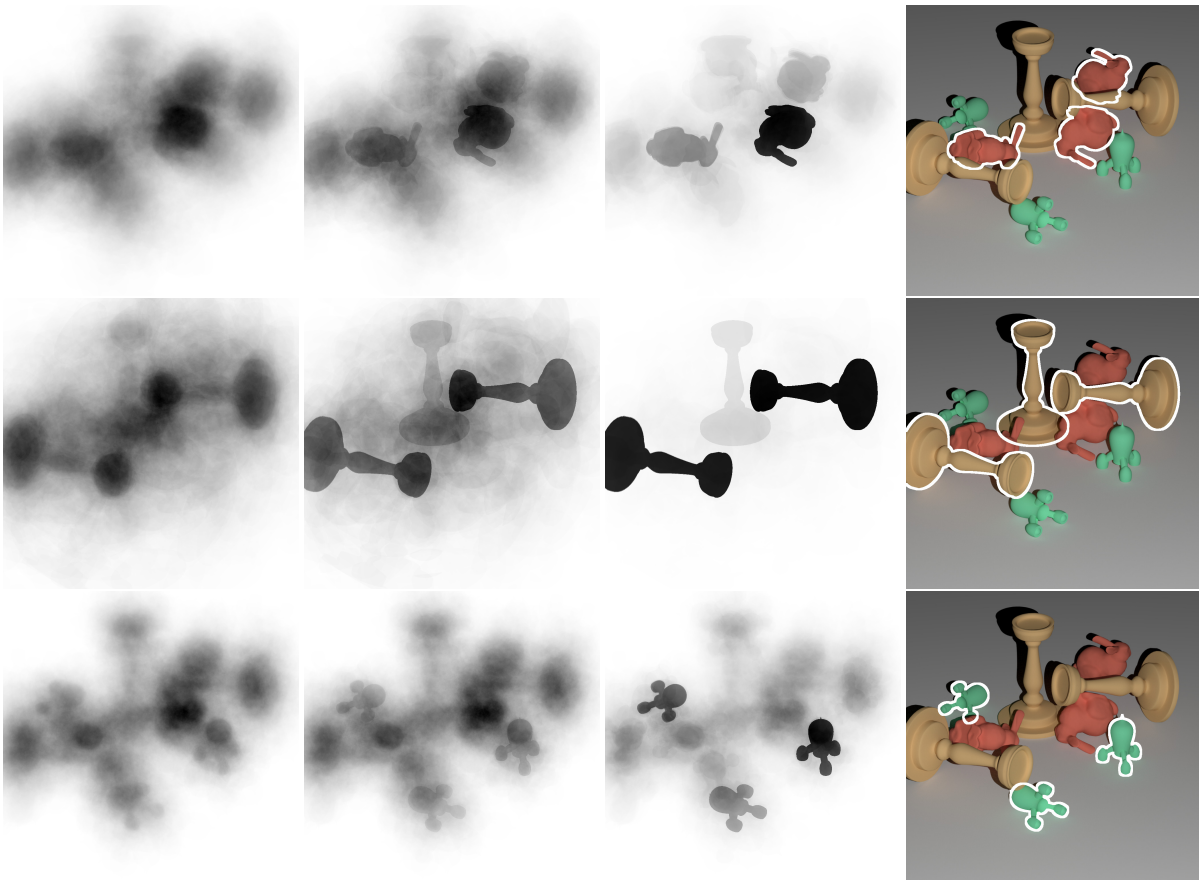
La détection efficace de modes sur l'espace de pose n'est pas un problème trivial. Les approches à base d'accumulateurs traditionnellement employées dans les méthodes de type Hough sont peu adaptées du fait de la dimension importante de l'espace de pose, à moins d'utiliser une structure parcimonieuse (Rodrigues et al., 2012) ou d'employer cette approche uniquement en tant que prétraitement sur quelques dimensions seulement (Drost et al., 2010; Rodrigues et al., 2012; Tejani et al., 2014). Une approche de détection de modes adaptée aux problèmes de dimensions importantes est *Mean Shift*, une méthode locale itérative et non paramétrique basée sur une estimation de densité par noyau. Cependant, cette dernière est conçue pour des espaces vectoriels, ce que n'est pas notre espace de pose. Fanelli et al. (2011) et Tejani et al. (2014) utilisent néanmoins Mean Shift avec une paramétrisation globale de l'espace de pose, mais une telle approche souffre des défauts intrinsèques évoqués dans l'introduction de ce chapitre. Tuzel et al. (2005) et plus tard Subbarao et Meer (2006) proposent des adaptations de Mean Shift adaptées au groupes de Lie et aux variétés riemanniennes qui pourraient permettre de dépasser ces difficultés, cependant leurs approches sont coûteuses en



(a) Paire stéréo utilisée pour la reconstruction 3D.



(b) Données 3D reconstruites (canal RGB uniquement à des fins de visualisation).



(c) De gauche à droite : distribution de pose sous forme de votes générée à partir des données 3D à l'aide de la méthode de [Drost et al. \(2010\)](#). Poses déplacées à l'aide de Mean Shift. Poses déplacées pondérées par la densité de la distribution initiale. Poses des instances d'objets détectées. Une distribution de pose est représentée en accumulant les silhouettes 2D de l'objet en ses différentes poses sur le plan image (plus un pixel appartient à de nombreuses silhouettes de poses, plus celui-ci est sombre). Les différents objets de la scène sont traités de manière indépendante.

FIGURE 3.10 – Exemple applicatif : détection d'instances d'objet et estimation de pose via une procédure de Mean Shift permettant d'extraire les modes principaux d'une distribution de pose initiale. Illustration avec trois objets présentant des propriétés de symétrie distinctes.



calcul car elles requièrent à chaque itération d'exprimer les échantillons sur le plan tangent du point courant, d'estimer dans le plan tangent le vecteur de déplacement à appliquer à celui-ci suivant la procédure de Mean Shift usuelle, et de retransformer ce dernier en un point sur la variété.

Dans cet exemple, nous montrons comment adapter l'algorithme Mean Shift de manière à localiser les modes d'une distribution sur l'espace de pose de manière efficace grâce à l'utilisation de notre distance, et cela même pour des objets présentant des symétries.

À partir de l'image de profondeur d'entrée, nous générons un ensemble de votes pour des poses d'objet  $\{\mathcal{P}_i\}_{i=1,\dots,n}$  au moyen de notre implémentation de la méthode de [Drost et al. \(2010\)](#) introduite section 2.5.3 de l'état de l'art. Nous utilisons un ratio d'échantillonnage du nuage de points de  $\tau_d = 0.025$  et considérons chaque échantillon comme point de référence (le lecteur intéressé par le sens de ces paramètres est renvoyé au papier de [Drost et al. \(2010\)](#)). La distribution de pose générée par cette méthode est relativement étalée ainsi que l'on peut l'observer par l'aspect flou de la représentation figure 3.10c, colonne 1.

Nous considérons alors en chacun de ces votes un point de départ de Mean Shift qu'il s'agira de déplacer vers un mode de densité. Une pratique usuelle permettant d'accélérer drastiquement les calculs lors de la recherche de modes consiste à ne considérer qu'un sous-ensemble des échantillons comme points de départ, par exemple en procédant par sous-échantillonnage aléatoire, mais nous n'utilisons pas ici une telle approche de manière à éviter l'introduction de paramètres additionnels. Pour chacune de ces poses à déplacer, nous procédons itérativement suivant la procédure usuelle de Mean Shift. Nous recherchons les poses parmi l'ensemble de votes dans un rayon  $r$  autour de la pose initiale, estimons leur moyenne, déplaçons la pose courante en cette pose moyenne et répétons la procédure jusqu'à convergence.

Nous fixons arbitrairement la valeur du rayon  $r$  à 1,5 fois la plus petite valeur propre de la matrice  $\Lambda$  pour les objets *lapin* et *chandelier* – ce qui correspond grossièrement à 75% de la plus petite dimension typique de l'objet. Nous utilisons un rayon plus faible correspondant à  $\sqrt{3}/2$  fois cette valeur propre pour l'objet *fusée*, qui correspond à la plus grande valeur satisfaisant la condition de la proposition 16 (voir annexe D). Il est possible qu'un rayon plus grand puisse produire expérimentalement de bons résultats, mais il ne bénéficierait pas des mêmes garanties théoriques. En effet, ces différents rayons pour les différents objets satisfont tous à la condition  $r < T/4$ , où  $T$  est la distance minimale entre représentants d'une même pose (voir définition 4). Nous avons établi section 3.5.1 qu'étant donné un représentant  $p$  d'une pose  $\mathcal{P}$  à déplacer via Mean Shift, les poses distantes de moins de  $r$  de la pose  $\mathcal{P}$  sont celles admettant un représentant distant de moins de  $r$  du représentant  $p$ . Ces représentants peuvent être identifiés de manière efficace au moyen d'une méthode classique de recherche de voisinage (*radius search*).  $r$  étant choisi strictement plus petit que  $T/2$ , les représentants ainsi obtenus par une telle requête correspondent nécessairement à des poses distinctes, sans doublons. Ces représentants étant de plus inclus dans une boule de rayon  $T/4$ , nous avons enfin l'assurance (proposition 16) qu'il est possible d'estimer de manière non ambiguë la moyenne des

poses correspondant à ces représentants par simple projection sur l'espace de pose de la moyenne arithmétique de ces représentants. Aussi, il est possible d'adapter la procédure de Mean Shift à l'espace de pose en n'ajoutant qu'une étape additionnelle comparée à la procédure usuelle dans un espace vectoriel. Cette étape consiste à projeter sur l'espace de pose la moyenne arithmétique des représentants retournés par la requête de voisinage et peut être réalisée suivant la méthode décrite section 3.7. L'algorithme 1 présente le pseudo-code de cette version adaptée de Mean Shift.

---

**Algorithme 1** Mean Shift sur l'espace de pose
 

---

**Entrée:**  $\{\mathcal{P}_i\}_{i=1,\dots,n}$  un ensemble de poses,  
 $\mathbf{p}_{in}$  un représentant de la pose à déplacer,  
 $r$  le rayon de Mean Shift.

**Sortie:** Un représentant de la pose déplacée.

- 1:  $R \leftarrow \text{GetRepresentatives}(\{\mathcal{P}_i\}_{i=1,\dots,n})$   
 # Prétraitement indépendant de  $\mathbf{p}_{in}$ .  
 #  $R$  contient l'ensemble des représentants de pose :  
 #  $\forall i \in \llbracket 1, n \rrbracket, \{R[i, j]\}_{j=1,\dots,|\mathcal{R}(\bullet)|} = \mathcal{R}(\mathcal{P}_i)$ .
- 2:  $\mathbf{p} \leftarrow \mathbf{p}_{in}$
- 3: **repeat**
- 4:    $\mathbf{p}_{old} \leftarrow \mathbf{p}$
- 5:    $\mathcal{N} \leftarrow \text{RadiusSearch}(R, \mathbf{p}, r)$   
 # Retourne l'ensemble des indices  $(i, j)$  des points de  $R$  dans un rayon  $r$  autour de  $\mathbf{p}$ .
- 6:   **if**  $\mathcal{N} \neq \emptyset$  **then**
- 7:      $\mathbf{m} \leftarrow \left( \sum_{(i,j) \in \mathcal{N}} R[i, j] \right) / |\mathcal{N}|$
- 8:      $\mathbf{p} \leftarrow \text{ClosestRepresentative}(\text{proj}(\mathbf{m}), \mathbf{p}_{old})$   
 # Retourne le représentant de  $\text{proj}(\mathbf{m})$  le plus proche de  $\mathbf{p}_{old}$ .
- 9:   **end if**
- 10: **until**  $\mathbf{p} \neq \mathbf{p}_{old}$
- 11: **return**  $\mathbf{p}$

---

La projection de la moyenne à chaque itération n'est en pratique pas nécessaire pour converger vers des modes pertinents, aussi nous ne réalisons celle-ci qu'une seule fois après convergence. Les poses déplacées par Mean Shift se concentrent au niveau des modes de la distribution initiale, et présentent donc une distribution plus *pointue* que la distribution originelle comme on peut l'observer figure 3.10c colonne 2 où les silhouettes des instances d'objet émergent.

Nous estimons alors la densité de la distribution en un mode  $\mathcal{M}$  au moyen d'un estimateur à noyau :

$$s(\mathcal{M}) = \sum_i H\left(\frac{d(\mathcal{M}, \mathcal{P}_i)}{r}\right) \quad (3.99)$$

où  $H$  représente le noyau d'Epanechnikov associé au noyau « haut de forme »  $\mathbb{1}_{[-1,1]}$  utilisé ici pour Mean Shift (Fukunaga et Hostetler, 1975) :

$$H(d) = \begin{cases} \frac{3}{4}(1 - d^2) & \text{if } |d| \leq 1 \\ 0 & \text{otherwise.} \end{cases} \quad (3.100)$$

Les modes les plus significatifs suivant ce critère peuvent alors être extraits. Ceux-ci sont supposés constituer de bonnes hypothèses pour les poses des instances d'objet de la scène, et se détachent typiquement de la distribution pondérée par cette densité (figure 3.10c colonne 3). Nous raffinons ces hypothèses par exemple suivant l'algorithme *Iterative Closest Point* (Besl et McKay, 1992), et les filtrons en vérifiant leur cohérence avec les données de manière à éviter les faux positifs, afin dans l'idéal de récupérer les poses des instances d'objets présentes dans la scène (figure 3.10c colonne 4).

**Limitations théoriques** L'interprétation en terme de densité de distribution faite ici est abusive et n'a pour but que de convoquer l'intuition de l'approche de Mean Shift. L'estimation de densité par noyau sur une variété riemannienne a notamment été étudiée par Pelletier (2005), mais notre approche ne rentre pas dans un tel cadre puisque nous ne considérons pas une métrique riemannienne, si ce n'est localement pour un rayon de Mean Shift infiniment petit (voir section 3.9). De telles considérations sortent cependant du cadre de ces travaux, et  $s(\mathcal{M})$  peut simplement être considéré comme un score exprimant le support des votes initiaux pour une pose  $\mathcal{M}$ .

### 3.10.2 Comparaison avec une métrique classique de $SE(3)$

Nous comparons ces résultats expérimentaux avec ceux obtenus avec une distance plus usuelle, adaptée à  $SE(3)$  :

$$d_{SE(3)}(T_1, T_2) = \sqrt{\|t_2 - t_1\|^2 + r^2\|R_2 - R_1\|^2}. \quad (3.101)$$

Nous choisissons cette distance particulière car l'algorithme Mean Shift nécessite de moyenniser des ensembles de poses, et une norme de Frobenius sur l'espace des rotations est bien adaptée à cette tâche (Curtis et al., 1993). Afin de limiter le biais de comparaison, nous choisissons pour valeur du facteur d'échelle  $r$  entre les termes de position et d'orientation

$$r = \sqrt{\frac{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}{3}}, \quad (3.102)$$

où  $\lambda_1 \leq \lambda_2 \leq \lambda_3$  représentent les valeurs propres de  $\Lambda$ . Ce choix est cohérent avec notre distance, en ce que le terme de rotation de la distance entre deux poses d'un objet sans symétrie propre correspond respectivement à  $2\sqrt{2}r \sin(\theta/2)$  pour la distance  $d_{SE(3)}$ , et  $2\sqrt{I_k} \sin(\theta/2)$  pour la distance proposée, où  $\theta$  est l'angle de la rotation relative entre les deux poses et  $I_k$  le moment d'inertie de l'axe  $k$  correspondant (voir section 3.4.4). Considérer un valeur typique de  $2/3(\lambda_1^2 + \lambda_2^2 + \lambda_3^2)$  pour  $I_k$  permet d'identifier ces deux termes.

Ainsi qu'illustré figure 3.11, nous observons peu de différences entre l'utilisation de ces deux distances pour l'objet *lapin*. Ce résultat n'est guère surprenant car les deux distances sont dans ce cas relativement semblables étant donné que le *lapin* ne présente pas de symétrie propre et possède une anisotropie limitée.

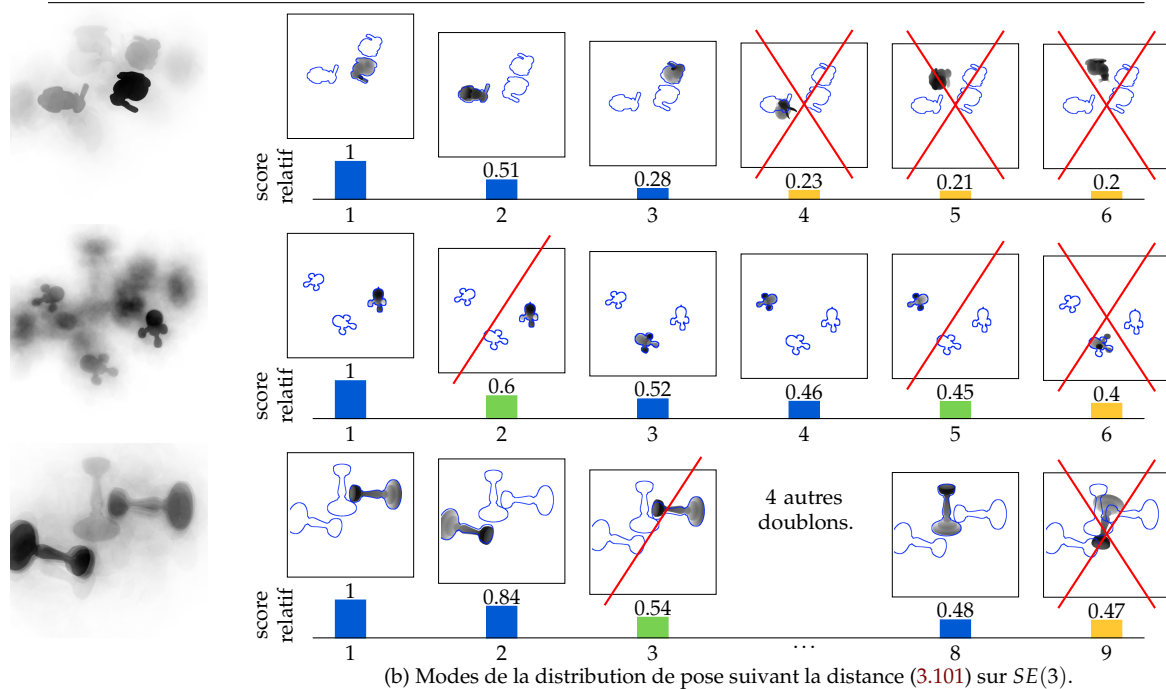
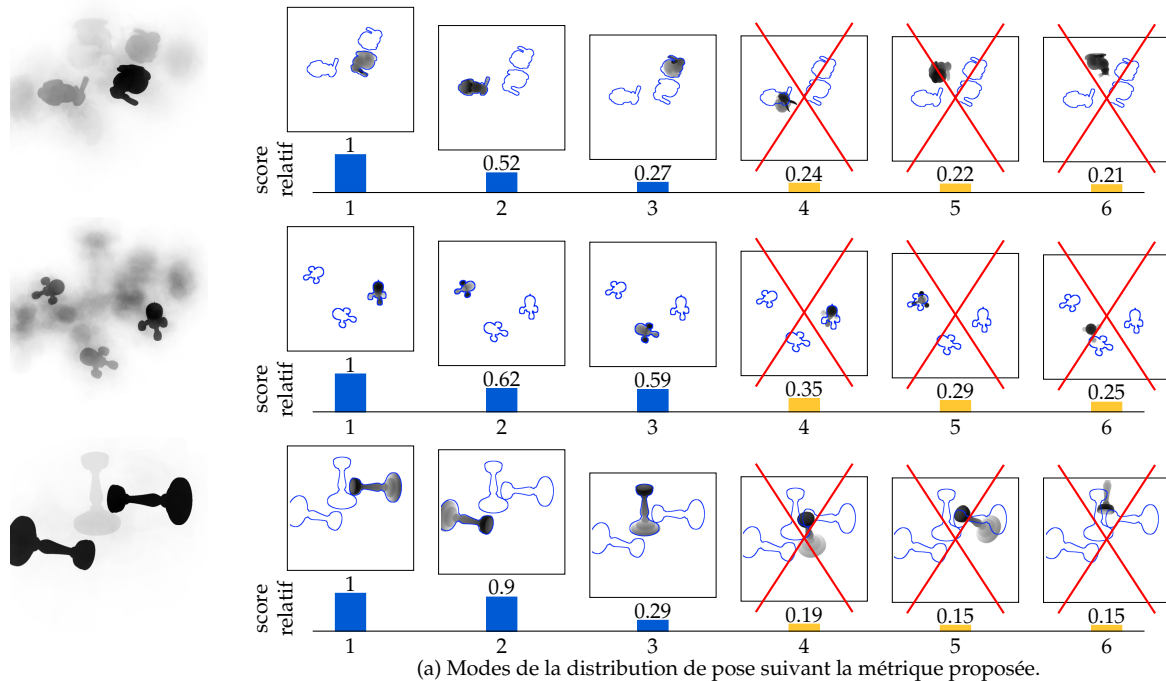


FIGURE 3.11 – Comparaison de l’utilisation de la distance proposée et d’une métrique de  $SE(3)$  à des fins d’estimation de pose (section 3.10.2). **À gauche** : distribution de pose après une opération de Mean Shift réalisée sur la distribution de votes initiale. **À droite** : modes principaux de la distribution de pose triés par scores décroissants (superposés avec les contours des silhouettes de la vérité terrain). Les modes principaux sont supposés constituer de bonnes hypothèses de poses, et sont ici classés en tant que vrais positifs (score en bleu), doublons (score en vert, simplement barré) et faux positifs (score en orange, doublement barré). Les deux distances produisent des résultats similaires pour l’objet *lapin* qui n’a pas de symétrie propre, et les premiers modes extraits correspondent aux poses des différentes instances. En revanche, la distance adaptée à  $SE(3)$  ne prend pas en compte les symétries des objets *fusée* et *chandelier* et conduit ainsi à la génération d’hypothèses de pose dupliquées, ce qui nécessite d’en considérer beaucoup avant de recouvrer la pose de chaque instance. La distance proposée exploite mieux l’information contenue dans l’ensemble de votes initial, générant des hypothèses de pose sans doublons et avec un écart relatif de score plus important entre les modes correspondant réellement à des instances d’objet et ceux infondés.

En revanche, l'intérêt de notre distance apparait avec les objets *fusée* et *chandelier*, qui présentent tous deux des symétries propres. Les votes initiaux pour des poses sont en effet étalés sur l'espace des transformations rigides  $SE(3)$  du fait des symétries, aussi la distance  $d_{SE(3)}$  conduit-elle à la détection de plusieurs modes correspondant à une même instance. Il est donc nécessaire de filtrer ces doublons avant toute application pratique, et cette redondance impose de plus de vérifier de nombreux modes si l'on souhaite trouver l'ensemble des instances de la scène. Dans notre exemple figure 3.11b, nous avons ainsi eu à tester respectivement jusqu'au quatrième et au huitième mode afin de recouvrir les poses des trois fusées et chandeliers présents dans la scène. A contrario, la distance proposée prend en compte les symétries propres de l'objet et permet ainsi de mieux exploiter l'information contenue dans l'ensemble initial de votes que la distance  $d_{SE(3)}$ . Dans notre exemple, les trois premiers modes extraits correspondent ainsi aux trois instances réellement présentes dans la scène, sans aucun doublon. De plus, ces modes disposent de d'avantage de support de la part de l'ensemble des votes et dès lors se distinguent plus clairement du bruit de fond, ce qui est important en terme de robustesse. La distribution de pose obtenue après Mean Shift est en effet visuellement plus marquée (figure 3.11, gauche), et les modes dus au bruit des objets *fusée* et *chandelier* présentent des scores valant moins de respectivement 59% et 66% de ceux des modes correspondant à de véritables instances, tandis que ce ratio n'est que de respectivement 89% et 98% dans le cadre de la distance  $d_{SE(3)}$ .

## 3.11 Synthèse

Il peut être difficile de discerner les résultats importants aux milieux des différents développements de ce chapitre. Aussi, nous synthétisons sous-section 3.11.1 les principales notions et résultats obtenus sous forme d'un guide pratique à destination de la personne ayant à manipuler la notion de pose d'un objet rigide, pour finalement conclure ce chapitre sous-section 3.11.2.

### 3.11.1 Résumé pratique des principaux résultats

#### Lien entre pose et transformation rigide (section 3.2.2)

Une pose  $\mathcal{P} \in \mathcal{C}$  d'un objet rigide peut être assimilée à une classe d'équivalence  $[T]$  de l'espace des transformations rigides :

$$\mathcal{P} = [T] \triangleq \{T \circ G, G \in G\}, \quad (3.103)$$

avec  $G \subset SE(3)$  le groupe des symétries propres de l'objet, spécifique à ce dernier mais pouvant être ramené à une des classes de symétrie illustrées tables 3.12 et 3.13, pourvu que l'objet soit physiquement admissible<sup>3</sup>.

3. De support borné et de groupe de symétrie propre fermé.

### Distance entre deux poses $\mathcal{P}_1, \mathcal{P}_2$ (section 3.4)

Nous définissons la distance entre deux poses comme la mesure du déplacement minimal entre celles-ci :

$$d(\mathcal{P}_1, \mathcal{P}_2) \triangleq \min_{G_1, G_2 \in G} d_{\text{no-sym}}(T_1 \circ G_1, T_2 \circ G_2). \quad (3.104)$$

La mesure d'un déplacement est pour se faire définie comme la moyenne RMS des déplacements de l'ensemble  $\mathcal{S}$  des points de l'objet (typiquement sa surface) :

$$d_{\text{no-sym}}(T_1, T_2) \triangleq \sqrt{\frac{1}{S} \int_{\mathcal{S}} \mu(x) \|T_2(x) - T_1(x)\|^2 ds}, \quad (3.105)$$

avec  $\mu$  une fonction de densité et  $S \triangleq \int_{\mathcal{S}} \mu(x) ds$ .

### Représentants de pose (section 3.5)

Afin de permettre des calculs efficaces, nous introduisons la notion de *représentants de pose*, qui à une pose  $\mathcal{P}$  identifie un ensemble fini de points  $\mathcal{R}(\mathcal{P}) \subset \mathbb{R}^N$  d'un espace à au plus 12 dimensions, selon les symétries de l'objet (cf. tables 3.12 et 3.13). Les représentants permettent de mettre en évidence le caractère *quasi-euclidien* de notre distance, via l'égalité suivante :

$$\forall p_1 \in \mathcal{R}(\mathcal{P}_1), d(\mathcal{P}_1, \mathcal{P}_2) = \min_{p_2 \in \mathcal{R}(\mathcal{P}_2)} \|p_2 - p_1\|. \quad (3.106)$$

Cette propriété apporte des solutions pratiques pour l'estimation de la distance entre poses ou la recherche de voisinage, en rendant possible l'utilisation de structures de recherche conçues pour les espaces euclidiens (grille, kD-tree, etc.).

### Projection sur l'espace de pose (section 3.7)

Nous définissons alors un opérateur permettant de projeter un point  $x \in \mathbb{R}^N$  de l'espace ambiant sur l'espace de pose

$$\text{proj}(x) \triangleq \underset{\mathcal{P}}{\text{argmin}} \min_{p \in \mathcal{R}(\mathcal{P})} \|p - x\|^2, \quad (3.107)$$

et proposons des expressions analytiques pour ce dernier, selon la classe de symétrie de l'objet (cf. tables 3.12 et 3.13).

### Moyenne de poses (section 3.8)

À l'aide de cet opérateur, nous montrons qu'il est possible d'estimer la moyenne d'un ensemble de poses  $\{\mathcal{P}_i\}_{i=1\dots n}$  pondérées par des coefficients positifs  $\{w_i\}_{i=1\dots n}$  de manière simple, dès lors que l'on dispose d'un n-uplet  $(p_i)_{i=1\dots n} \in \prod_i \mathcal{R}(\mathcal{P}_i)$  de représentants des poses  $\{\mathcal{P}_i\}_{i=1\dots n}$  satisfaisant une propriété particulière de *cohérence*.

Cette propriété est en particulier satisfaite lorsque les  $(\mathbf{p}_i)_{i=1\dots n}$  sont distants deux à deux de moins de  $T/2$ , où  $T = \min_{\mathbf{p}, \mathbf{q} \in \mathcal{R}(\mathcal{P}), \mathbf{p} \neq \mathbf{q}} \|\mathbf{p} - \mathbf{q}\|$  représente la distance minimale entre deux représentants d'une même pose  $\mathcal{P}$ . En ce cas, la moyenne peut alors être estimée comme suit :

$$\widehat{\text{moyenne}}((\mathcal{P}_i, w_i)_{i=1\dots n}) \triangleq \text{proj} \left( \frac{\sum_i w_i \mathbf{p}_i}{\sum_i w_i} \right). \quad (3.108)$$

### Expressions pour les différentes classes de symétrie

Les tables 3.12 et 3.13 synthétisent les expressions des différentes notions introduites pour l'ensemble des classes de symétries potentielles d'un objet rigide 3D ou 2D.

Conventions utilisées :

- Centre de masse de l'objet choisi comme origine du repère objet.
- Axe de révolution choisi comme axe  $\mathbf{e}_z$  du repère objet dans le cas d'un objet de révolution.

Notations pour un objet 3D :

$$\mathbf{\Lambda} \triangleq \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \mathbf{x} \mathbf{x}^\top ds \right)^{1/2}.$$

$$\lambda \triangleq \sqrt{\lambda_r^2 + \lambda_z^2} \text{ pour les objets de révolution, où } \mathbf{\Lambda} = \text{diag}(\lambda_r, \lambda_r, \lambda_z).$$

Notations pour un objet 2D :

$$\lambda \triangleq \left( \frac{1}{S} \int_S \mu(\mathbf{x}) \|\mathbf{x}\|^2 ds \right)^{1/2}.$$

$$\forall \alpha \in \mathbb{R}, \mathbf{R}^\alpha \triangleq \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}, \text{ et } e^{i\alpha} \triangleq (\cos(\alpha), \sin(\alpha)).$$

TABLE 3.12 – Synthèse des différentes expressions proposées dans le cas d’un objet rigide 3D (section 3.11.1).









Classe de symétrie	Groupe de symétrie propre $G$	Représentants de pose $\mathcal{R}(\mathcal{P}) \subset \mathbb{R}^N$	Projection $[(R, t)]$ d’un point $x \in \mathbb{R}^N$ sur l’espace de pose
 Sphérique	$SO(3)$	$t \in \mathbb{R}^3$	$t = x.$
 Révolution sans invariance par rotoreflexion	$\{R_z^\alpha   \alpha \in \mathbb{R}\}$	$(\lambda(Re_z)^\top, t)^\top \in \mathbb{R}^6$	$R$ une rotation d’axe $x_r / \ x_r\ $ , $t = x_t$ , avec $x_r, x_t \in \mathbb{R}^3$ tels que $x = (x_r^\top, x_t^\top)^\top$ .
 Révolution avec invariance par rotoreflexion	$\{R_x^\delta R_z^\alpha   \delta \in \{0, \pi\}, \alpha \in \mathbb{R}\}$	$\{(\pm\lambda(Re_z)^\top, t)^\top\} \subset \mathbb{R}^6$	
 Sans symétrie propre	$\{I\}$	$(\text{vec}(R\Lambda)^\top, t)^\top \in \mathbb{R}^{12}$	$R = USV^\top, t = x_t$ , où $UDV^\top$ est une décomposition en valeurs singulières (ordonnées de manière décroissante) de $X_r\Lambda$ , avec $X_r \in \mathcal{M}_{3,3}(\mathbb{R})$ et $x_t \in \mathbb{R}^3$ tels que $x = (\text{vec}(X_r)^\top, x_t^\top)^\top$ , et où $S$ est définie par $S = \begin{cases} I_{3 \times 3} & \text{si } \det(U) \det(V) > 0 \\ \text{diag}(1, 1, -1) & \text{sinon.} \end{cases}$
 Fini non trivial	Fini	$\{(\text{vec}(RGA)^\top, t)^\top   G \in G\} \subset \mathbb{R}^{12}$	



TABLE 3.13 – Synthèse des différentes expressions proposées dans le cas d'un objet rigide 2D (section 3.11.1).

Classe de symétrie	Groupe de symétrie propre $G$	Représentants de pose $\mathcal{R}(\mathcal{P}) \subset \mathbb{R}^N$	Projection $[(R, t)]$ d'un point $x \in \mathbb{R}^N$ sur l'espace de pose
 Circulaire	$SO(2)$	$t \in \mathbb{R}^2$	$t = x$ .
 Sans symétrie propre	$\{I\}$	$(\lambda e^{i\theta}, t^\top)^\top \in \mathbb{R}^4$	$R = R^{\arg(x_r)}$ , $t = x_t$ , avec $x_r, x_t \in \mathbb{R}^2$ tels que $x = (x_r^\top, x_t^\top)^\top$ et $\arg(x_r)$ l'argument de $x_r$ vu comme nombre complexe.
 Cyclique (d'ordre $n \in \mathbb{N}^*$ )	$\{R^{2k\pi/n} \mid k \in \llbracket 0, n \rrbracket\}$	$\{(\lambda e^{i(\theta+2k\pi/n)}, t^\top)^\top \mid k \in \llbracket 0, n \rrbracket\} \subset \mathbb{R}^4$	

### 3.11.2 Conclusion

Au travers des développements théoriques présentés dans ce chapitre, nous avons tâché de résoudre les difficultés liées à la notion de pose d'un objet rigide, à la fois dans les cas 2D et 3D.

Bien qu'une pose soit ordinairement supposée équivalente à une transformation rigide, cette hypothèse n'est pas adaptée dans le cas général du fait des symétries potentielles de l'objet. Nous proposons donc une définition plus générale de la notion de pose et qui consiste en un état statique distinguable de l'objet. Nous montrons qu'avec cette définition, une pose peut être identifiée à une classe d'équivalence de l'espace des transformations rigides, grâce à l'introduction du groupe de symétrie propre spécifique à l'objet. Nous pensons que cette notion est loin d'être anecdotique, car de nombreux objets manufacturés présentent des propriétés de symétrie et ne peuvent être représentés convenablement sans celle-ci.

En s'appuyant sur cette définition, nous proposons une distance permettant de quantifier la similarité entre poses. Cette distance consiste en une mesure du plus petit déplacement entre deux poses, la longueur d'un déplacement correspondant à la moyenne RMS des distances de déplacement des points à la surface de l'objet. Hormis le fait que cette distance soit définie pour tout objet physique rigide, celle-ci présente l'intérêt de prendre en compte la géométrie de l'objet, sans dépendre de choix arbitraires de repères ou de facteurs d'échelle.

Dans un souci d'efficacité de calcul, nous proposons un cadre cohérent permettant de représenter les poses dans un espace euclidien de dimension au plus 12 afin d'estimer des distances, de réaliser des requêtes de voisinage ou encore de moyenner des poses, tout en fournissant des preuves théoriques de ces résultats.

Ces développements permettent d'envisager l'usage de notre distance pour des tâches de haut niveau telles que la détection et l'estimation de pose à partir d'un ensemble de votes, où elle s'avère plus adaptée qu'une métrique destinée à  $SE(3)$ .

# Chapitre 4

## Solution proposée

*S'il n'y a pas de solution, c'est qu'il n'y a pas de problème.*

– Jacques Rouxel, *Les Shadoks*.

---

4.1	Régression probabiliste locale . . . . .	105
4.2	Forêt de Hough . . . . .	111
4.3	Extraction d'hypothèses de pose . . . . .	120
4.4	Raffinement d'hypothèses . . . . .	120
4.5	Vérification et filtrage . . . . .	123
4.6	Apprentissage d'un arbre de décision . . . . .	128
4.7	Synthèse . . . . .	140

---

Forts des outils introduits dans le chapitre précédent, nous développons une méthode de détection et d'estimation de pose d'instances d'objet rigide à partir d'une image de profondeur, que nous présentons dans ce chapitre.

**Vue générale** Lors de l'exécution, cette approche peut être décomposée en différentes phases, dont une illustration est proposée figure 4.1.

Nous cherchons à générer des hypothèses de pose pertinentes au vu des données d'entrée. Nous nous plaçons pour cela dans un cadre probabiliste, et estimons une distribution représentant la probabilité de présence d'une instance d'objet en chaque pose, étant donné l'observation des données d'entrée. Cette estimation est difficile du fait de la variabilité des scènes et des données qu'il est possible de rencontrer. Nous abordons donc ce problème à partir d'une approche locale simplifiée qui consiste à estimer la densité de probabilité de pose d'une instance d'objet, étant donné un voisinage local centré autour d'un point de référence lui appartenant. Cette estimation est réalisée au moyen d'une forêt de décision, apprise de manière hors-ligne au moyen d'images synthétiques idéales.

En considérant un ensemble de points de référence sélectionnés à partir des données d'entrée, on dispose alors d'un ensemble d'estimateurs locaux qu'il est possible de combiner pour produire un estimateur prenant en compte l'entière des données. On extrait les modes de la distribution de probabilité

inférée grâce à ce dernier, qui correspondent à des hypothèses pertinentes concernant la pose d'instances présentes dans la scène. Ces hypothèses sont ensuite potentiellement raffinées de manière itérative, pour être finalement filtrées et retournées comme résultat.

Les différentes sections de ce chapitre décrivent plus en détail l'ensemble de ces différentes étapes.

Au sein de la littérature, les travaux de [Tejani et al. \(2014\)](#) sont ceux se rapprochant le plus de notre approche, développée indépendamment. La spécificité fondamentale de nos travaux réside en l'usage d'une représentation de pose théoriquement fondée, ne souffrant pas des problèmes évoqués chapitre 3 et à même de prendre en compte tout type d'objet rigide, y compris ceux symétriques.

**Visualisation d'une distribution de pose** La notion de distribution sur l'espace des poses d'objet est fondamentale à notre approche, cependant une telle distribution n'est pas triviale à visualiser, du fait de la topologie particulière de l'espace de pose et de sa dimension trop grande pour permettre des représentations sur papier ou en 3D (se référer au chapitre 3 où nous définissons cet espace). Pour pallier à cette limitation, nous représentons une distribution sous forme d'une image en niveau de gris, telle que celles illustrées figure 4.1, produites comme suit :

Une distribution est représentée dans notre implémentation de manière parcimonieuse, sous forme d'une collection d'échantillons de poses. Pour la visualiser, on réalise un rendu de la silhouette de l'objet en chacune de ces poses, que l'on accumule dans une image. Plus le nombre de silhouettes accumulant pour un pixel donné est élevé, plus celui-ci sera sombre.

L'image finale constitue ainsi en quelque sorte une projection de la distribution sur le plan image, et permet d'intuiter certaines propriétés de la distribution. Une distribution étalée produira ainsi une image floue, alors qu'une distribution concentrée autour de modes bien définis laissera apparaître la silhouette des poses correspondantes à ces derniers. C'est ce dernier type de distribution, discriminant bien les poses vraisemblables des autres, que nous cherchons à produire.

## 4.1 Régression probabiliste locale

La détection et la localisation d'un nombre inconnu d'instances est un problème délicat à formuler. [Barinova et al. \(2012\)](#) ont proposé un intéressant formalisme probabiliste global où la détection 2D d'un ensemble d'instances d'objet dans des images est ramené à un problème d'optimisation de type *Maximum A Posteriori*. Leur approche dépend cependant fondamentalement de la discrétisation de l'espace de paramètres des instances (p. ex. coordonnées 2D dans l'image), et il s'agit d'une limitation qui semble non triviale à dépasser. Dans notre cas d'application, l'espace de paramètres (c.-à-d. l'espace de pose) des instances d'objet est de dimension relativement importante, ce qui rend sa discrétisation difficilement envisageable avec une résolution suffisante pour une localisation précise.

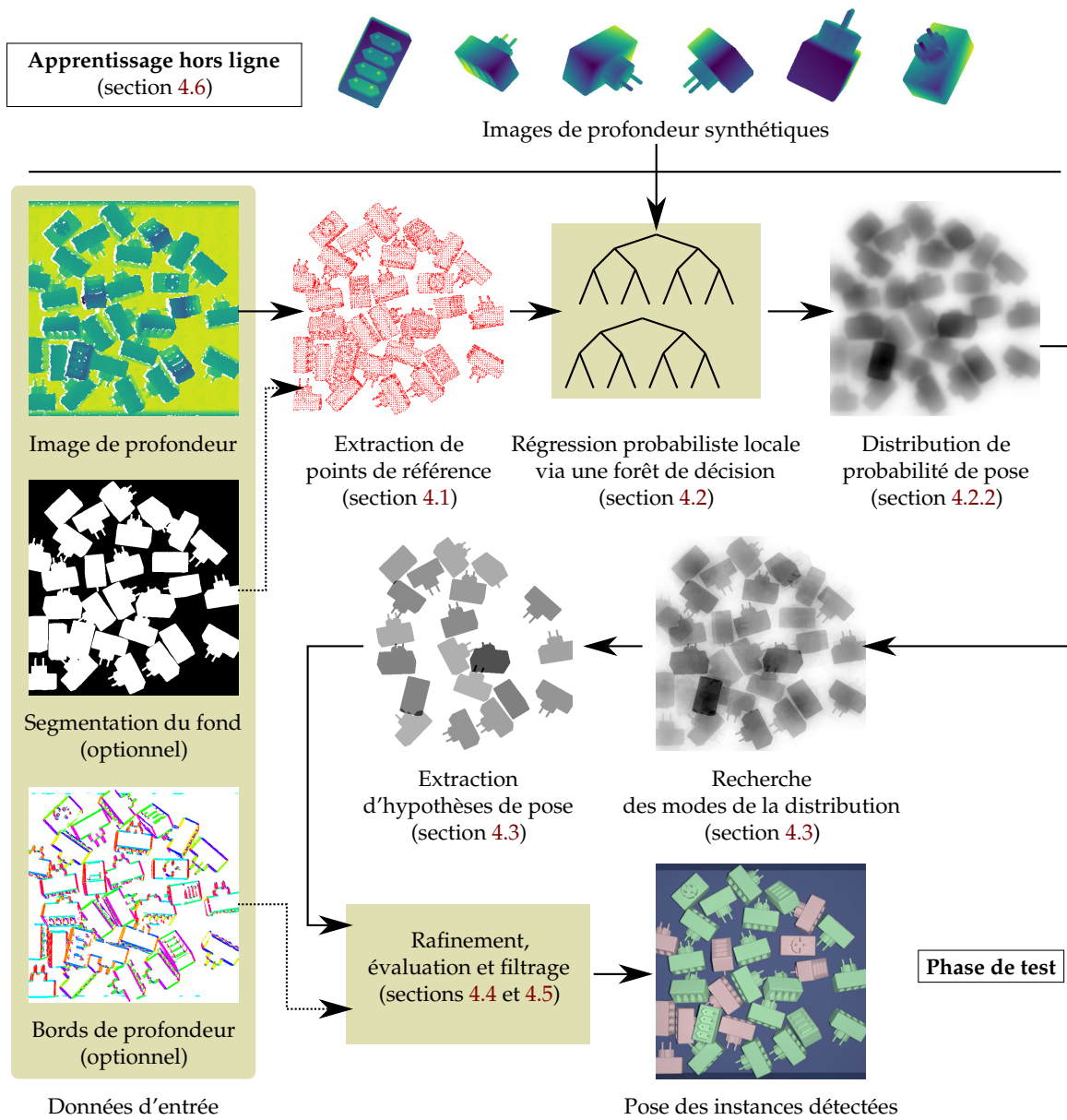


FIGURE 4.1 – Schéma de notre méthode de détection et d'estimation de pose.

Aussi nous abandonnons l'idée d'un formalisme global cohérent et nous nous rapportons à un problème local, mieux posé. Étant donné un point de référence choisi dans l'image de profondeur, il s'agit de déterminer :

1. si le point appartient ou non à une instance d'objet : il s'agit là d'un problème de *classification*.
2. le cas échéant, la pose de l'instance à laquelle appartient le point de référence. C'est là un problème de *régression*, que nous abordons de manière probabiliste.

En fusionnant les solutions de ces problèmes locaux, il nous sera possible d'exprimer en chaque pose  $\mathcal{P} \in \mathcal{C}$  la *probabilité a posteriori*  $f(\mathcal{P})$  qu'il existe une instance d'objet de la scène au voisinage de cette pose. Les maxima locaux de cette distribution  $f$  pourront alors légitimement être considérés comme des hypothèses pertinentes concernant la pose d'instances réellement présentes dans la scène.

Il est important de préciser ici que  $f$  ne représente alors pas une fonction de densité de probabilité – c.-à-d. n'est notamment pas nécessairement d'intégrale 1 sur l'espace de pose –, car celle-ci représente la pose non pas d'une mais d'un nombre arbitraire d'instances d'objet. De fait l'ensemble des hypothèses ainsi produites ne constituent aucunement l'interprétation globale la plus probable de la scène, mais simplement un ensemble des hypothèses de pose les plus probables comparé à leurs voisinages, indépendamment les unes des autres.

#### 4.1.1 Points de référence

**Classification d'un point de référence** Notre scénario typique d'application consiste en des scènes d'instances d'objet en vrac au sein d'un environnement industriel relativement contrôlé. Dès lors, il est relativement aisé de segmenter les instances d'objet dans une image de profondeur, par exemple en comparant ces données avec un modèle du conteneur dans lequel sont stockées les instances d'objet. Nous supposons donc que cette segmentation est fournie en entrée de notre méthode, sous forme d'un masque binaire tel que celui figure 4.1. Déterminer si un point de référence appartient ou non à une instance d'objet revient alors à une simple lecture de la valeur du masque en les coordonnées du point de référence, aussi nous ne nous attardons pas d'avantage sur ce point.

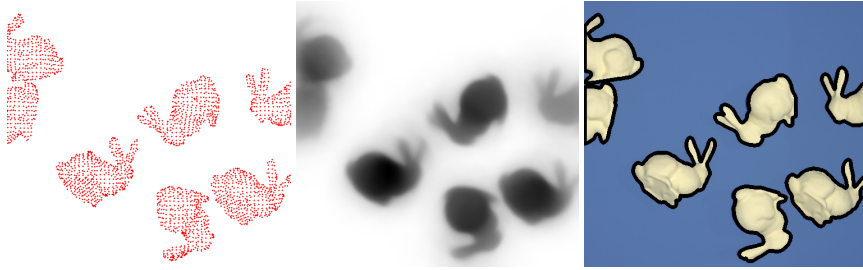
En pratique, notre approche n'est de plus pas trop dépendante de cette segmentation et il est possible de supposer que chaque point de référence appartient à une instance d'objet, au prix de l'ajout de bruit dans la distribution de probabilité de pose estimée lorsque ce n'est pas le cas, ainsi que l'illustre la figure 4.2.

**Choix des points de référence** Bien qu'il soit envisageable de considérer l'ensemble des pixels de la carte de profondeur comme points de référence, il est préférable de limiter le nombre de ces derniers pour des raisons de temps de calcul.

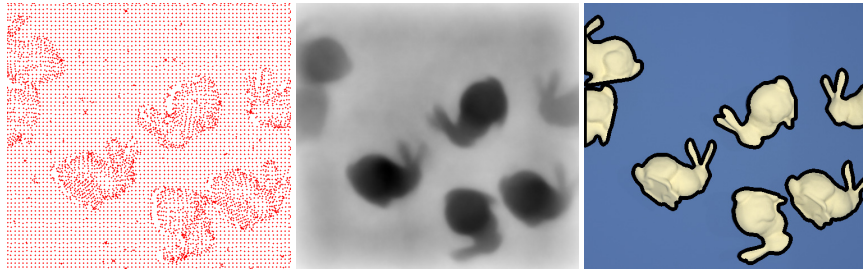
Dans une première approche, nous avons testé l'utilisation de points de référence correspondant à des caractéristiques saillantes de l'image de profondeur, de manière à pouvoir sélectionner ceux-ci de manière relativement



(a) Données d'entrée : image de profondeur, masque de segmentation de l'objet (optionnel), et image RGB pour visualisation.

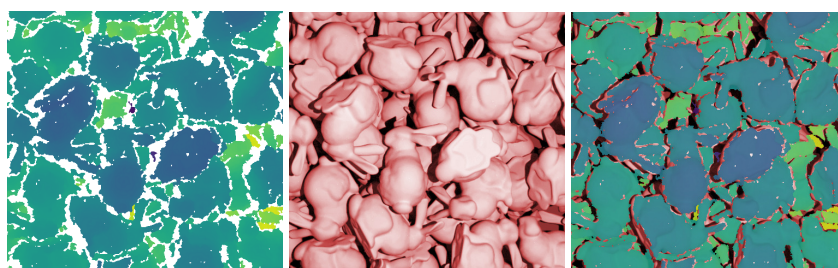


(b) Avec masque de segmentation.



(c) Sans masque de segmentation.

FIGURE 4.2 – Segmentation facultative du fond. (b, c) Points de référence considérés (à gauche) et distribution de probabilité de pose (au centre) estimée à partir de l'image de profondeur d'entrée, (b) en exploitant une segmentation préalable des instances ou (c) en supposant à tort que chaque point de référence extrait de l'image de profondeur appartient à une instance d'objet. Ce dernier cas conduit à la production d'une distribution de probabilité d'avantage bruitée mais qui reste néanmoins exploitable pour localiser des instances d'objet présentes dans la scène (à droite).



(a) Profondeur. (b) Zones occultées pour la seconde caméra. (c) Superposition des deux images.

FIGURE 4.3 – Origine de l’absence d’information dans une image de profondeur. Dans le cas d’un capteur stéréoscopique, une part importante des pixels sans information de profondeur (en blanc figure (a)) est due à l’occultation des points correspondants selon le point de vue de la seconde caméra avec laquelle est réalisé l’appariement stéréo. Ces zones d’occultation sont équivalentes aux zones d’ombres portées produites par une source ponctuelle qui serait placée au niveau de cette caméra (zones d’ombres figure (b)), et sont particulièrement fréquentes au niveau du contour des instances, du fait du saut de profondeur qui y est rencontré.

répétable, ce qui aurait simplifié la régression de pose. Les caractéristiques saillantes dans une image de profondeur consistent typiquement en des bords de profondeur (discontinuités) ou encore des zones de rupture de pente (discontinuité de la pente), cependant nous avons obtenu des résultats mitigés en exploitant celles-ci. En effet, les images de profondeur que nous exploitons ne sont pas denses, mais présentent des zones sans information. Dans le cas d’un capteur par triangulation, cette absence d’information se rencontre particulièrement au niveau des bords de profondeurs et est alors due à l’occultation de ces zones pour le point de vue utilisé pour la triangulation, comme illustré figure 4.3. L’information de profondeur lorsqu’elle y est malgré tout disponible est souvent peu fiable, car elle est alors d’avantage le fruit de la régularisation introduite par les algorithmes de reconstruction 3D que des données elle-mêmes. Aussi la position des caractéristiques saillantes se retrouve elle-même relativement bruitée, ce qui conduit à une répétabilité d’extraction de points d’intérêts médiocre.

Nous avons donc opté pour une approche plus simple, consistant à répartir de manière régulière les points de référence. L’ensemble des données de profondeur, à l’exception des pixels ne correspondant pas à des instances d’objet selon le masque de segmentation fourni en entrée, est vu comme un nuage de points que l’on répartit suivant une grille régulière de voxels, de dimension liée aux dimensions de l’objet<sup>1</sup>. Le centroïde de chaque voxel non vide est alors estimé et utilisé comme point de référence. La figure 4.2 illustre un exemple de ces points de référence considérés.

1. Le pas de la grille est typiquement choisi de manière à avoir un nombre de points de référence fixé (75) étant donné la surface efficace de l’objet (produit largeur  $\times$  hauteur).



### 4.1.2 Formulation probabiliste

Pour chaque point de référence  $x_i$  extrait de l'image (avec  $i = 1 \dots n$ , et  $n \in \mathbb{N}$ ), un estimateur dont on note la sortie  $o_i$  analyse un voisinage local autour de  $x_i$  afin d'inférer la pose de l'instance d'objet à laquelle ce point appartient. Cet estimateur est mis en œuvre au moyen d'une forêt de décision, et on note  $\mathcal{P} \mapsto f(\mathcal{P}|o_i)$  la densité de probabilité conditionnelle produite par ce dernier.

Notre objectif consiste alors à fusionner ces différentes inférences en une unique estimation  $\mathcal{P} \mapsto f(\mathcal{P})$ , prédisant en chaque pose la probabilité a posteriori de la présence d'une instance d'objet.

Étant donné une pose  $\mathcal{P} \in \mathcal{C}$ , on note  $\omega_{\mathcal{P}}$  l'évènement correspondant au fait qu'il existe une instance d'objet de la scène au voisinage de cette pose, c.-à-d.

$$\omega_{\mathcal{P}} \triangleq \left( \exists \mathcal{P}_0 \in S / \mathcal{P}_0 \in N_{d\mu}(\mathcal{P}) \right), \quad (4.1)$$

où  $S \in \cup_{k \in \mathbb{N}} \mathcal{C}^k$  représente l'ensemble des poses des instances d'objet présentes dans la scène et  $N_{d\mu}(\mathcal{P})$  un voisinage local de  $\mathcal{P}$  de mesure infinitésimale  $d\mu$ . La probabilité a posteriori de l'évènement  $\omega_{\mathcal{P}}$  que l'on souhaite estimer s'exprime alors une fois normalisée

$$f(\mathcal{P}) \triangleq \frac{1}{d\mu} \Pr[\omega_{\mathcal{P}} | o_1, \dots, o_n]. \quad (4.2)$$

Nous utilisons à dessein dans l'ensemble de cette section cette notion d'un volume infinitésimal, afin d'éviter toute confusion de  $f$  avec une fonction de densité de probabilité.

**Hypothèses simplificatrices** Afin de mener à bien la fusion des différentes inférences locales, nous posons les hypothèses simplificatrices suivantes :

- Les sorties  $(o_i)_{i=1 \dots n}$  de l'estimateur local sont indépendantes conditionnellement à  $\omega_{\mathcal{P}}$ .
- Pour  $i = 1 \dots n$ , la sortie  $o_i$  de l'estimateur local ne fournit d'information a posteriori pertinente que pour l'instance d'objet à laquelle le point de référence  $x_i$  appartient. Les prédictions de ce dernier constituent donc des valeurs aberrantes en ce qui concerne la pose d'autres instances, ce à quoi il faut ajouter le cas où  $x_i$  n'appartiendrait pas réellement à une instance d'objet – en cas d'erreur dans le calcul du masque de segmentation d'objet notamment, voire en l'absence de ce dernier. Afin d'introduire une certaine robustesse à ces prédictions aberrantes, nous modélisons la probabilité a posteriori de l'évènement  $\omega_{\mathcal{P}}$  avec le modèle simple suivant, dans lequel la probabilité  $\epsilon$  d'une prédiction pertinente est supposée très faible ( $\epsilon \ll 1$ ) :

$$\Pr[\omega_{\mathcal{P}} | o_i] = (1 - \epsilon) \Pr[\omega_{\mathcal{P}}] + \epsilon f(\mathcal{P} | o_i) d\mu. \quad (4.3)$$

**Développements** Nous développons alors l'expression de  $f(\mathcal{P})$ . D'après le théorème de Bayes, celle-ci peut être réécrite

$$\frac{1}{d\mu} \frac{\Pr[o_1, \dots, o_n | \omega_{\mathcal{P}}] \Pr[\omega_{\mathcal{P}}]}{\Pr[o_1, \dots, o_n]}. \quad (4.4)$$

Grâce à l'hypothèse d'indépendance conditionnelle des  $(o_i)_{i=1\dots n}$ , cette expression peut être décomposée en

$$f(\mathcal{P}) = \frac{1}{d\mu} \frac{\Pr[\omega_{\mathcal{P}}]}{\Pr[o_1, \dots, o_n]} \left( \prod_{i=1}^n \Pr[o_i | \omega_{\mathcal{P}}] \right), \quad (4.5)$$

soit en appliquant de nouveau la règle de Bayes,

$$\begin{aligned} f(\mathcal{P}) &= \frac{\Pr[\omega_{\mathcal{P}}]}{\Pr[o_1, \dots, o_n]} \left( \prod_{i=1}^n \frac{\Pr[\omega_{\mathcal{P}} | o_i] \Pr[o_i]}{\Pr[\omega_{\mathcal{P}}]} \right) \\ &= \frac{\alpha}{d\mu} \Pr[\omega_{\mathcal{P}}] \left( \prod_{i=1}^n \frac{\Pr[\omega_{\mathcal{P}} | o_i]}{\Pr[\omega_{\mathcal{P}}]} \right). \end{aligned} \quad (4.6)$$

avec

$$\alpha \triangleq \frac{\prod_{i=1}^n \Pr[o_i]}{\Pr[o_1, \dots, o_n]} \quad (4.7)$$

une constante indépendante de  $\mathcal{P}$ .

En injectant le modèle de probabilité  $\Pr[\omega_{\mathcal{P}} | o_i]$  défini équation (4.3) pour chaque  $i = 1 \dots n$ , et en procédant à un développement limité en  $\epsilon$  de cette expression, on aboutit à l'égalité suivante :

$$\begin{aligned} f(\mathcal{P}) &= \frac{\alpha}{d\mu} \Pr[\omega_{\mathcal{P}}] \left( \prod_{i=1}^n \left( \epsilon \frac{f(\mathcal{P} | o_i) d\mu}{\Pr[\omega_{\mathcal{P}}]} + (1 - \epsilon) \right) \right) \\ &= \alpha (1 - \epsilon)^n \left( \frac{\Pr[\omega_{\mathcal{P}}]}{d\mu} + \epsilon \sum_{i=1}^n f(\mathcal{P} | o_i) + o(\epsilon) \right). \end{aligned} \quad (4.8)$$

En pratique, on considère une distribution de probabilité a priori  $\mathcal{P} \mapsto \Pr[\omega_{\mathcal{P}}]$  pour ainsi dire uniforme (voir section 4.6.2), faute d'information supplémentaire. Aussi, les maxima locaux de  $f$  se confondent-ils avec ceux de

$$\mathcal{P} \in \mathcal{C} \mapsto \sum_{i=1}^n f(\mathcal{P} | o_i), \quad (4.9)$$

de sorte que la simple accumulation des régressions probabilistes locales  $\mathcal{P} \mapsto f(\mathcal{P} | o_i)$  est pertinente pour estimer les hypothèses de pose les plus probables en présences de nombre de prédictions aberrantes. Cette approche est ainsi similaire à la forêt de Hough introduite par Gall et al. (2011) pour la détection de piétons, dont nous proposons ici une justification théorique.

## 4.2 Forêt de Hough

Nous réalisons la régression locale visant à estimer la pose de l'instance d'objet à laquelle appartient un point de référence de manière probabiliste, à l'aide d'une forêt de décision, composée de plusieurs arbres de décision.

Le processus de régression mis pour cela en œuvre à partir d'un arbre de décision est illustré figure 4.4. Étant donné un point de référence  $x$

supposé appartenir à une instance d'objet et l'image de profondeur d'entrée, on descend l'arbre suivant le résultat de tests binaires (également appelés *classifieurs faibles*) associés à chaque nœud rencontré, jusqu'à atteindre une feuille de l'arbre. Cette dernière est associée à une distribution de pose apprise préalablement, qui constitue la prédiction.

On fusionne ensuite les distributions de pose prédites pour l'ensemble des points de référence et l'ensemble des arbres d'apprentissage, de manière à obtenir une distribution représentant la probabilité de présence d'une instance d'objet en chaque pose étant donné les données d'entrée.

### 4.2.1 Classifieur faible

Des descripteurs ou classifieurs simples présentant individuellement un faible pouvoir discriminant peuvent se montrer étonnamment efficaces lorsque ceux-ci sont combinés au sein d'architectures profondes. Une tendance de fond en apprentissage par ordinateur consiste à se tourner vers ce type d'approche, qui ont été étudiées notamment dans le cadre des cascades de classifieurs, des forêts de décision, et aujourd'hui des réseaux de neurones profonds. Notre approche basée sur une forêt de décision s'inscrit donc dans cette tendance, et nous considérons pour chaque nœud de chaque arbre d'apprentissage un classifieur binaire faible  $h$ . Dans la suite, nous notons  $x$  un point de référence de l'image de profondeur  $Z$  d'entrée. Suivant une approche usuelle ([Criminisi et Shotton, 2013](#)), on produit un tel classifieur par comparaison de la sortie d'un descripteur scalaire faible  $g$  avec un seuil fixe  $\tau$ . Le descripteur considéré est paramétré par un jeu de paramètres  $\theta$  propre au nœud considéré :

$$h(x, Z, \theta, \tau) = [g(x, Z, \theta) < \tau] \in \{0, 1\}. \quad (4.10)$$

Une famille de descripteurs adapté aux images de profondeur particulièrement populaire pour sa simplicité consiste en les différences de profondeur entre deux pixels de coordonnées  $u_1, u_2 \in \mathbb{R}^2$  choisis autour du projeté du point de référence dans l'image  $u = \Pi(x) \in \mathbb{R}^2$  :

$$Z(u_2) - Z(u_1). \quad (4.11)$$

Celle-ci a notamment été popularisée par le succès des travaux de [Shotton et al. \(2013b\)](#) sur l'estimation de pose humaine à l'aide du capteur Kinect, reprenant eux-mêmes une approche déjà exploitée dans des images 2D par exemple chez [Ozuysal et al. \(2010\)](#). Nous nous inspirons de cette dernière afin de définir notre propre famille de descripteurs, dont l'expression finale est donnée équation (4.18).

#### 4.2.1.1 Invariance en perspective

Régresser la pose d'une instance d'objet de manière locale relativement à un point de référence  $x$  appartenant à cette instance et non de manière absolue permet de simplifier grandement le problème d'estimation de pose. En effet, pour peu que l'on soit capable de travailler de manière invariante en la position de  $x$ , il suffit d'être capable de régresser la pose d'une instance

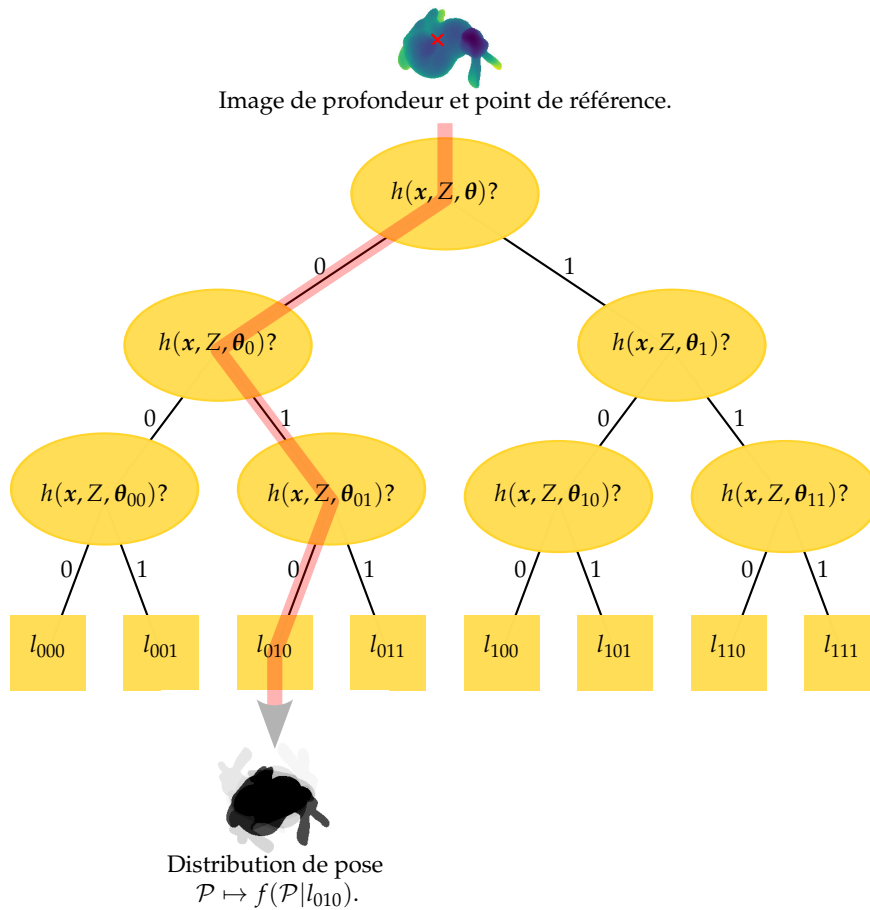


FIGURE 4.4 – Illustration du processus de régression probabiliste de pose à l’aide d’un arbre de décision. Étant donné une image de profondeur  $Z$  en entrée et un point de référence  $x$  supposé appartenir à une instance d’objet, on descend les branches de l’arbre suivant un parcours dépendant du résultat d’un test binaire  $h(x, Z, \theta_i)$  de paramètre  $\theta_i$  associé à chaque nœud  $i$  traversé. On retourne alors la distribution de probabilité associée au nœud terminal (ou *feuille*) atteint, exprimée dans un repère lié à  $x$ .

relativement à un point de référence donné pour être capable d'effectuer cette régression pour n'importe quel point de référence.

Il s'agit donc d'être capable de s'abstraire de la position de  $x$ . Pour ce faire, nous exprimons la pose de l'objet dans un repère local  $\mathcal{F}_x$  propre au point d'intérêt considéré, et définissons notre famille de descripteur de manière à être invariante en la position de  $x$ , suivant le modèle de projection du capteur de profondeur considéré. Nous discutons dans cette section le cas d'un capteur de profondeur perspective, tel qu'un capteur stéréo binoculaire, ou un capteur Kinect. Il serait relativement facile d'étendre les résultats de cette discussion à d'autres modèles. Notamment, un capteur à projection orthographique tel qu'un scanner linéaire plan LASER peut être vu comme un cas limite de capteur perspective, de focale  $f$  infinie, à distance infinie  $z$  de la scène et présentant un facteur d'échelle  $s$  entre coordonnées dans le monde et dans l'image :

$$\left\{ \begin{array}{l} f \rightarrow \infty \\ z \rightarrow \infty \\ \frac{f}{z} \rightarrow s. \end{array} \right. \quad (4.12)$$

**Perspective faible** Les approches de détection d'objet dans une image décrites dans la littérature considèrent généralement implicitement un modèle de projection en perspective dit *faible*. Suivant celui-ci, un objet translaté suivant une direction parallèle au plan image du capteur produit une image similaire à celle initiale, translatée dans le plan image. Une translation suivant l'axe optique du capteur est quant-à-elle supposée induire une homothétie de l'image de l'objet. Toute les vues d'un même objet suivant une orientation donnée relativement à la caméra sont ainsi considérées identiques à une translation 2D et un changement d'échelle près, indépendamment de sa position, et il est possible d'exploiter cette propriété pour concevoir une représentation invariante en translation. C'est notamment là la base des approches de détection d'objet de type *fenêtre glissante* où un patron 2D d'objet est recherché dans une image à différentes positions et échelles (cf. section 2.4.1 de l'état de l'art).

De manière formelle, en notant  $\bar{z}$  la profondeur typique de l'objet considéré et  $(x, y, z)^\top \in \mathbb{R}^3$  un point de ce dernier, les coordonnées  $(u, v)^\top \in \mathbb{R}^2$  de ce point dans l'image sont estimées selon le modèle de perspective *faible* comme suit :

$$\left\{ \begin{array}{l} u = c_u + f \cdot x / \bar{z} \\ v = c_v + f \cdot y / \bar{z}. \end{array} \right. \quad (4.13)$$

Il s'agit d'une approximation raisonnable lorsque l'épaisseur de l'objet est faible devant la distance à la caméra ( $|z - \bar{z}| \ll \bar{z}$ ), et lorsque l'angle entre l'axe optique de la caméra et la direction de l'objet est faible ( $\sqrt{x^2 + y^2} \ll z$ ).

Dans leurs travaux, [Shotton et al. \(2013b\)](#) adaptent notamment le descripteur (4.11) à ce modèle, en le paramétrisant par deux décalages  $\delta_1, \delta_2 \in \mathbb{R}^2$  dans le plan  $(e_x, e_y)$  parallèle au capteur :

$$Z \left( \mathbf{u} + \frac{f}{z} \delta_2 \right) - Z \left( \mathbf{u} + \frac{f}{z} \delta_1 \right), \quad (4.14)$$

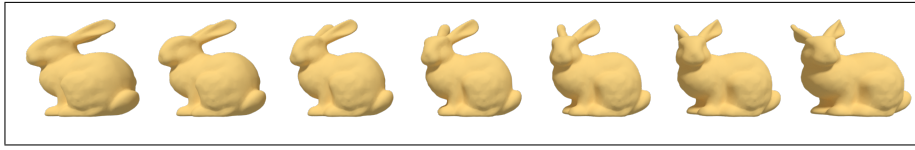


FIGURE 4.5 – Parallaxe due à la translation de l’objet parallèlement au plan image. Illustration sur un champ de vision horizontal de  $90^\circ$ .

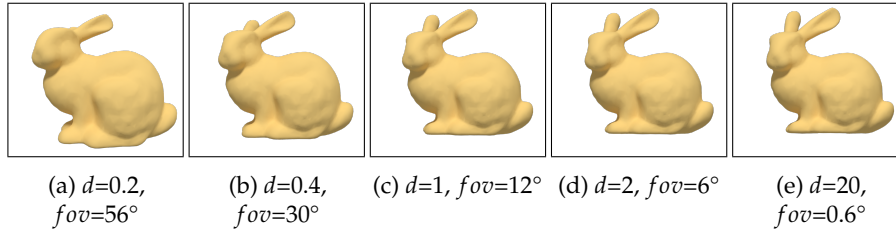


FIGURE 4.6 – Parallaxe due à la translation de l’objet suivant un rayon passant par le centre optique. Illustration pour différentes distances  $d$  d’un objet à la caméra (objet de diamètre typique 0.2). Le champ de vision  $fov$  de la caméra a été adapté de manière à ce que les images de l’objet aient des dimensions identiques quelque soit sa profondeur.

où on note  $\mathbf{u} = \Pi(\mathbf{x}) \in \mathbb{R}^2$  les coordonnées du projeté dans l’image du point de référence  $\mathbf{x} = (x, y, z)^\top \in \mathbb{R}^3$ . La réponse du descripteur (4.14) pour un objet donné est alors sensée être indépendante de la position du point de référence considéré selon le modèle de perspective faible, dès lors que la pose de l’objet, exprimée relativement au repère local  $\mathcal{F}_x^{faible} = (x, e_x, e_y, e_z)$  lié à  $\mathbf{x}$  est fixée. Ce repère est illustré figure 4.7b.

**Meilleure prise en compte de la parallaxe** Le modèle de perspective faible est cependant un modèle approché. Un objet se déplaçant suivant une direction parallèle au plan image ne produit en réalité pas une image toujours identique, mais semble être observé suivant des orientations différentes du fait de la parallaxe. Ce phénomène est particulièrement visible lorsque l’angle de champ est important, ainsi qu’illustré figure 4.5. L’image d’un objet s’éloignant suivant un rayon passant par l’axe optique ne subit de même pas qu’un simple facteur d’échelle mais présente bien un point de vue différent de l’objet ainsi qu’illustré figure 4.6. Ce second point est cependant moins problématique que le premier en pratique car ses effets sont relativement faibles dès lors que la taille apparente de l’objet est suffisamment petite (typiquement inférieure à  $20^\circ$ ), ce qui est fréquent dans notre cas d’application où de nombreuses instances d’objet se partagent le champ de vision de la caméra.

Nous nous proposons donc d’adapter la famille de descripteurs considérée de manière à mieux prendre en compte ce phénomène de parallaxe, en rendant cette dernière invariante à la direction du rayon optique passant par le point d’intérêt  $\mathbf{x}$ . Pour ce faire, nous travaillons dans un repère local  $\mathcal{F}_x = (x, e_1, e_2, e_3)$  illustré figure 4.7c, où  $e_3$  est aligné avec la direction du

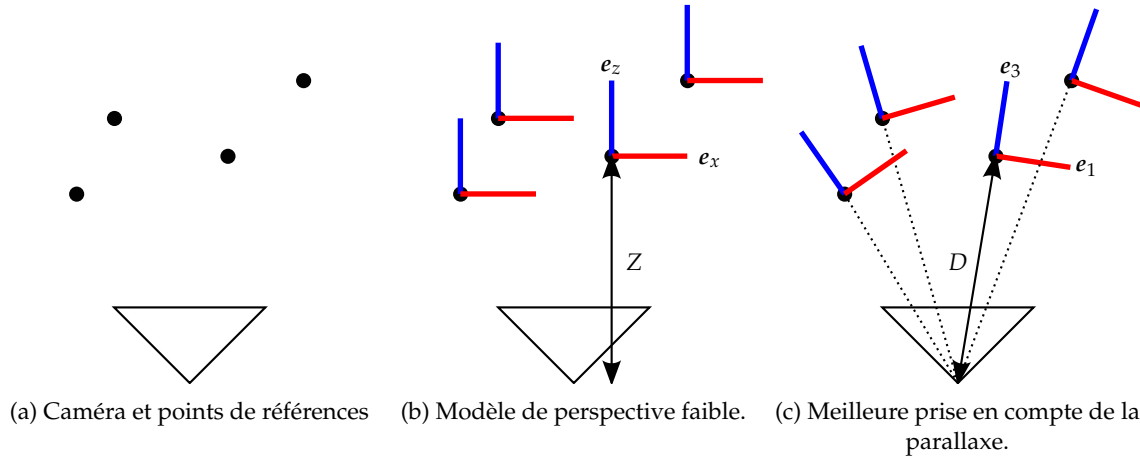


FIGURE 4.7 – Repère relatif à un point de référence, défini afin de formuler la régression de pose de manière invariante en la position de ce point.

rayon optique passant par  $x$  caméra, et défini par

$$\begin{cases} \mathbf{e}_1 = \frac{\mathbf{e}_x - (\mathbf{e}_x \cdot \mathbf{e}_3)\mathbf{e}_3}{\|\mathbf{e}_x - (\mathbf{e}_x \cdot \mathbf{e}_3)\mathbf{e}_3\|} \\ \mathbf{e}_2 = \mathbf{e}_3 \times \mathbf{e}_1 \\ \mathbf{e}_3 = \frac{\mathbf{x}}{\|\mathbf{x}\|}. \end{cases} \quad (4.15)$$

La notion de profondeur  $Z$  – c.-à-d. de distance au plan image – n'est pas invariante suivant la direction du rayon optique passant par  $x$ . Aussi, nous remplaçons celle-ci par la notion de distance au centre optique  $D$ , illustrée figure 4.8b. Le passage de l'image de profondeur  $Z$ , à une image de distance au centre optique  $D$  peut s'effectuer par simple multiplication élément par élément par une matrice  $A$  de la manière suivante :

$$D(u, v) = \underbrace{\left( \sqrt{\left(\frac{u - c_u}{f}\right)^2 + \left(\frac{v - c_v}{f}\right)^2} + 1 \right)}_{A(u, v)} Z(u, v), \forall (u, v) \in \mathbb{R}^2. \quad (4.16)$$

$A$  peut être précalculée indépendamment de  $Z$ , rendant le passage d'une représentation à l'autre très peu coûteuse. Nous définissons alors un descripteur basé sur la différence de distance au centre optique entre deux pixels, de coordonnées paramétrées par deux offsets  $(\delta_{i,1}, \delta_{i,2}) \in \mathbb{R}^2, i = 1, 2$  dans le plan  $(\mathbf{e}_1, \mathbf{e}_2)$  :

$$D(\Pi(\mathbf{x} + \delta_{2,1}\mathbf{e}_1 + \delta_{2,2}\mathbf{e}_2)) - D(\Pi(\mathbf{x} + \delta_{1,1}\mathbf{e}_1 + \delta_{1,2}\mathbf{e}_2)). \quad (4.17)$$

Cette famille de descripteurs permet ainsi une meilleure prise en compte de la parallaxe qu'une famille basée sur un modèle de perspective faible,

ce qui accroît son pouvoir discriminant. Une fois intégrée dans notre forêt de décision, entraînée à partir de vues de l'objet suivant différentes orientations, cette famille de descripteur permet en effet de générer des distributions de pose présentant des modes de densité bien marqués même selon des directions éloignées du centre optique comme illustré figure 4.8. A contrario, les descripteurs (4.14) produisent une distribution de pose plus étalée, notamment loin de l'axe optique où ont été réalisés les vues d'apprentissages. Le surcoût en temps de calcul lié à l'utilisation de ce descripteur au lieu de (4.14) étant négligeable devant le temps total d'exécution dans notre implémentation, nous choisissons donc de privilégier ce dernier.

La parallaxe liée à la variation de distance à la caméra n'est néanmoins toujours pas parfaitement prise en compte par le descripteur, cependant celle-ci reste limitée en pratique ainsi qu'évoqué précédemment.

#### 4.2.1.2 Saturation du descripteur

Nous pensons qu'une part conséquente de l'information dans une image de profondeur concernant la pose d'une instance est apportée par la silhouette de celle-ci. Les contours de cette silhouette sont cependant assez mal définis en général dans les données de profondeur, de par les technologies d'acquisition 3D utilisées. L'information de silhouette est néanmoins présente dans ces données sous la forme d'une différence de profondeur souvent notable entre les points de la silhouette de l'objet et ceux à l'extérieur (le fond). Nous souhaitons permettre au descripteur de représenter cette différence, sans pour autant dépendre de la profondeur du fond.

Pour se faire, nous choisissons de saturer la réponse du descripteur, en la contraignant dans un intervalle  $[-S, S]$  où  $S$  est une constante choisie suivant les dimensions de l'objet (fixée à 75% de l'épaisseur de l'objet). L'introduction de cette saturation permet de modéliser partiellement cette invariance en la profondeur du fond, en ce que si  $u_1$  correspond à un pixel de l'objet, et  $u_2$  à un pixel du fond, la profondeur de  $u_2$  est alors typiquement plus grande que celle de  $u_1$  d'une valeur au moins égale à l'épaisseur typique de la pièce, et le descripteur vaut alors  $S$ , indépendamment  $D(u_2)$ . Le descripteur type que l'on considère dans notre forêt de décision, paramétré par  $\theta = (\delta_{i,j})_{i,j=1,2} \in \mathbb{R}^4$ , est donc finalement défini pour un point de référence  $x$  et une image de distance au centre optique  $D$  de la manière suivante :

$$g(x, D, \theta) = \min(\max(D(u_2) - D(u_1), -S), S). \quad (4.18)$$

avec  $u_i = \Pi(x + \delta_{i,1}e_1 + \delta_{i,2}e_2)$ , pour  $i = 1, 2$ .

**Absence d'information** En certains pixels, il n'y a pas d'information de profondeur disponible, aussi la sortie du descripteur peut ne pas être définie. C'est particulièrement le cas autour de la silhouette des objets du fait d'un phénomène d'ombre portée évoqué section 4.1.1. Afin de tolérer qu'un des pixels  $u_1, u_2$  soit dans une de ces ombres portées, nous affectons une profondeur infinie (Inf) aux pixels pour lesquels on ne dispose pas d'information de profondeur. Suivant les règles de calcul flottant définies par la norme



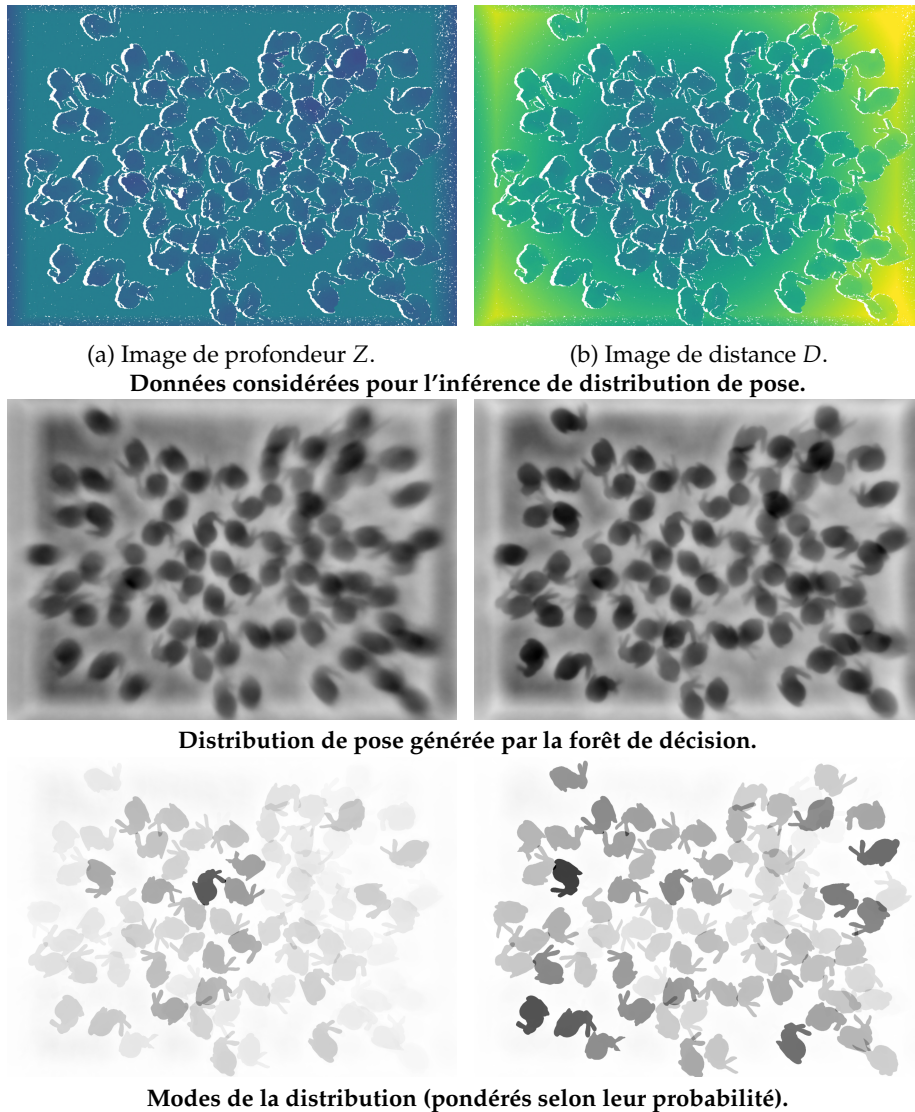


FIGURE 4.8 – Influence du modèle de perspective considéré pour rendre les descripteurs de notre forêt de décision invariants en la position du point de référence. Illustration dans le cas d'un capteur grand angle (champ horizontal de  $84^\circ$ ). Le modèle de perspective faible, utilisant pour descripteur la différence de profondeur entre deux pixels (à gauche) conduit à produire une distribution de pose relativement étalée dès lors que l'on s'éloigne de l'axe optique, où a été réalisé l'apprentissage. Les modes de cette distribution sont conséquemment moins marqués loin de l'axe optique, rendant la détection des instances d'objet correspondantes plus difficile. Le modèle proposé, basé sur la différence entre deux pixels de distance au centre optique (à droite) modélise mieux le phénomène de perspective, et permet une détection des instances plus uniforme indépendamment de leur position dans l'image.

IEEE 754, la sortie du descripteur vaut alors

$$\left\{ \begin{array}{ll} \min(\max(D(\mathbf{u}_2) - D(\mathbf{u}_1), -S), S) & , \text{ si } D(\mathbf{u}_1) \neq \text{Inf et } D(\mathbf{u}_2) \neq \text{Inf} \\ S & , \text{ si } D(\mathbf{u}_1) = \text{Inf et } D(\mathbf{u}_2) \neq \text{Inf} \\ -S & , \text{ si } D(\mathbf{u}_1) \neq \text{Inf et } D(\mathbf{u}_2) = \text{Inf} \\ \text{non définie} & \text{sinon } (D(\mathbf{u}_1) = D(\mathbf{u}_2) = \text{Inf}). \end{array} \right. \quad (4.19)$$

Il s'agit là d'un choix relativement arbitraire, mais dont l'utilisation a montré un gain de performance lors de nos expérimentations. Il peut toujours arriver que la valeur d'un tel descripteur ne soit pas définie lorsque l'on ne dispose d'information de profondeur ni en  $\mathbf{u}_1$  ni en  $\mathbf{u}_2$ . En ce cas, on choisit lors de l'exécution la branche dans laquelle poursuivre la descente de l'arbre de manière aléatoire suivant une loi équiprobable.

### 4.2.2 Aggrégation des votes

Étant donné un point de référence  $x_i$ , (pour  $i \in \llbracket 1, m \rrbracket$ ), on atteint dans chaque arbre de décision  $\mathcal{T}_j$ , avec  $j \in \llbracket 1, n \rrbracket$ , une feuille  $l_{i,j}$  en descendant les branches de ce dernier suivant les résultats des classifieurs faibles associées aux nœuds explorés. Dans cette feuille est stockée une distribution représentant la densité de probabilité de pose de l'instance à laquelle appartient  $x_i$ , sachant la sortie des classifieurs faibles explorés pour atteindre cette feuille. Cette distribution est représentée de manière parcimonieuse pour des raisons pratiques, sous forme de  $p$  échantillons de pose  $\{[T_{i,j,k}]\}_{k=1\dots p} \subset \mathcal{C}$  exprimés relativement au repère local  $\mathcal{F}_{x_i}$ , et associés à des poids  $\{w_k\}_{k=1\dots p} \subset \mathbb{R}^{+*}$ . Ces échantillons peuvent être vus comme des votes pour des poses particulières, suivant le formalisme des méthodes de type Hough.

Nous fusionnons alors l'ensemble des distributions obtenues pour chacun des points de référence et chacun des arbres dans un repère absolu  $\mathcal{R}$ , de manière à obtenir une distribution représentant la probabilité de présence d'une instance d'objet en chaque pose, étant donné l'entièreté des informations à notre disposition. Cette fusion est produite en moyennant les distributions locales obtenues, suivant le modèle probabiliste proposé sous-section 4.1.2. Dans le cadre de notre représentation parcimonieuse, ce moyennage revient à agréger l'ensemble des votes des différents arbres et points de référence ensemble, après les avoir exprimés dans un même repère. La distribution finale est ainsi représentée par l'ensemble des couples de poses et de poids associés :

$$\bigcup_{i=1\dots m, j=1\dots n} \left\{ ([T_{\mathcal{F}_{x_i} \rightarrow \mathcal{R}} \circ T_{i,j,k}], w_k) \mid k = 1 \dots p \right\}, \quad (4.20)$$

où  $T_{\mathcal{F}_{x_i} \rightarrow \mathcal{R}}$  représente la transformation rigide active permettant de passer du repère  $\mathcal{F}_{x_i}$  au repère  $\mathcal{R}$ .

**Approche multi-vue** Cette technique d'agrégation de votes rend la fusion de prédictions issues de plusieurs vues d'une même scène particulièrement simple, en ce qu'il suffit d'agréger ensemble les votes générés pour les

différentes vues, une fois exprimés dans un même repère. Nous n'avons néanmoins pas expérimenté un tel scénario dans le cadre de nos travaux.

### 4.3 Extraction d'hypothèses de pose

La distribution de pose précédente synthétise l'ensemble des informations locales concernant la pose des instances d'objets de la scène, et dès lors les maxima locaux (ou *modes*) de cette distribution constituent des hypothèses pertinentes concernant la pose de celles-ci.

Nous extrayons ces modes grâce à la technique de Mean Shift introduite section 3.10.1, qui consiste à faire converger des poses vers des modes  $\{\mathcal{P}_i^m\}_i$  d'une distribution de densité représentée par un ensemble d'échantillons. La pseudo-densité en chacun de ces modes  $\mathcal{P}_i^m$  est alors estimée via un estimateur à noyau

$$f(\mathcal{P}_i^m) = \sum_k w_k H\left(\frac{d(\mathcal{P}_i^m, \mathcal{P}_k)}{h_0}\right), \quad (4.21)$$

où  $\{(\mathcal{P}_k, w_k)\}_k$  représente l'ensemble des votes pondérés de la distribution,  $H$  le noyau d'Epanechnikov défini équation (3.100), et  $h_0$  le rayon du noyau considéré. Le choix de ce rayon présente un impact notable sur les performances de la méthode, ainsi qu'illustré figure 4.9. Le choix d'un rayon trop faible n'introduit en effet pas la régularisation suffisante pour compenser le caractère épars de l'agrégat de votes de pose et conduit à la détection de nombreux maxima locaux ne correspondant pas nécessairement à des instances d'objet. Au contraire, un rayon trop grand introduit trop de régularisation et n'offre pas le pouvoir de résolution suffisant pour distinguer des instances d'objet distinctes. Dans le cadre de nos expérimentations, on assigne à ce rayon la valeur de  $\min(\lambda_1, \lambda_2, \lambda_3)$ , où  $\lambda_1, \lambda_2, \lambda_3$  représentent les valeurs propres de la matrice  $\Lambda$  introduite section 3.5.3. Cette valeur correspond à la moitié de l'épaisseur typique de l'objet, et il s'agit là d'une bonne heuristique dans le cas des objets testés.

On extrait alors les  $k \in \mathbb{N}^*$  hypothèses de pose correspondant aux modes principaux de la distribution parmi l'ensemble des  $\{\mathcal{P}_i^m\}_i$ , en filtrant les doublons dans un rayon  $d_0 = h_0/2$  autour des hypothèses extraites, suivant l'algorithme 2. En pratique, entre 5 et 10 hypothèses sont suffisantes afin de pouvoir localiser de manière relativement fiable suffisamment d'instances d'objet pour une application robotique. Dans le cadre de nos évaluations, nous considérons en revanche parfois jusqu'à  $k = 100$  hypothèses de manière à évaluer le taux de rappel qu'il est possible d'atteindre.

### 4.4 Raffinement d'hypothèses

Afin que les hypothèses de pose générées soient ajustées au mieux aux données de profondeur, nous procédons à une étape de raffinement de ces dernières. Nous mettons pour cela en œuvre une méthode de type *Iterative Closest Point* (ICP, Besl et McKay (1992)), tout en exploitant le fait que l'on

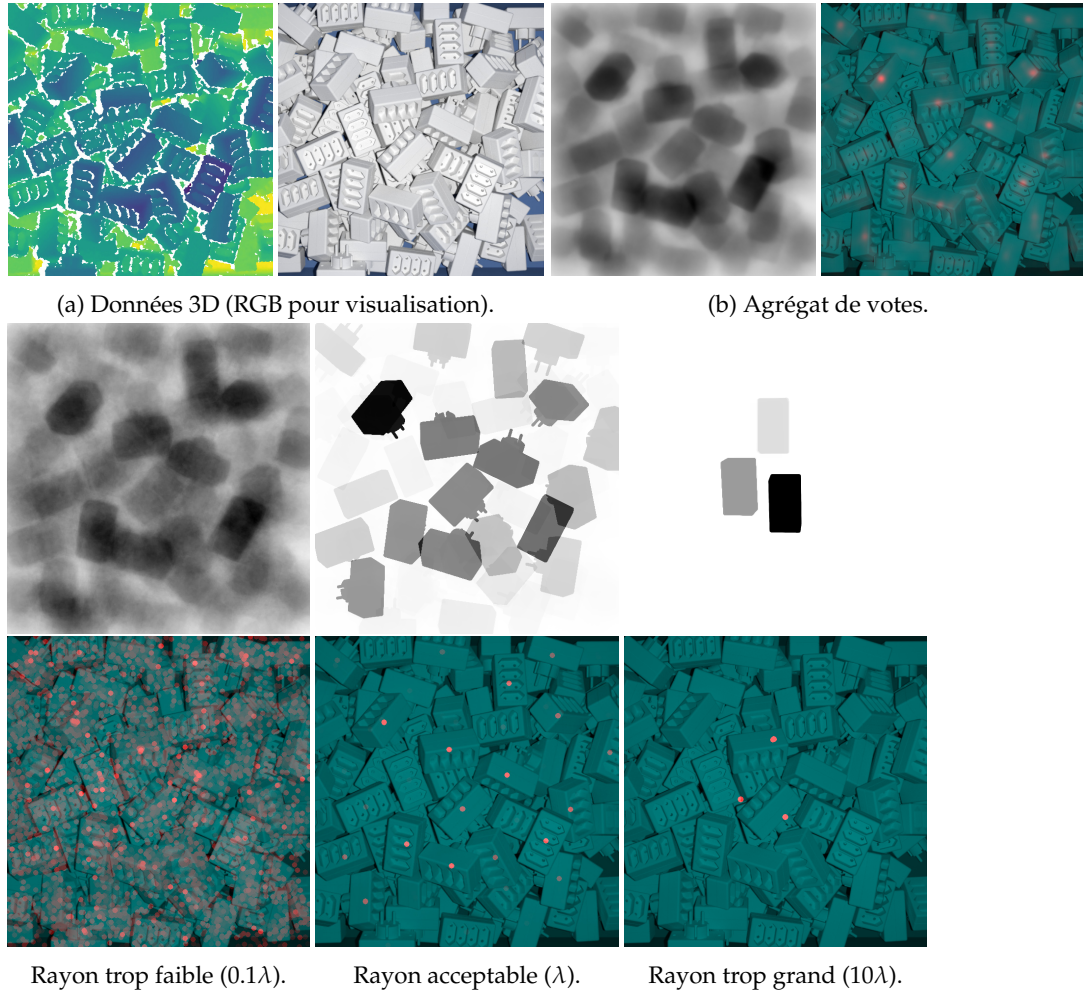


FIGURE 4.9 – Influence du rayon de Mean Shift sur l'extraction d'hypothèses de pose de l'agrégat de votes initial. Une distribution de pose est représentée par superposition des silhouettes de l'objet en les poses de cette distribution (images en niveau de gris), ainsi que par les projections du centre de l'objet dans le plan image (carte de chaleur en rouge superposée à la scène en bleu).

---

**Algorithme 2** Extraction d'au plus  $k$  modes principaux de la distribution de pose, à partir de la sortie de l'algorithme Mean Shift.

---

**Entrée:**

Poses  $\{\mathcal{P}_i^m\}_{i=1\dots n}$  ayant convergé vers des modes de la distribution à la suite de Mean Shift,  
 $d_0$  une distance seuil.

**Sortie:**

Ensemble  $\mathcal{H}$  d'au plus  $k$  hypothèses de poses .

```

1: Trier les  $\{\mathcal{P}_i^m\}_i$  par ordre décroissant de  $f(\mathcal{P}_i^m)$ .
2:  $\{\mathcal{P}_1^m\} \rightarrow \mathcal{H}$ 
3:  $2 \rightarrow i$ 
4: while  $|\mathcal{H}| < k$  and  $i \leq n$  do
5:   true  $\rightarrow$  valide
6:    $1 \rightarrow j$ 
7:   while valide and  $j \leq |\mathcal{H}|$  do
8:     if  $d(\mathcal{P}_i^m, \mathcal{H}[j]) < d_0$  then
9:       false  $\rightarrow$  valide
10:    end if
11:     $j + 1 \rightarrow j$ 
12:  end while
13:  if valide then
14:    Ajouter  $\mathcal{P}_i^m$  à  $\mathcal{H}$ 
15:  end if
16:   $i + 1 \rightarrow i$ 
17: end while

```

---

travaille avec des images de profondeur en procédant suivant la manière suivante, pour chaque hypothèse de pose :

1. Nous générons un rendu d'image de profondeur idéale correspondant à la pose supposée de l'objet.
2. Nous apparions chaque point  $\mathbf{y}_i \in \mathbb{R}^3$  du rendu ( $i \in \llbracket 1, n \rrbracket$ ) au point le plus proche de celui-ci  $\mathbf{x}_i \in \mathbb{R}^3$  parmi les points de l'image de profondeur d'entrée, pourvu que la distance entre ces derniers soient inférieur à un seuil  $\delta \in \mathbb{R}^{+*}$ .
3. Nous estimons la transformation rigide  $\hat{T} \in SE(3)$  à appliquer aux  $(\mathbf{y}_i)_i$  de manière à minimiser le carré des distance entre ces deux jeux de points

$$\hat{T} = \operatorname{argmin}_T \sum_{i=1}^n (T\mathbf{y}_i - \mathbf{x}_i)^2. \quad (4.22)$$

4. La transformation  $\hat{T}$  est alors appliquée à l'hypothèse de pose, et le processus est réitéré jusqu'à convergence ou un épuisement d'un nombre fixé d'itérations.

La distance seuil  $\delta$  permet d'introduire une certaine résistance aux appariements aberrants. Dans sa version basique, ICP reste cependant propice à converger vers un minimum local ne correspondant pas à l'alignement souhaité. Nous traitons ce point au moyen d'une approche multi-résolution en procédant au raffinement suivant une pyramide d'images de profondeur illustrée figure 4.10 de résolution allant croissantes à mesure que l'on considère une distance d'appariement  $\delta$  plus faible, selon une approche similaire à celle décrite par [Jost et Hügli \(2003\)](#). Les étages basses résolutions de la pyramide introduisent une certaine régularisation des appariements de par le moyennage des valeurs de profondeur réalisé lors du sous-échantillonnage, tout en accélérant les calculs, tandis que les étages plus de haute résolution permettent un alignement précis.

## 4.5 Vérification et filtrage

L'ensemble des hypothèses de pose générées ne correspondent pas nécessairement à de véritables instances d'objet, aussi afin d'accroître la précision de notre méthode, nous procédons à un filtrage de ces dernières. Dans un premier temps, nous vérifions la pertinence vis-à-vis des données de chaque hypothèse indépendamment (section 4.5.1) de manière à ne conserver que les hypothèses les plus cohérentes. Nous procédons ensuite à un second filtrage (section 4.5.2) visant à assurer la cohérence globale de la liste de poses d'instances retournées.

### 4.5.1 Qualité intrinsèque d'une hypothèse

Afin d'évaluer la qualité intrinsèque d'une hypothèse de pose  $\mathcal{P}$ , nous estimons la pertinence de celle-ci vis-à-vis des données disponibles. Nous avons pour cette étape fait le choix de la simplicité et les critères que nous

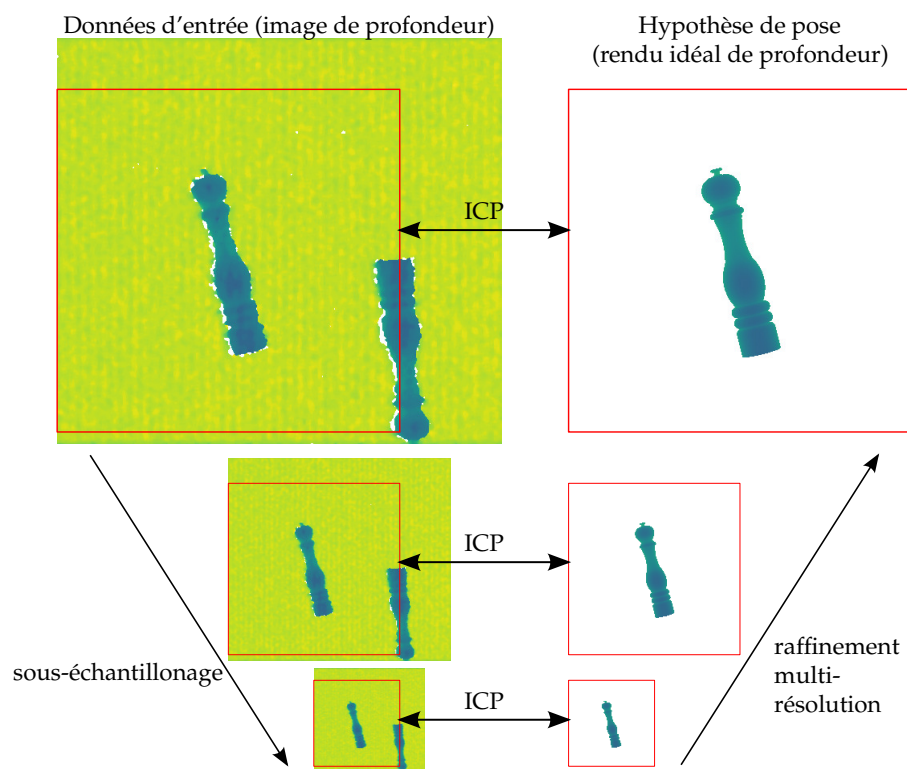


FIGURE 4.10 – Raffinement multi-résolution des hypothèses de poses.

utilisons sont relativement grossiers, cependant ceux-ci permettent en pratique de discriminer relativement bien les bonnes hypothèses de poses des autres.

**Pertinence vis à vis des données de profondeur** Nous comparons une image de profondeur synthétique  $Z_s$  de l'objet dans cette pose (telle que celle illustrée en haut à droite figure 4.10) avec l'image de profondeur réelle  $Z$ .

Chaque pixel  $\mathbf{u}$  de la silhouette  $\mathcal{S} \subset \mathbb{R}^2$  sur le plan image de l'objet dans la pose  $\mathcal{P}$  est considéré indépendamment, et on évalue l'adéquation de la profondeur correspondante à ce dernier dans les données  $Z(\mathbf{u})$  avec celle du rendu synthétique  $Z_s(\mathbf{u})$ . Nous considérons pour se faire une distance seuil  $\delta$  représentant l'incertitude typique en profondeur des données produites par le capteur, et classifions cette adéquation suivant différents critères illustrés figure 4.11 :

- On considère ainsi que les données synthétiques sont *appariées* avec les données réelles en  $\mathbf{u}$  lorsque la profondeur synthétique est similaire à celle réellement mesurée :  $Z_s(\mathbf{u}) \in [Z(\mathbf{u}) - \delta, Z(\mathbf{u}) + \delta]$ .
- De même, l'hypothèse de pose est considérée comme *cohérente* en  $\mathbf{u}$  lorsque la profondeur synthétique est similaire ou supérieure à celle observée des données réelles, ce dernier cas pouvant se produire pour des raisons d'occultation de l'objet :  $Z_s(\mathbf{u}) \geq Z(\mathbf{u}) - \delta$ .
- Au contraire, l'hypothèse de pose est considérée comme *incohérente* en  $\mathbf{u}$  lorsque la profondeur du rendu synthétique est significativement inférieure à celle des données, ce qui ne fait pas sens dès lors que le capteur de profondeur n'est pas capable d'imager à travers la matière :  $Z_s(\mathbf{u}) < Z(\mathbf{u}) - \delta$ .
- Dans le cas où la profondeur  $Z(\mathbf{u})$  de ce pixel n'est pas définie dans l'image réelle (trou dans l'image de profondeur ou hypothèse de pose sortant du champ du capteur), le pixel n'entre dans aucune de ces catégories.

On agrège alors ces informations pour l'ensemble de la silhouette  $\mathcal{S}$  sous forme d'indicateurs globaux. On définit le *ratio d'appariement*, comme la fraction de la silhouette pour laquelle données synthétiques et réelles sont appariées :

$$\text{matchingRatio} = \frac{|\{\mathbf{u} \in \mathcal{S} | Z_s(\mathbf{u}) \in [Z(\mathbf{u}) - \delta, Z(\mathbf{u}) + \delta]\}|}{|\mathcal{S}|}. \quad (4.23)$$

Dans le cas d'une hypothèse de pose valide, un tel ratio traduit typiquement le taux d'occultation de l'instance dans l'image. Dès lors que la détection d'instances très occultées n'a pas nécessairement d'intérêt dans notre cadre applicatif, il est possible filtrer ces dernières en ne conservant que celles ayant un ratio d'appariement supérieur à un seuil donné. De manière générale, cette grandeur traduit le support par les données de l'hypothèse de pose et une valeur importante apporte ainsi une confiance certaine dans la vraisemblance de l'hypothèse. Il est donc sensé d'ordonner les hypothèses de poses retournées par ratio d'appariement décroissant, de manière à privilégier les plus vraisemblables.



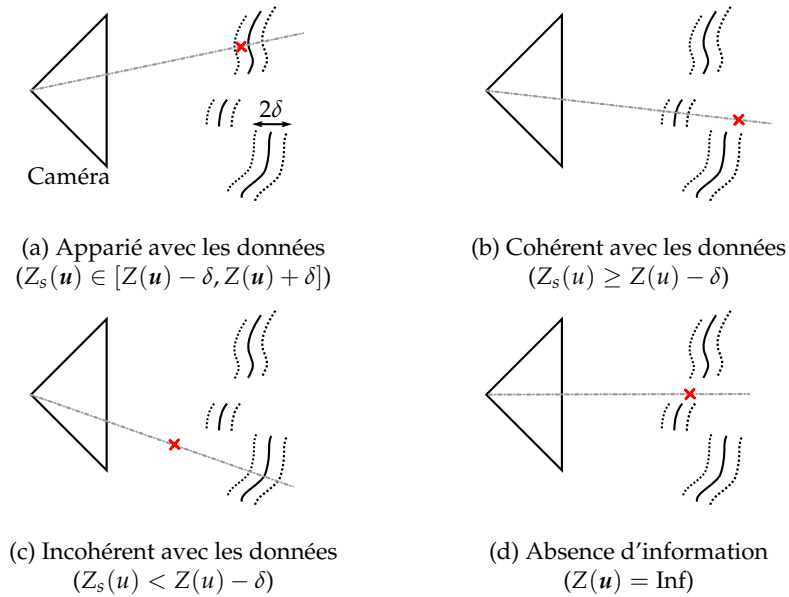


FIGURE 4.11 – Caractérisation locale d'une hypothèse de pose  $\mathcal{P}$  par comparaison en un pixel  $\mathbf{u}$  (rayon pointillé) de la profondeur  $Z_s(\mathbf{u})$  en ce pixel d'un rendu synthétique de l'objet en cette pose (croix rouge), avec la profondeur  $Z(\mathbf{u})$  observée dans les données réelles (courbe noire).

On définit également le *ratio d'incohérence* comme la fraction des pixels de la silhouette pour lesquels l'hypothèse de pose est incohérente avec les données :

$$\text{inconsistencyRatio} = \frac{|\{\mathbf{u} \in \mathcal{S} | S(\mathbf{u}) < D(\mathbf{u}) - \delta\}|}{|\mathcal{S}|}. \quad (4.24)$$

Ce ratio devrait être nul dans le cas d'une hypothèse valide et de données idéales. Du fait du bruit du capteur et de l'alignement non parfait des hypothèses de poses avec les données, il convient néanmoins de tolérer un certain écart à cette idéalité.

**Pertinence vis à vis des contours** Dans le cas où l'on dispose également en entrée d'une image RGB ou d'intensité, il est possible d'exploiter cette modalité afin de tester la pertinence d'une hypothèse. Nous extrayons pour ce faire les contours détectés dans l'image afin de les comparer avec ceux de l'hypothèse de pose. Ces contours sont segmentés par simple seuillage de la norme du gradient, et afin de nous affranchir de la texture de la scène nous ignorons les contours 2D ne correspondant pas à des discontinuités dans l'image de profondeur. La figure 4.12 présente un exemple de tels contours extraits.

Les pixels de contours sont considérés comme orientés selon la direction du gradient (modulo  $\pi$ ), et on estime la fraction des pixels  $\mathbf{c} \in \mathcal{C}$  des contours de l'hypothèse de pose dont l'orientation  $\text{ori}(\mathcal{H}, \mathbf{c})$  est cohérente

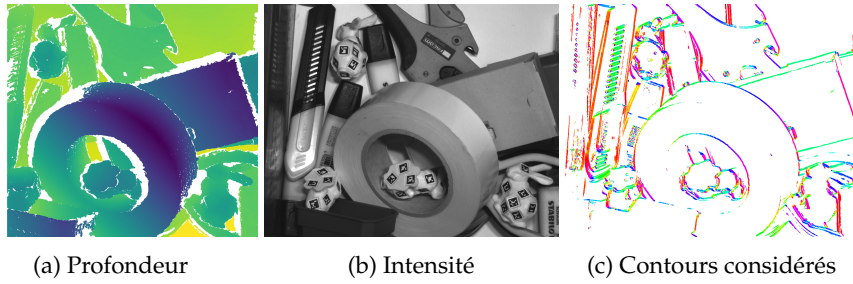


FIGURE 4.12 – Extraction de contours de profondeur orientés, par filtrage des contours extraits d’une image d’intensité dans les zones sans discontinuité de profondeur.

avec celle  $\text{ori}(\mathcal{I}, c)$  observée dans l’image en ce pixel, au moyen de la mesure de similarité proposée par [Hinterstoisser et al. \(2011\)](#) :

$$\text{edgesMatchingRatio} = \frac{1}{|\mathcal{C}|} \left( \sum_{c \in \mathcal{C}} \max_{t \in [-r, r]^2} |\cos(\text{ori}(\mathcal{I}, c + t) - \text{ori}(\mathcal{H}, c))| \right). \quad (4.25)$$

Cette mesure introduit également une robustesse aux petites erreurs de position de moins de  $r$  pixels, et les orientations sont discrétisées pour des raisons d’efficacité de calcul. Le lecteur est renvoyé aux publications de [Hinterstoisser et al. \(Hinterstoisser et al., 2011, 2012a\)](#) pour plus d’information.

**Filtrage** Les différentes mesures de pertinence d’une hypothèse de pose présentées précédemment peuvent être utilisées afin de filtrer les hypothèses retournées, au moyen de seuils durs. Les hypothèses présentant un ratio d’incohérence non négligeable (p. ex. supérieur à 10%) peuvent par exemple être écartées sans risque important d’erreur, et de même il est possible d’écartier les hypothèses présentant un appariement faible avec les données de profondeur ou les contours pourvu que l’on ne soit pas intéressé par la détection et l’estimation de pose d’instances très occultées.

**Score de vraisemblance** Dans de nombreux cas, il est nécessaire de se ramener à une mesure unique quantifiant la vraisemblance d’une hypothèse de pose relativement aux données, notamment afin de pouvoir ordonner celles-ci. Dans le cadre de notre implémentation, nous fusionnons pour se faire les différentes mesures proposées de manière naïve en une unique mesure définie entre 0 et 1 :

$$\text{likelihoodRatio} = \text{matchingRatio} \cdot (1 - \text{inconsistencyRatio}) \cdot \text{edgesMatchingRatio}. \quad (4.26)$$

Ce choix est relativement arbitraire. La fusion des différentes mesures pourrait sans doute être améliorée afin d’optimiser la séparation de bonnes hypothèses et de faux positifs, notamment en s’appuyant sur des exemples annotés. La mesure retenue présente néanmoins l’avantage de la simplicité.

### 4.5.2 Cohérence globale

Dans un souci de simplicité d'utilisation de notre méthode pour une application de robotique industrielle, nous choisissons de ne retourner qu'une unique interprétation de la scène, sous forme d'une liste d'hypothèses de pose compatibles entre elles, correspondant à différentes instances d'objets détectées. Nous considérons ici des hypothèses de pose comme compatibles entre elles si des instances d'objets en ces poses ne s'intersectent pas.

Choisir la meilleure explication de scène à partir des hypothèses de poses vraisemblables dont nous disposons est un problème combinatoire de complexité exponentielle, aussi nous utilisons une approche gloutonne afin de trouver une bonne solution à ce problème. Étant donnée les différentes hypothèses de pose triées par vraisemblance décroissante (cf. section 4.5.1), nous retenons la première (la plus vraisemblable relativement aux données) dans notre liste de poses finales. Nous testons alors itérativement si l'hypothèse suivante est compatible avec celles précédemment sélectionnées et si oui l'ajoutons à la liste des résultats.

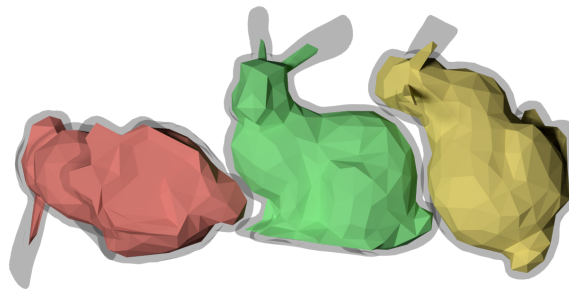
Nous nous appuyons pour cela sur la bibliothèque *Bullet Physics* (Coulmans, 2017), un moteur de simulation physique libre, afin de réaliser les tests d'intersection. L'objet est modélisé sous forme d'un maillage triangulaire, et le test d'intersection entre deux instances consiste simplement en le test de l'existence d'au moins une intersection entre les triangles de ces derniers<sup>2</sup>. Afin d'être robuste aux imprécisions d'estimation de pose conduisant à ce que deux hypothèses bien que valides aient des modèles qui s'intersectent légèrement, nous considérons durant cette phase un modèle ne correspondant pas exactement à la géométrie de l'objet. Dans le cas d'un objet *plein*, il est en effet pertinent d'utiliser l'érosion morphologique 3D du modèle de l'objet initial suivant un noyau de dimension  $d$  de manière à pouvoir réaliser des test d'intersection ayant une tolérance de pénétration de  $2d$  (voir exemple figure 4.13a). Cette approche atteint néanmoins ses limites avec des objets présentant des zones de faible épaisseur tel que le cylindre creux figure 4.13b dont l'érosion peut rapidement se ramener à l'ensemble vide. Il convient en ce cas d'utiliser un modèle spécifique, tel qu'un cylindre plein de plus petite dimension dans notre exemple.

## 4.6 Apprentissage d'un arbre de décision

Notre approche se base sur l'utilisation d'arbres de décision capables étant donné un point de référence  $x$  et une image de profondeur, de produire une estimation de la pose de l'instance à laquelle appartient  $x$  sous forme d'une distribution de pose. Cette capacité n'est pas innée mais est le fruit d'un apprentissage automatique, dont nous abordons les aspects dans cette section.

---

2. Du fait que les deux instances possèdent la même géométrie (il s'agit du même objet), il est impossible que l'une des instances soient à l'intérieur de l'autre sans que leurs surfaces (c.-à-d. leurs triangles) ne s'intersectent, ce qui garantit l'exactitude du test.



(a) Modèle érodé (en rouge, vert et jaune) de l'objet initial (transparent).



(b) Modèle spécifique (en vert) dans le cas d'un objet présentant des zones de faible épaisseurs (à gauche).

FIGURE 4.13 – Cohérence globale d'une explication de scène. On sélectionne les hypothèses de pose renvoyées de manière à ce qu'elles soient compatibles entre elles – c.-à-d. ne s'intersectent pas. Afin d'être robuste aux petites inexactitudes de pose, on procède aux tests d'intersection à l'aide d'un modèle plus *petit* que l'objet lui-même.

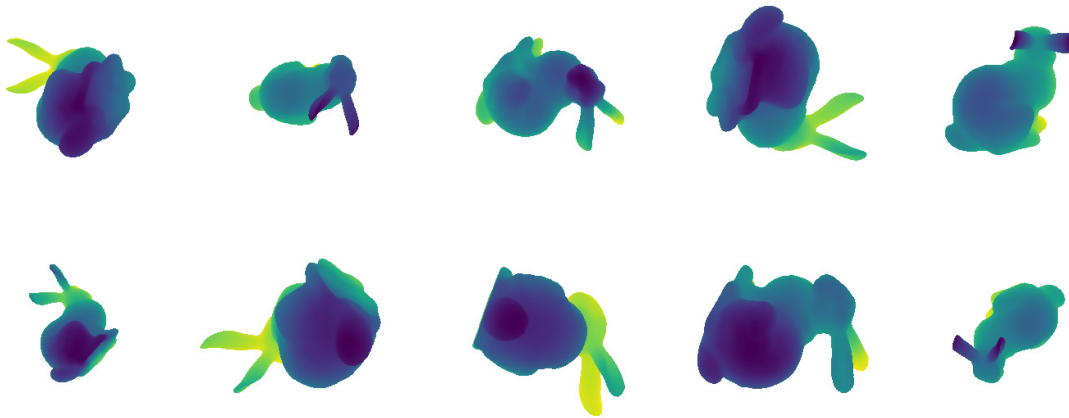


FIGURE 4.14 – Données d’apprentissage : exemple de rendus de profondeur synthétiques générés suivant une distribution d’orientation uniforme.

#### 4.6.1 Données d’apprentissage

Nous réalisons l’apprentissage d’un arbre de décision de manière supervisée, au moyen d’images de profondeur d’instances d’objet dont nous connaissons la pose.

Nous utilisons pour cela des rendus synthétiques correspondant à des vues de l’objet suivant différentes poses, réalisés par rendu rasterisé d’un modèle 3D de l’objet à l’aide de la bibliothèque OpenGL. La figure 4.14 fournit un exemple de telles données d’apprentissage. L’usage de données synthétiques permet de pallier simplement à la difficulté d’annotation de données réelles avec la pose des instances.

Nous utilisons des données de profondeur idéales dépourvues de bruit lors de l’apprentissage. Bien qu’il soit possible de générer des images de synthèse présentant un bruit réaliste comparativement à un capteur donné (cf. section 5.1.2.2), l’usage de données idéales permet de s’affranchir du choix de la technologie de capteur considéré et notre approche s’avère en pratique relativement robuste au bruit des données, même sans avoir fait l’objet d’un apprentissage spécifique (cf. expérimentations section 5.6). Des données idéales s’avèrent de plus rapides à générer<sup>3</sup>, de sorte qu’il est possible de les produire à la volée afin de limiter l’empreinte mémoire nécessaire lors de l’apprentissage.

Pour des raisons similaires, nos images d’apprentissages correspondent à des vues d’une instance d’objet isolé, sans occultation. Les expérimentations de Brachmann et al. (2014) ont mis en avant l’importance du fond dans l’apprentissage d’estimation de poses à partir de carte de profondeur. Ces derniers ont en effet obtenus de meilleures performances en considérant durant l’apprentissage un fond plan en retrait par rapport à l’objet comparé à l’utilisation d’un fond sous forme de bruit blanc. Ces travaux corroborent

3. de l’ordre de 2.5 ms pour une image de profondeur au format VGA à l’aide d’un chipset graphique intégré Intel HD Graphics 4600.

la thèse que la silhouette de l'objet apporte une information importante pour l'estimation de pose, et sont en accord avec nos résultats expérimentaux section 5.4.1. La saturation des descripteurs faibles utilisés dans nos arbres de décision introduite section 4.2.1.2 permet d'introduire une certaine invariance en la profondeur du fond pourvue que celle-ci soit suffisante relativement à l'objet, aussi nous considérons durant l'apprentissage des vues d'une instance d'objet isolée, placé devant un fond de profondeur infinie.

#### 4.6.2 Distribution a priori de pose

Les poses de l'objet relativement à la caméra considérées pendant l'apprentissage devraient idéalement représenter la distribution de probabilité a priori que l'on retrouve dans des données réelles. La caractérisation précise de cette distribution est délicate, en ce que son estimation à partir de données réelles nécessiterait de disposer d'une méthode d'estimation de pose d'objet performante et nous ramène ainsi à un problème du type de *l'œuf et de la poule* hasardeux à résoudre.

La connaissance du modèle 3D de l'objet permet néanmoins d'obtenir certaines informations concernant la distribution typique d'orientation sans disposer de données réelles. Park et al. (2010) dans leurs travaux ont par exemple cherché à identifier celle-ci par le biais de simulations physiques de chute d'instances d'objet en vrac. Si cette approche peut être viable pour certaines applications, elle n'est néanmoins pas générale. La distribution d'orientation de l'objet au sein d'un vrac dépend en effet d'un nombre élevé de paramètres, parmi lesquels :

- La géométrie de l'objet. Un objet présente notamment souvent un petit nombre de faces d'équilibre sur lesquelles il va avoir tendance à reposer lorsqu'il sera déposé sur un fond plat.
- La géométrie du conteneur – notamment celle du fond (plat, incliné, ondulé, etc), et des bords – ainsi que la distance des instances aux parois du conteneur. La géométrie du conteneur influe sur la pose des objets qu'il contient et notamment leur orientation, mais cette influence tend à diminuer avec l'augmentation de la distance aux parois de ce dernier (conteneur de grande dimension devant les dimensions de l'objet, et épaisseur de vrac importante).
- La méthode de dépose des objets dans le conteneur. Les objets peuvent par exemple être déposés suivant une configuration précise pour des raisons de fragilité ou de gain de place, et constituer ainsi un vrac « rangé », ou encore être déposés par un processus spécifique qui va influencer sur la distribution de pose des objets. Un convoyeur faisant tomber des objets dans un bac va ainsi introduire une distribution d'orientation particulière, de la même manière que la hauteur particulière d'une table conduit à faire tomber préférentiellement une tartine sur sa face beurrée (Matthews, 1995, 2001).
- Le transport du conteneur et les secousses qu'il subit. Ces dernières peuvent notamment avoir un effet de relaxation, amenant le contenu du conteneur dans un état d'énergie potentiel typiquement plus faible en ordonnant les instances d'objets de manière plus compacte.

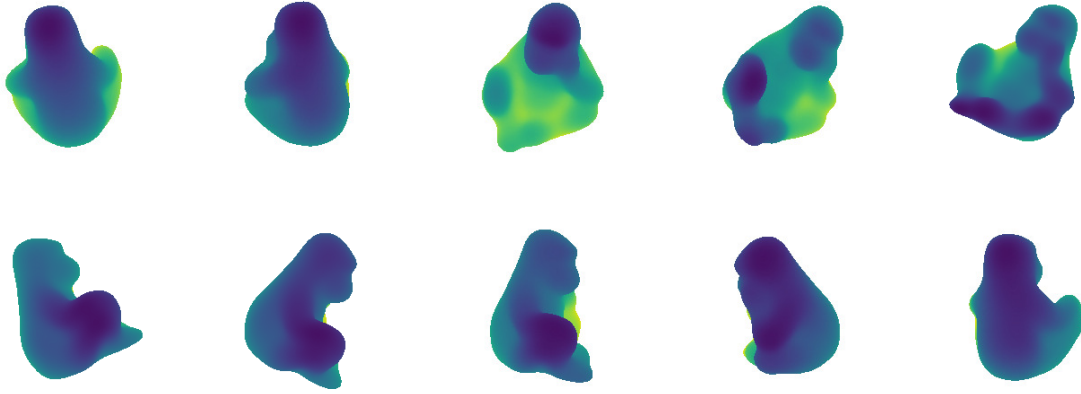


FIGURE 4.15 – Exemple de données d’apprentissage générées avec un a priori concernant la distribution de pose de l’objet relativement à la caméra. Cas de l’objet *Ape* du jeu de données ACCV (Hintertoussier et al., 2012b), posé sur une table et imagé systématiquement la tête vers le haut, selon une vue plongeante.

- De nombreux autres paramètres physiques. On citera notamment les coefficients de frottement statiques et dynamiques, et les propriétés d’amortissement des corps en présence.

Aussi devant la complexité de la tâche et afin de proposer une solution aussi générique que possible, nous procédons au choix des poses d’apprentissage par tirage aléatoire en laissant à l’utilisateur la possibilité d’introduire manuellement un a priori sur la distribution de pose, au moyen d’une paramétrisation simple.

La distance de l’objet à la caméra est ainsi choisie de manière uniforme entre deux constantes définies par l’utilisateur. Ce dernier peut également spécifier une plage d’orientation  $A_{L,U} \subset SO(3)$  privilégiée avec une probabilité  $\alpha \in [0, 1]$ , afin d’exprimer un a priori sur l’orientation de l’objet relativement au repère caméra. Cette plage est définie à partir de bornes inférieures et supérieures  $L, U \in \mathcal{M}_{3 \times 3}(\mathbb{R})$  de la manière suivante :

$$A_{L,U} = \left\{ \mathbf{R} \in SO(3) \mid \forall (i, j) \in \llbracket 1, 3 \rrbracket^2, L_{i,j} \leq \mathbf{R}_{i,j} \leq U_{i,j} \right\} \quad (4.27)$$

En pratique, cette paramétrisation s’avère suffisante pour exprimer des a priori courants concernant la pose de l’objet relativement à la caméra, comme par exemple celui illustré figure 4.15. Les poses d’apprentissage sont alors tirées de manière aléatoire par la méthode du rejet (*rejection sampling*) à partir d’une distribution uniforme sur l’espace des rotations 3D.

### 4.6.3 Procédure d’apprentissage

Étant donné  $m$  vues de l’objet suivant des poses  $([T_i])_{i=1\dots m}$ , on procède à l’apprentissage d’un arbre de décision de la manière suivante.

Pour chaque vue  $i \in \llbracket 1, m \rrbracket$ , on extrait de l'image de profondeur un ensemble de points de référence suivant la même approche que durant la phase de test (section 4.1.1). Un sous-ensemble  $(x_{i,j})_{j=1\dots n_i}$  de ces points est sélectionné de manière aléatoire (50% typiquement) afin d'être utilisés comme exemples d'apprentissage. Ceux-ci sont affectés au nœud racine de l'arbre de décision, sous forme d'une collection de vues de l'objet et de points de référence associés  $([T_i], x_{i,j})_{i=1\dots m, j=1\dots n_i}$ .

L'apprentissage de l'arbre est alors réalisé de manière itérative et gloutonne, niveau par niveau. Pour un niveau de profondeur de l'arbre donné, nous tirons aléatoirement un nombre  $a \in \mathbb{N}^*$  de paramètres  $(\theta_k)_{k=1\dots a}$  définissant chacun un descripteur faible  $g_k \triangleq g(\cdot, \cdot, \theta_k)$  (cf. section 4.2.1). Nous évaluons alors ces différents descripteurs pour chacun des exemples d'apprentissage. Les images de profondeur correspondantes à chacune des poses  $([T_i])_{i=1\dots m}$  sont pour se faire générées à la volée afin de ne pas avoir à les stocker en mémoire.  $m = 50000$  images distinctes sont en effet typiquement utilisées pour l'apprentissage d'un arbre de décision, ce qui représente un volume de données conséquent<sup>4</sup>.

Pour chacun des nœuds actifs de la profondeur courante, nous générons des classifieurs faibles  $h_{k,l}(\cdot, \cdot) \triangleq [g_k(\cdot, \cdot) < \tau_{k,l}]$  par comparaison de la sortie de chaque descripteur  $g_k$  avec  $b \in \mathbb{N}^*$  différents seuils  $(\tau_{k,l})_{l=1\dots b}$ , pour  $k = 1, \dots, a$ . Ces seuils sont choisis par tirage aléatoire dans l'intervalle de valeurs que prennent les descripteurs, et nous évaluons la manière dont les classifieurs générés permettent de scinder en deux l'ensemble des échantillons d'apprentissage associés au nœud. Le classifieur permettant la meilleure séparation des exemples d'apprentissage – au sens d'un critère de gain d'information défini section 4.6.4 – est alors retenu, et on affecte ce classifieur au nœud courant. Deux nœuds enfants sont alors créés et les échantillons du nœud courant sont affectés à ces derniers, suivant la sortie du classifieur. Dans le cas où la valeur du classifieur serait non définie pour un échantillon du fait de l'absence d'information de profondeur en certains pixels, l'échantillon est alors affecté aléatoirement à un des nœuds enfants de manière équiprobable.

Nous itérons ensuite de manière à traiter le niveau de profondeur suivant de l'arbre. Nous arrêtons de scinder les nœuds lorsqu'une profondeur maximale de l'arbre est atteinte, ou lorsque le nombre d'échantillons affectés à ceux-ci deviennent inférieur à un seuil donné. Dans le cadre de nos expériences, nous considérons typiquement une profondeur maximale de 20, et évaluons pour chaque nœud de l'arbre  $a = 30$  descripteurs associés à  $b = 5$  seuils différents, tant que ce nœud contient plus de 10 échantillons. Le choix de ces valeurs représente un compromis raisonnable entre :

- les performances, qui augmentent typiquement avec la profondeur des arbres et le nombre de classifieurs testés lors de l'apprentissage (cf. section 5.4).
- la consommation mémoire, linéaire en le nombre d'échantillons et exponentielle en la profondeur des arbres. Une profondeur de 20 conduit typiquement à  $2^{20} \approx 10^6$  feuilles, ce qui correspond à une

4. Environ 60 Go de données brutes dans le cas d'images VGA stockées sous forme de tableau de valeurs flottantes 32 bits, bien que ce volume puisse sans doute être largement compressé.



consommation mémoire d'environ 100Mo par arbre suite à l'étape de compression décrite plus bas section 4.6.5.

- la durée d'apprentissage, typiquement linéaire en la profondeur des arbres et le nombre de classifieurs faibles testés. Avec ces paramètres, celle-ci est de l'ordre de 1h40 par arbre sur un ordinateur de bureau standard équipé d'un processeur Intel Core i7-4790 @ 3.6GHz.

#### 4.6.4 Critère de sélection d'un classifieur faible

Dans cette section nous définissons une fonction objectif permettant de comparer la pertinence des classifieurs faibles testés lors de l'apprentissage, de sorte à pouvoir sélectionner le meilleur, au sens de cette fonction. Cet objectif est basé sur la notion de gain d'information, s'appuyant sur la notion d'entropie, au sens de Shannon.

##### 4.6.4.1 Gain d'information

La notion de gain d'information  $I$  est une notion classique en induction d'arbre de décision, à la base de l'algorithme ID3 (Quinlan, 1986). En notant  $S$  la distribution de pose représentée par les échantillons d'apprentissage atteignant le nœud courant, et  $S_L$  et  $S_R$  les distributions associées aux deux nœuds enfants, obtenues après classification des échantillons de  $S$  en deux classes par un classifieur faible, le gain d'information est défini à partir de l'entropie de Shannon  $H$  comme suit :

$$I = H(S) - \sum_{i \in L,R} \frac{|S_i|}{|S_L| + |S_R|} H(S_i). \quad (4.28)$$

Malgré son nom, cette notion quantifie en toute rigueur l'espérance de perte d'information au sens de Shannon (c.-à-d. d'entropie) qu'apporte le classifieur, et correspond à la différence entre l'espérance d'information  $H(S)$  contenue dans la distribution initiale, et l'espérance d'information moyenne  $\sum_{i \in L,R} |S_i| / (|S_L| + |S_R|) H(S_i)$  après la classification du nœud. Dans le cas de la régression de pose, un gain d'information  $I$  positif correspond à un gain de certitude sur la pose de l'objet, ce qui au sens de la théorie de l'information correspond à une diminution du caractère informatif de la pose en tant que variable aléatoire.

Nous basons notre procédure d'apprentissage sur ce critère, et étant donné plusieurs classifieurs faibles, choisissons celui apportant le gain d'information le plus grand. Fanelli et al. (2011); Tejani et al. (2014) considèrent également un tel critère et utilisent pour se faire un estimateur d'entropie supposant une distribution gaussienne des paramétrisations de poses basées sur des angles d'Euler. Cette approche revient à chercher à minimiser la variance des distributions enfants dans cet espace de paramètres, et présente certaines limitations :

- Un estimateur paramétrique gaussien n'est pas en mesure de quantifier correctement l'entropie d'une distribution multimodale, ce qui peut conduire à des choix sous-optimaux de classifieurs ainsi que l'illustre la figure 4.16.

- La paramétrisation à base d'angles d'Euler n'induit pas une mesure sur l'espace de poses objective (c.-à-d. invariante en les repères considérés, voir chapitre 3), ce qui peut être à l'origine d'artefacts.
- Cette approche ne tient pas en compte des symétries des objets.

Afin de dépasser ces limitations, nous préférons quant à nous considérer un estimateur d'entropie plus spécifique, adapté aux propriétés de l'espace de pose.

#### 4.6.4.2 Estimateur non paramétrique d'entropie

L'espace de pose défini chapitre 3 n'est pas un espace Euclidien et dès lors les estimateurs paramétriques d'entropie usuels ne sont pas des plus adaptés pour ce dernier. Nous considérons donc un estimateur d'entropie non paramétrique, défini à partir de la distance du  $k$ -ième plus proche voisin et proposé par Singh et al. (2003) d'après les travaux originaux de Kozačenko et Leonenko (1987). Étant donné une distribution définie sur une variété de dimension  $d$  représentée par un tirage aléatoire de  $n$  échantillons  $\{x_i\}_{i \in \llbracket 1, n \rrbracket}$ , l'entropie de la distribution est estimée par recherche du  $k$ -ième plus proche voisin comme suit :

$$\widehat{H}_k = \frac{d}{n} \sum_{i=1..n} \log(\rho_{i,k,n}) + \log \left( \frac{\pi^{d/2}}{\Gamma(d/2 + 1)} \right) + \gamma - L_{k-1} + \log(n), \quad (4.29)$$

où  $\Gamma$  représente la fonction gamma,  $\gamma$  est la constante d'Euler-Mascheroni valant approximativement 0.577,  $(L_k)_{k \in \mathbb{N}}$  une suite définie par

$$L_0 = 0 \text{ et } \forall j \geq 1, L_j = \sum_{i=1}^j \frac{1}{i} \quad (4.30)$$

et où  $\rho_{i,k,n}$  représente la distance au  $k$ -ième plus proche voisin de l'échantillon  $x_i$  parmi l'ensemble des échantillons  $\{x_j\}_{j \in \llbracket 1, n \rrbracket, j \neq i}$ .

Nous considérons plus particulièrement une estimation à partir du plus proche voisin ( $k = 1$ ), dont l'expression se réduit à

$$\widehat{H}_1 = \frac{d}{n} \sum_{i=1..n} \log(\rho_{i,1,n}) + \log \left( \frac{\pi^{d/2}}{\Gamma(d/2 + 1)} \right) + \gamma + \log(n). \quad (4.31)$$

L'usage d'un estimateur non paramétrique tel que ce dernier est intéressant pour l'apprentissage d'arbres de décision, en ce qu'il ne fait pas d'hypothèses fortes sur la forme de la distribution de probabilité considérée et est à même d'appréhender convenablement des distributions multimodales, qui peuvent parfois être d'avantage discriminantes que des distributions à un seul mode, comme illustré figure 4.16, où on compare cet estimateur à un estimateur d'entropie paramétrique supposant une distribution gaussienne. Le lecteur intéressé est renvoyé aux travaux de Nowozin (2012) pour une étude comparative de différentes fonctions objectif pour l'induction d'arbre de décision.

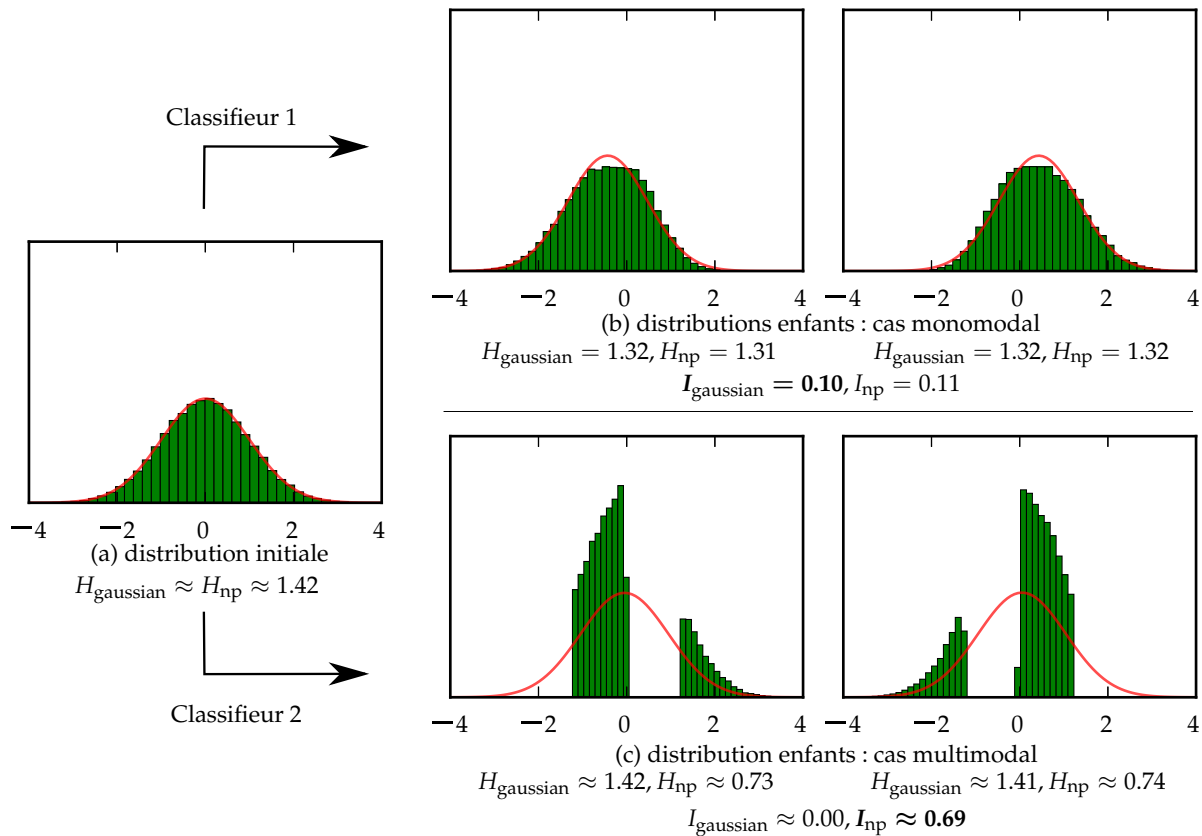
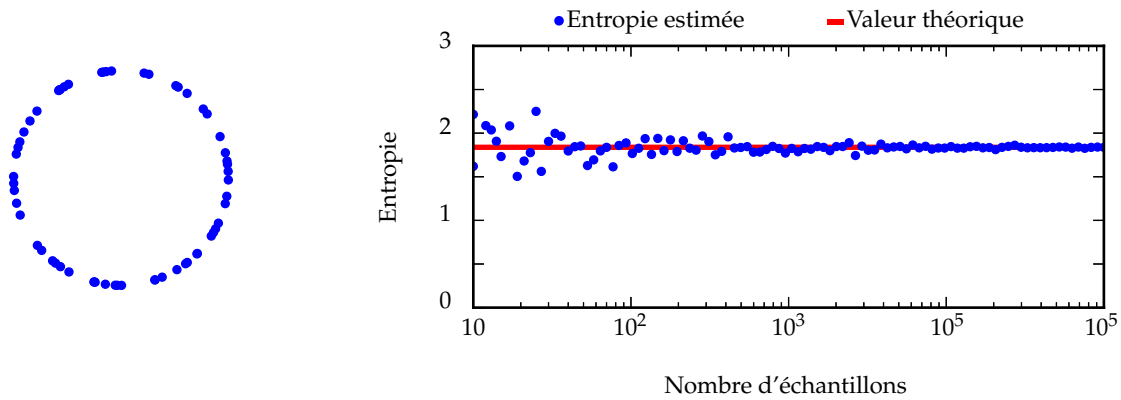


FIGURE 4.16 – Intérêt d’un estimateur d’entropie non paramétrique dans le cas de distributions multimodales (exemple-jouet). On cherche à comparer les performances de deux classificateurs binaires pour discriminer parmi un ensemble de données. Les classificateurs permettent de scinder un ensemble d’échantillons (histogramme de la distribution initiale à gauche) en deux (distributions à droite, respectivement en haut et en bas). Le pouvoir discriminant de ces classificateurs est mesuré à l’aide du critère de gain d’information, calculé à partir d’une estimation de l’entropie des distributions. Contrairement à l’estimateur non paramétrique  $H_{\text{np}}$  défini équation (4.31), un estimateur paramétrique supposant une distribution gaussienne  $H_{\text{gaussien}}$  introduit un biais conséquent dans l’estimation d’entropie dès lors que le modèle de distribution supposé n’est pas valide (distributions (c), en bas). Celui-ci conduit à une estimation biaisée du gain d’information  $I_{\text{gaussien}}$  du classificateur. Dans notre exemple, l’estimateur paramétrique attribue ainsi un gain d’information  $I_{\text{gaussien}}$  supérieur pour le classificateur 1 comparé au classificateur 2, à l’opposé de l’estimateur non paramétrique  $I_{\text{np}}$  qui privilégie le classificateur 2 qui apporte effectivement d’avantage de certitude sur la valeur des données, en conduisant à des distributions moins étalées. En vert, histogramme des distributions. En rouge, fonction de densité d’une loi normale ajustée aux données, considérée par l’estimation paramétrique.



(a) Tirage aléatoire selon une distribution uniforme.

(b) Estimation de l'entropie de la distribution à partir d'un tirage aléatoire, convergeant vers la valeur théorique  $\log(2\pi)$ .

FIGURE 4.17 – Estimation non-paramétrique d'entropie d'une distribution sur une variété : illustration avec le cas d'un cercle.

**Travail sur une variété topologique** L'estimateur (4.31) ne dépend que d'une estimation de distance entre échantillons, et peut ainsi être utilisé sur des distributions de probabilité définies sur des variétés telles que notre espace de pose. Nous illustrons cette possibilité figure 4.17 avec le cas d'une distribution uniforme définie sur un cercle unité, variété de dimension 1 plongée dans un espace de dimension 2. Cette distribution présente une densité de probabilité uniforme de valeur  $p = 1/(2\pi)$  sur tout le cercle, à partir de laquelle la valeur théorique de l'entropie de celle-ci peut être estimée

$$H_{\text{théorique}} = - \int_0^{2\pi} p \log(p) d\theta = \log(2\pi). \quad (4.32)$$

Des expérimentations numériques, basées sur des tirages aléatoires de cette distribution illustrent la bonne convergence de l'estimateur 4.31 vers cette valeur théorique, en considérant la distance de l'espace ambiant (distance euclidienne 2D) entre points.

Afin d'utiliser l'estimateur 4.31 dans notre cas d'application, il est nécessaire de spécifier la dimension  $d$  de la variété sur laquelle sont définis les exemples d'apprentissage. Celle-ci diffère de la dimension de l'espace de pose, car les exemples d'apprentissage correspondent à des poses exprimées relativement à un point de référence à la surface de l'objet, et sont donc définies sur une sous-variété de l'espace de pose, ayant une dimension de moins que l'espace initial (correspondant à la perte d'un degré de liberté en translation). Le tableau 4.18 synthétise les valeurs de  $d$  suivant la classe de symétrie de l'objet.

#### 4.6.4.3 Limitations

L'estimateur d'entropie non paramétrique précédent présente de nombreux avantages, cependant celui-ci possède néanmoins certaines limitations

Classe de symétrie propre	Dimension de l'espace de pose	Dimension $d$ de la variété
Sphérique	3	2
De révolution (non sphérique)	5	4
Groupe de symétrie propre fini	6	5

TABLE 4.18 – Dimensions de la variété sur laquelle sont défini les exemples d'apprentissage suivant la classe de symétrie propre de l'objet.

que nous souhaitons évoquer ici.

**Fléau de la dimension** La dimension "importante" de l'espace de travail conduit à ce que les échantillons de pose considérés lors de l'apprentissage soient relativement épars, et distants les uns des autres. Ce phénomène, connu sous le nom de *fléau de la dimension*<sup>5</sup>, rend l'estimation de l'entropie bruitée à moins de considérer un très grand nombre d'échantillons. En considérant un exemple-jouet constitué d'une distribution uniforme sur une variété similaire à celle que l'on pourrait rencontrer durant l'apprentissage d'un objet de révolution, on observe en effet expérimentalement qu'il faut plusieurs milliers d'échantillons avant que l'estimation d'entropie soit robuste – c.-à-d. ait une variance relativement limitée – vis-à-vis du tirage aléatoire considéré (cf. figure 4.19). Afin de limiter la durée d'apprentissage, l'estimation de l'entropie est cependant réalisée en pratique à partir d'un nombre limité d'échantillons tirés aléatoirement (typiquement 1000). De plus, il arrive fréquemment que l'on ne dispose pas d'autant d'exemples d'apprentissage, tout particulièrement pour les couches les plus profondes de l'arbre où seuls une dizaine d'échantillons atteignent typiquement chaque nœud. Ces échantillons ne sont en ce cas pas représentatifs de la distribution de pose *vraie*, ce qui conduit à un phénomène de sur-apprentissage, et participe au caractère aléatoire des arbres de décisions appris. Le fait de considérer une forêt d'arbres, ainsi que de fusionner les votes locaux (section 4.2.2), permet de limiter ce phénomène.

**Biais d'estimation** Bien que la distance entre poses considérée ici soit localement équivalente à une distance géodésique, cette dernière est systématiquement sous évaluée par notre distance, de la même manière que le segment de droite reliant deux points à la surface d'une sphère est plus court que le plus petit arc reliant ces points à la surface de la sphère. Aussi, l'estimateur (4.31), bien qu'asymptotiquement non biaisé, possède un biais tendant à surestimer l'entropie, et ce d'autant plus que le nombre d'échantillons est faible (et donc que les distances entre échantillons sont importantes), comme on l'observe figure 4.19. Ce biais n'est néanmoins pas problématique pour notre apprentissage dans lequel nous sommes moins intéressés par l'exactitude de l'estimation d'entropie sur une variété topologique, que par une mesure permettant de quantifier le caractère *compact* ou *étalé* d'une distribution sur l'espace de pose, ce que traduit bien l'estimateur (4.31).

5. *Curse of dimensionality* en anglais, expression attribuée à Richard Bellman.

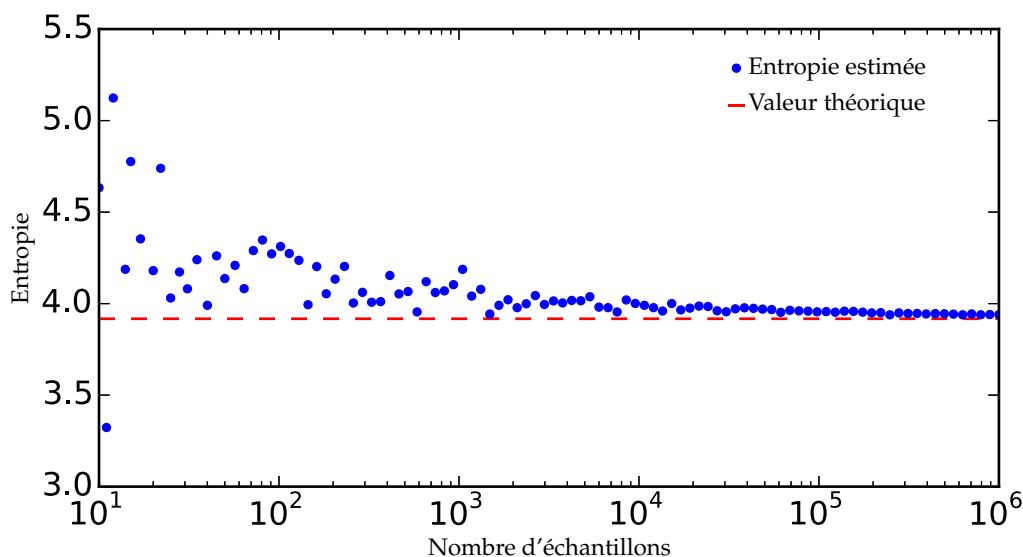


FIGURE 4.19 – Biais de l'estimateur non paramétrique d'entropie. Exemple-jouet avec une distribution de probabilité uniforme sur  $[-1, 1]^2 \times \{0\} \times SO(3) \subset \mathbb{R}^6$ .

#### 4.6.5 Représentation synthétique d'un apprentissage

À l'issue de l'apprentissage, l'ensemble des exemples d'apprentissage sont répartis entre les feuilles de l'arbre, et représentent pour chaque feuille la connaissance que l'on a de la distribution de pose de l'objet relativement au repère de référence, à la suite des classifications effectuées le long des branches de l'arbre de décision. Un inconvénient majeur de cette représentation est son besoin en mémoire important, du fait qu'elle nécessite de conserver l'ensemble des exemples d'apprentissage. À titre d'exemple, nous représentons la pose d'un objet sans invariance par un vecteur de dimension 12 encodé en valeurs flottantes 32 bits, aussi le stockage de  $10^7$  exemples d'apprentissage (ordre de grandeur utilisé pour un arbre) nécessite 480 Mo. Pourvu que l'apprentissage ait été fructueux, il n'est cependant pas nécessaire de conserver l'ensemble de ces derniers, car l'a priori de pose est alors important sachant quelle feuille de l'arbre de décision a été atteinte.

Afin d'obtenir une représentation plus synthétique et suivant une démarche similaire à celle de [Shotton et al. \(2013a\)](#), nous choisissons donc de *compresser* cette dernière en ne représentant la distribution de pose en chaque feuille que par les  $k \in \mathbb{N}^*$  modes principaux de la distribution, associés à des poids correspondant à leur pseudo-densité de probabilité. Nous procédons pour ce faire à une étape de Mean Shift de manière à détecter les modes de densité, suivie d'une estimation de la pseudo-densité de probabilité en ces modes pour quantifier l'importance de ces dernières, et ne conservons que les  $k$  principaux. Les techniques employées pour ces étapes sont les mêmes que lors de la phase de test décrites section 4.3.

En pratique, nous choisissons typiquement  $k = 2$  car cette valeur permet

TABLE 4.20 – Influence de la compression d’un apprentissage sur les performances en test. Le stockage d’un nombre limité d’échantillons par feuille (en l’occurrence 2) suffit à obtenir des performances aussi bonnes sinon meilleures qu’avec l’apprentissage complet.

	Nombre d’échantillons stockés par feuille			
	1	2	3	5
Gain relatif de performance	$-0.35\% \pm 0.07\%$	$+0.08\% \pm 0.05\%$	$+0.12\% \pm 0.03\%$	$+0.07\% \pm 0.02\%$

Performances quantifiées en terme de Précision Moyenne pour la détection des instances moins de 50% occultées. Évaluations menées sur le jeu de données *bunny* avec 5 apprentissages indépendants de 5 arbres de profondeur maximale 20. Le lecteur est renvoyé au chapitre 5 pour une description du protocole utilisé.

une compression efficace – en divisant typiquement par 5 la taille d’un apprentissage – tout en étant suffisante afin d’obtenir des performances similaires au cas sans compression (cf. tableau 4.20). Hormis la question de l’impact mémoire, cette compression a également pour avantage de borner et de réduire significativement le nombre de votes générés pour un point de référence, ce qui permet d’accélérer l’extraction d’hypothèses de pose durant la phase de test.

## 4.7 Synthèse

Dans ce chapitre, nous avons introduit une nouvelle approche de détection et d’estimation de pose d’instances d’objet rigide à partir d’une image de profondeur. Celle-ci est formalisée sous forme d’un problème de régression probabiliste local, où il s’agit pour un point 3D de référence de la scène de prédire la pose de l’instance d’objet auquel ce point appartient. Cette prédiction est réalisée au moyen d’une forêt de décision, apprise à partir de données synthétiques et suivant un critère de gain d’information adapté à la topologie de l’espace de pose de l’objet considéré. En agrégeant les prédictions d’un ensemble de tels points de référence échantillonnés de manière à couvrir l’ensemble de la scène, on obtient une distribution représentant la probabilité qu’une instance d’objet soit présente dans une pose donnée. On extrait de cette distribution un ensemble d’hypothèses de poses par une procédure de Mean Shift, et une étape de raffinement et de vérification permet de filtrer ces dernières avant de retourner les résultats.

Le chapitre suivant complète cette présentation descriptive de notre approche par une étude plus quantitative et expérimentale de ses capacités. Ainsi qu’on le verra, celle-ci s’avère relativement performante et robuste dans le cadre de scénarios de dévracage. L’applicabilité de nos travaux sur la notion de pose ne se limite cependant pas à la méthode proposée ici, et de futurs travaux pourraient notamment explorer l’utilisation de notre métrique pour l’entraînement de méthodes de détection et d’estimation de pose par apprentissage profond.

## Chapitre 5

# Évaluation expérimentale

*The view that machines cannot give rise to surprises is due, I believe, to a fallacy to which philosophers and mathematicians are particularly subject. This is the assumption that as soon as a fact is presented to a mind all consequences of that fact spring into the mind simultaneously with it. It is a very useful assumption under many circumstances, but one too easily forgets that it is false. A natural consequence of doing so is that one then assumes that there is no virtue in the mere working out of consequences from data and general principles.*

– Alan Turing, *Computing machinery and intelligence*.

---

5.1	Méthodologie d'évaluation . . . . .	142
5.2	Expérimentations sur différents jeux de données de vrac . . .	160
5.3	Expérimentations sur le jeu de données LINEMOD . . . . .	178
5.4	Caractérisation de notre forêt de décision . . . . .	182
5.5	Influence de l'occultation . . . . .	189
5.6	Robustesse au bruit . . . . .	190
5.7	Synthèse . . . . .	192
6.1	Résumé . . . . .	196
6.2	Contributions . . . . .	197
A	Méthodes de calcul pour un maillage triangulaire . . . . .	199
B	Simplification de l'expression de la distance proposée dans le cas d'un objet de révolution sans invariance par rotoréflexion	200
C	Condition de cohérence pour un doublet de représentants . . .	201
D	Détail de calcul : distance minimale entre représentants . . .	202
E	Robustesse au bruit . . . . .	203

---

Le chapitre précédent introduit une nouvelle approche de détection et d'estimation de pose. Celle-ci se veut raisonnablement performante vis-à-vis de l'existant – tout particulièrement dans le cas de scènes de vrac – et adaptée à des objets de forme arbitraire grâce à une prise en compte de la géométrie de l'objet et notamment de ses symétries. Il convient donc de tester ces hypothèses de manière expérimentale, ce qui est l'objet de ce chapitre.

Nous commençons pour se faire par définir une méthodologie avec laquelle mener nos évaluations section 5.1. Nous mettons celle-ci en œuvre



section 5.2 sur un jeu de données que nous avons spécialement conçu pour représenter notre problématique, puis comparons section 5.3 notre approche à l'état de l'art du domaine sur le jeu de données LINEMOD largement utilisé dans la littérature. Enfin, les sections 5.4 et suivantes ont pour objet de caractériser certaines propriétés de notre solution, de manière à éclairer notre compréhension de celle-ci ainsi que de certains choix de conception.

## 5.1 Méthodologie d'évaluation

Si un nombre conséquent de méthodes de détection et d'estimation de pose existent, les performances de celles-ci dans notre contexte industriel ne sont que peu connues. Les industriels sont en effet peu enclins à diffuser des informations quant aux performances de leurs solutions, et les approches proposées dans la littérature académique ne sont pas des plus adaptées à cette problématique<sup>1</sup>.

Afin d'essayer de combler ces lacunes, nous abordons dans cette section la question de l'évaluation d'une méthode de détection et d'estimation de pose d'instances d'objet. Nous commençons dans la sous-section 5.1.1 par dresser un panorama de l'existant dans ce domaine, et forts de nos observations, nous tâchons de remédier à ses limitations principales. Nous proposons pour ce faire sous-section 5.1.2 deux solutions permettant de générer facilement des jeux de données annotés de scènes complexes, et nous formalisons dans la sous-section 5.1.3 une méthodologie d'évaluation des performances.

Bien que nous ne considérons ici que le cas de scènes où figurent plusieurs instances d'un même objet (il s'agit de notre cas d'étude), la méthodologie introduite se veut générale et des scènes où plusieurs objets sont présents pourraient être traitées de manière similaire, pourvu qu'aucune relation particulière entre ces derniers n'aient à être considérée, notamment sous forme de catégories<sup>2</sup>.

### 5.1.1 Les approches d'évaluation existantes et leurs limitations

Les techniques d'évaluation existantes du problème de détection et d'estimation de pose d'objet 3D peuvent être classées en deux catégories principales : les approches expérimentales en ligne, où il s'agit d'évaluer les performances d'une méthode dans l'exécution d'une certaine tâche, et les approches hors ligne, basées sur l'utilisation de jeux de données annotés. Cette seconde catégorie est plus propice à des évaluations comparatives de par la facilité de répétabilité qu'elle offre. Cependant les jeux de données existants présentent certaines limitations les rendant peu représentatifs de scénarios tels que celui du dévracage, et les mesures de performance utilisées avec ces derniers s'avèrent inadaptées au cas de scènes représentant de nombreuses instances d'objets potentiellement symétriques.

1. La récente publication par l'entreprise MVTec (Drost et al., 2017) d'un jeu de données plus adapté à des problématiques de vision industrielle, simultanément à la publication de nos propres travaux (Brégier et al., 2017a) qui sont décrits ici, pourrait amorcer un changement dans ce domaine.

2. Par exemple une catégorie *chaise* et une catégorie *chaussure* regroupant chacune plusieurs objets distincts (chaussure de sport, ballerine, etc.).

### 5.1.1.1 Évaluation en ligne

Nous qualifions de *en ligne* une approche où il s'agit d'évaluer les performances d'exécution d'une tâche complète, typiquement de manipulation robotisée d'objet, intégrant une étape de détection et d'estimation de pose d'instances d'objet. Une telle approche est utile pour démontrer et quantifier l'applicabilité d'une solution d'estimation de pose à une tâche donnée, et plusieurs auteurs proposent donc des montages expérimentaux de préhension d'objet à l'aide d'un robot (Horn et Ikeuchi, 1983; Buchholz et al., 2010; Rodrigues et al., 2012). Il s'agit également d'une approche assez naturelle dans l'industrie où il est essentiel de connaître les performances d'une machine, et qui peut être réalisée assez simplement quoique de manière fastidieuse en comptant le nombre de succès et d'échecs dans l'exécution de la tâche souhaitée de manière à produire des informations statistiques adéquates.

Il est cependant difficile de désintriquer l'estimation de pose d'objets des différents autres aspects à l'œuvre dans ces évaluations : choix du point de prise, planification de trajectoire, préhension et manipulation, etc. Ces expériences ne peuvent de plus évaluer avec fiabilité que la capacité à localiser une unique instance d'objet prenable à la fois, du fait du risque de perturber la pose des instances d'objet dans la scène dès lors qu'il y a manipulation. Enfin, il est difficile d'évaluer l'exactitude d'estimation de pose lors d'expérimentations réelles, du fait du manque d'une vérité terrain. Liu et al. (2012) contournent cette difficulté en considérant plusieurs acquisitions d'un même objet suivant différents points de vues connus, et évaluent ainsi la cohérence des estimations de pose entre elles par le biais de leur écart à une pose « médiane ». Plus récemment, Abbeles et Goedemé (2016) ont simulé le processus complet de débrassage d'une boîte remplie d'instances d'objets synthétiques et purent ainsi estimer directement l'exactitude des estimations de pose, au détriment du réalisme cependant du fait de l'usage de données synthétiques relativement simplistes.

De manière générale, la reproductibilité des évaluations en ligne demeure un problème, et des comparaisons de performance ne peuvent dès lors être effectuées que sur un plan statistique.

### 5.1.1.2 Évaluation hors ligne

**Jeux de données** L'utilisation de jeux de données annotés avec la pose des instances d'objets visibles permet une évaluation plus objective et reproductible, de manière *hors ligne*, et facilite la comparaison de méthodes entre elles. Au fil des années, un certain nombre de jeux de données annotés ont été proposés et nous synthétisons certaines caractéristiques des principaux ayant trait à notre problématique dans le tableau 5.1. Nous ne considérons cependant pas ces derniers suffisamment pertinents pour l'évaluation de notre problématique de débrassage, et c'est pourquoi nous développons nos propres jeux de données, suivant la méthodologie décrite section 5.1.2.

En effet, à l'exception de T-LESS (Hodañ et al., 2017) et du récent MVTEC ITODD (Drost et al., 2017), les jeux de données existants consistent en des acquisitions de scènes où figurent des objets reposant sur une face donnée sur une surface horizontale (typiquement une table), imagées en tournant

TABLE 5.1 – Caractéristiques des principaux jeux de données de détection et d’estimation de pose d’objets rigides utilisés dans la littérature. Suivant la définition de Hodan et al. (2016), nous distinguons le problème de *localisation*, qui consiste en l’estimation de la pose d’un nombre connu d’instances de chaque objet, du problème de *détection*, où le nombre d’instances à localiser dans les données est inconnu. Nous qualifions de *redondant* un jeu de données qui contient plusieurs acquisitions de la même scène (suivant des points de vue différents).

Jeu de données	Modalité	Classe de problème	Plusieurs instances	Plusieurs objets	Clutter	Redondance	Variabilité de pose
 Mian et al. (2006)	Nuage de points (scan LASER)	localisation	non	oui	non	non	limitée
 TUW (Aldoma et al., 2014)	RGBD (Kinect)	détection	oui	oui	oui	oui	limitée
 Hinterstoisser et al. (2012b); Brachmann et al. (2014)	RGBD (Kinect)	localisation	non	oui <sup>2</sup>	oui	oui	limitée
 Desk3D (Bonde et al., 2014)	Nuage de points (Kinect, filtré)	détection	non	oui	oui	oui	limitée
 Tejani et al. (2014)	RGBD (Kinect)	localisation <sup>1</sup>	oui	non	oui	oui	limitée
 T-LESS (Hodan et al., 2017)	RGBD (multimodal)	détection	oui	oui	oui	oui	oui

<sup>1</sup> Le nombre d’instances à détecter est constant.

<sup>2</sup> Il s’agit de ne localiser qu’une instance d’un objet spécifique par image dans le cas des annotations originales.

autour de la scène et en variant l'inclinaison du capteur, avec peu de rotation autour de son axe optique. Une telle configuration est pertinente dans le cadre de tâches telles que l'analyse de scènes d'intérieur, cependant elle engendre une variabilité limitée de pose des objets par rapport à la caméra, et n'est donc pas représentative de scénarios d'objets en vrac dans des poses arbitraires.

De plus, l'annotation manuelle de la pose des objets dans chaque scène est une tâche difficile et fastidieuse, aussi les jeux de données comprenant plus d'une dizaine d'images (c.-à-d. tous ceux du tableau 5.1 à l'exception de celui de Mian et al. (2006)) ont recours à des techniques d'annotation automatiques. Pour se faire, les instances d'objets sont typiquement disposées dans une scène statique, et une série d'acquisitions de celle-ci sont produites suivant des points de vues différents. Une fois connu le point de vue de chaque acquisition – par détection dans les données de marqueurs fiduciaires (Hinterstoisser et al., 2012b; Aldoma et al., 2014; Bonde et al., 2014; Tejani et al., 2014; Hodaň et al., 2017) ou en utilisant un dispositif permettant de positionner précisément le capteur lors des acquisitions tel qu'un robot (Aldoma et al., 2014) – il suffit alors d'annoter la pose des instances dans la scène pour pouvoir propager celles-ci dans l'ensemble de la série d'acquisitions. L'annotation demeure cependant une tâche fastidieuse, et Aldoma et al. (2014) proposent de réaliser celle-ci de manière automatique à l'aide d'une méthode de détection et d'estimation de pose d'objets multi-vues, suivie d'une phase de validation manuelle. Ce type d'approche permet d'annoter relativement facilement de larges jeux de données, cependant ceux-ci consistent par construction en de nombreuses acquisitions de quelques scènes suivant des points de vues différents. Les échantillons présentent alors une forte corrélation entre eux ainsi qu'illustré figure 5.2. Un tel jeu de donnée est de fait peu représentatif de la distribution de données d'une application réelle, et il convient d'être prudent lors de son usage vis-à-vis des problèmes de surapprentissage, si une partie de ce dernier est utilisé pour de l'apprentissage ou de l'ajustement de paramètres.

**Exactitude d'une estimation de pose** Une part essentielle de l'évaluation consiste en la sélection d'un critère permettant de décider si une hypothèse de pose correspond ou pas avec une pose de la vérité terrain. Un tel critère est typiquement défini à partir d'une mesure de la similarité de l'hypothèse de pose avec la pose *vraie* annotée, en considérant un seuil fixé. Diverses mesures de similarités ont été utilisées dans la littérature pour cette tâche, mais présentent cependant certaines limitations.

Le ratio *Intersection over Union* des silhouettes des deux poses, projetées sur le plan image est une métrique couramment utilisée dans le domaine de la reconnaissance d'objet ou de la segmentation 2D, et mesure le pourcentage de recouvrement entre deux formes 2D. Ce ratio n'est néanmoins pas capable de distinguer des poses projetant la même silhouette, et n'est donc pas des plus adapté pour comparer les poses d'un objet 3D.

D'autres mesures fondées sur des distances sur l'espace des transformations rigides ont également été employées, telles que l'erreur en translation et en rotation entre les deux poses (Drost et al., 2010) ou le déplacement moyen des sommets d'un modèle 3D de l'objet entre les deux poses (Hinterstoisser

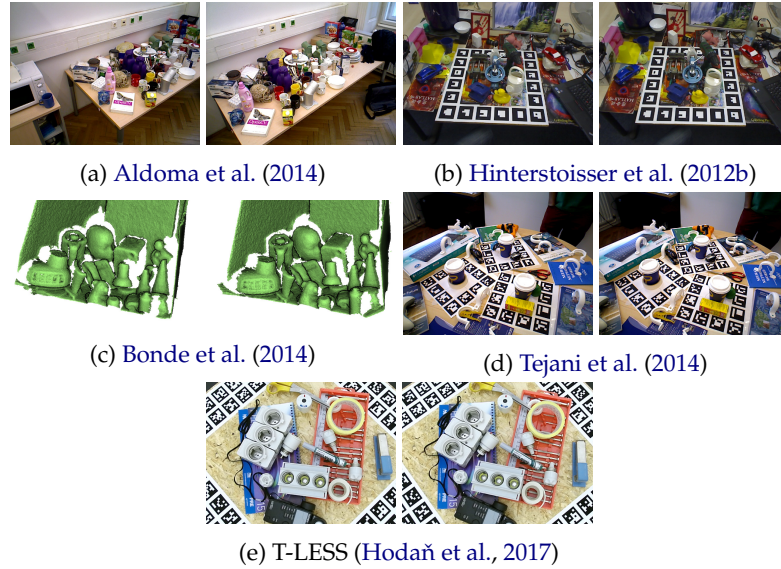


FIGURE 5.2 – Exemples de la corrélation importante entre les échantillons des jeux de données existants. Cette corrélation peut être source d'artefacts d'évaluation importants, particulièrement lorsqu'une partie des données est utilisée à des fins d'apprentissage ou de validation.

et al., 2012b). D'avantage adaptées aux objets 3D, ces mesures ne peuvent néanmoins gérer convenablement les objets présentant des propriétés de symétrie propre. Hinterstoisser et al. (2012b) proposent un contournement de cette difficulté pour les objets symétriques, à l'aide de la mesure de dissimilarité suivante, correspondant à une distance entre deux nuages de points non appariés représentant le modèle  $\mathcal{M}$  de l'objet en les deux poses considérées, paramétrées par des transformations rigides  $T_1, T_2 \in SE(3)$  :

$$\text{moyenne} \left( \min_{x_1 \in \mathcal{M}} \left\| T_1(x_1) - T_2(x_2) \right\| \right). \quad (5.1)$$

Une telle mesure demeure néanmoins problématique en ce qu'elle ne permet pas de distinguer des poses de formes 3D similaires mais pourtant différentes, telles que l'orientation « à l'envers » ou « à l'endroit » d'une boîte pavéoidale texturée asymétriquement (p. ex. une boîte d'emballage quelconque), ou encore dans le cas des gobelets à café de l'exemple figure 5.9a. Hodaň et al. (2016) ont récemment suggéré l'usage d'une mesure d'erreur de pose « invariante aux ambiguïtés », considérant une hypothèse de pose comme valide si et seulement si celle-ci est plausible étant donné les données disponibles. Bien qu'il soit intéressant de prendre en compte les ambiguïtés d'interprétation d'une scène (notamment dans le cadre de la vision active), nous considérons cette approche peu adaptée à la définition d'un protocole d'évaluation, car de nombreuses applications dépendent de la bonne estimation de pose des objets et ne peuvent se satisfaire d'hypothèses plausibles. Cela est d'autant plus vrai lorsqu'il s'agit de considérer des instances fortement occultées, pour lesquelles le nombre de poses plausibles peut être

infiniment grand.

**Précision et rappel** Une fois capable de classifier les hypothèses de poses retournées par un algorithme, on définit alors des métriques permettant de quantifier la performance de ce dernier. Lorsqu'il s'agit d'estimer la pose d'une instance unique d'objet par scène, les performances sont typiquement décrites en terme de *taux de reconnaissance* (Johnson et Hebert, 1999; Drost et al., 2010; Hinterstoisser et al., 2012b), c'est à dire de fraction des scènes du jeu de données pour lesquelles la pose retournée est correcte. Cette métrique est en revanche non pertinente dans le cas de scènes contenant un nombre inconnu d'instances à retrouver, car le taux de reconnaissance ne fournit aucune information sur les faux positifs produits. Dans un tel scénario, la performance est plutôt typiquement décrite en terme de taux de précision et de rappel (Tejani et al., 2014) sur l'ensemble du jeu de données. Ces grandeurs dans leurs définitions classiques traduisent l'objectif de détecter et localiser toutes les instances présentes dans chaque scène sans générer de faux positifs, qui peut cependant ne pas être le véritable objectif recherché dans certaines applications.

Aussi dans la suite, nous nous proposons de dépasser certaines des limitations des jeux de données existants, en proposant une méthodologie permettant de générer facilement des jeux de données annotés représentant de nombreuses instances d'objet dans des poses arbitraires, et constitués de scènes indépendantes, réelles ou synthétiques. Nous suggérons également l'usage de mesures de performance adaptées à des scènes où figurent de nombreux objets, présentant potentiellement des propriétés de symétrie.

## 5.1.2 Génération de jeux de données annotés

Dans cette section, nous proposons deux méthodes pour générer des jeux de données de scènes indépendantes présentant de nombreuses instances d'objet dans des poses arbitraires, automatiquement annotées avec les poses des ces instances. La première méthode est basée sur l'annotation de données réelles au moyens de marqueurs placés sur les instances d'objet, tandis que la seconde s'appuie sur des simulations numériques.

### 5.1.2.1 Jeux de données réel

Nous couvrons la surface de chaque instance d'objet avec un ensemble de marqueurs fiduciaires uniques de manière suffisamment dense afin qu'au moins un marqueur soit toujours visible selon n'importe quel point de vue. La pose d'un marqueur relativement à l'instance auquel il appartient est supposée connue, et dans les expériences que nous avons réalisé nous sommes assurés de cette propriété en sculptant des emplacements pour les marqueurs à la surface d'un modèle 3D de l'objet, à partir duquel nous avons produit les instances par impression 3D (cf. figure 5.3).

Afin de générer des données annotées, les instances d'objet sont disposées en vrac sous un montage stéréoscopique constitué de deux caméras, d'un projecteur de motif pseudo-aléatoire, et d'un éclairage diffus. En éclairant la scène à l'aide du motif texturé, nous réalisons une acquisition sté-

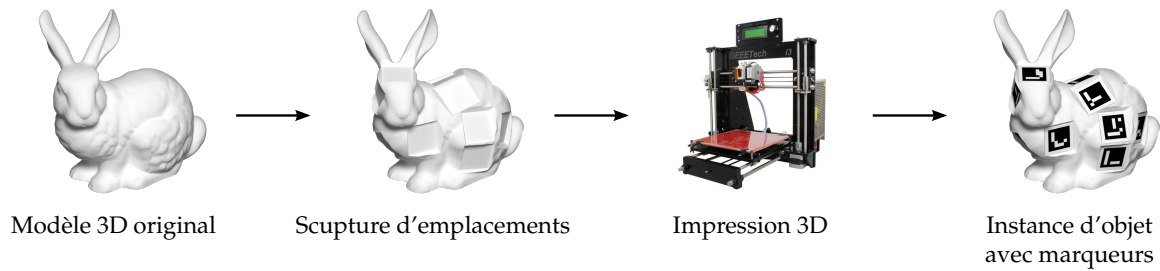


FIGURE 5.3 – Réalisation d’instances d’objet couvertes de marqueurs fiduciaires en des emplacements connus, utilisées afin de générer des jeux de données annotés.

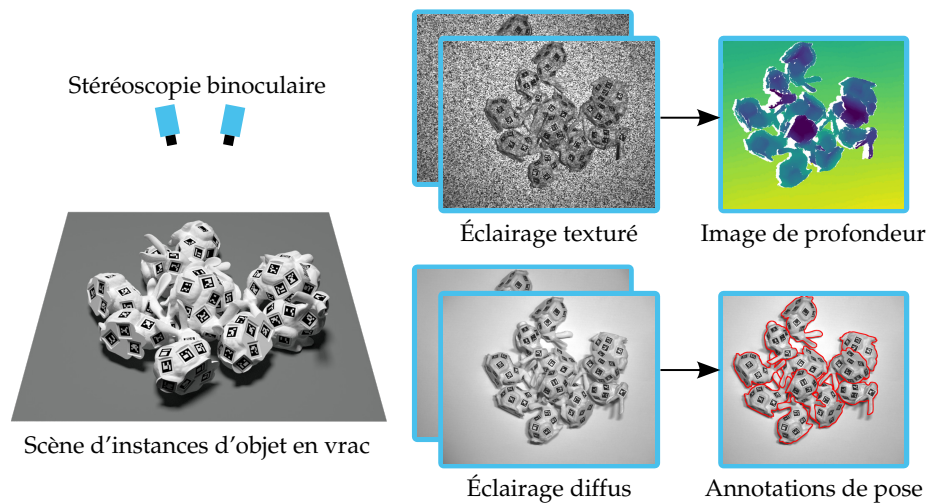


FIGURE 5.4 – Dispositif expérimental permettant de générer des données annotées. Étant donné une scène d’instances d’objet, une image de profondeur est générée par appariement stéréoscopique actif, tandis que les poses des instances d’objet sont annotées automatiquement par détection des marqueurs placés sur ces dernières dans des acquisitions produites avec un éclairage diffus.

réoscopique de la scène afin d’en générer une image de profondeur à l’aide d’un algorithme d’appariement stéréoscopique existant<sup>3</sup>. Une seconde acquisition stéréoscopique utilisant cette fois un éclairage diffus permet de produire une paire d’images où les marqueurs sont visibles, que nous exploitons afin d’estimer automatiquement la pose des instances et ainsi constituer une vérité terrain. L’ensemble de ces étapes sont illustrées figure 5.4.

Les marqueurs visibles peuvent être détectés, identifiés et localisés de manière relativement fiable au sein d’images 2D au moyen d’outils sur étagère, et nous nous appuyons dans nos expériences sur la bibliothèque ArUco (Garrido-Jurado et al., 2014). La détection des instances d’objet est alors extrêmement simple, en ce qu’une instance d’objet peut être considérée comme présente dans la scène si et seulement si au moins un des

3. Ici l’algorithme *Semi Global Block Matching*, une variante de la méthode de Hirschmuller (2008) implémentée dans la bibliothèque OpenCV.

marqueurs attribués à celle-ci est détecté, chaque marqueur n'étant assigné qu'à une instance unique d'objet. Étant donné une instance d'objet détectée, la localisation des marqueurs de cette dernière visibles dans l'image  $j \in \{1, 2\}$  permet de disposer des coordonnées 2D d'un ensemble de points dans l'image  $\mathbf{p}_{i,j} \in \mathbb{R}^2, i \in M_j$ , correspondant dans nos expériences aux quatre coins des marqueurs. La pose de ces marqueurs est par construction connue par rapport à l'objet, aussi chaque  $\mathbf{p}_{i,j}$  peut être mis en relation avec le point 3D correspondant  $\mathbf{X}_{i,j} \in \mathbb{R}^3$  exprimé dans un repère lié à l'objet. Il est alors possible d'estimer la pose  $[\hat{\mathbf{R}}, \hat{\mathbf{t}}]$  de l'instance – décrite par une matrice de rotation  $3 \times 3 \hat{\mathbf{R}}$  et un vecteur de translation 3D  $\hat{\mathbf{t}}$  en résolvant le problème d'ajustement de faisceaux multivues suivant :

$$[\hat{\mathbf{R}}, \hat{\mathbf{t}}] = \underset{\mathbf{R}, \mathbf{t}}{\operatorname{argmin}} \sum_{j \in \{1, 2\}} \sum_{i \in M_j} \|\pi_j(\mathbf{R}\mathbf{X}_{i,j} + \mathbf{t}) - \mathbf{p}_{i,j}\|^2, \quad (5.2)$$

où  $\pi_j$  représente l'opération de projection d'un point 3D exprimé dans le système de coordonnées de référence sur le plan image de la caméra  $j$ . La procédure d'annotation est ainsi automatique, mais nous procédons cependant à une étape de validation visuelle afin de s'assurer de la qualité de la vérité terrain produite.

**Expérimentation** Nous expérimentons cette approche en générant des jeux de données consistant en des scènes figurant un nombre variable d'instances (entre 0 et 11) d'un objet correspondant au lapin de Stanford couvert de 19 marqueurs, illustré figure 5.3. Ces instances sont placées en vrac sur un fond situé à une distance variable du capteur stéréoscopique, afin d'introduire une certaine variabilité de profondeur. Nous considérons deux types de surface pour le fond, illustrés figure 5.5 : un fond plat (308 scènes), représentatif du fond typique d'une boîte de laquelle on pourrait vouloir extraire des objets, et un fond bosselé (325 scènes), permettant une variabilité accrue de pose par rapport au précédent de manière à produire une distribution de pose plus représentative d'un scénario présentant de nombreuses instances entassées. Nous produisons également un petit jeu de données *clutter* (46 scènes), représentant des scènes encombrées où figurent entre 0 et 4 instances d'objet, afin de démontrer l'applicabilité de notre méthode d'annotation automatique à de tels scénarios. La validation manuelle des annotations a au total demandé moins de 40 minutes à une personne pour contrôler environ 700 scènes. 6% des images ont été écartées du fait d'échecs du détecteur de marqueur utilisé, correspondant soit à des faux positifs, des faux négatifs, une mauvaise identification d'un marqueur ou à une imprécision dans la localisation des coins de ce dernier. Il serait certainement possible de diminuer significativement ce taux d'échec en améliorant la robustesse des détections de marqueur ou en introduisant une tolérance aux valeurs aberrantes dans l'équation (5.2), cependant nous avons jugé ce taux d'échec acceptable pour notre application étant donné que l'acquisition et la validation manuelle sont des étapes relativement rapides. Des annotations supplémentaires telles que le taux d'occultation des instances et une segmentation des images sont ensuite produites au moyen de rendus synthétiques.



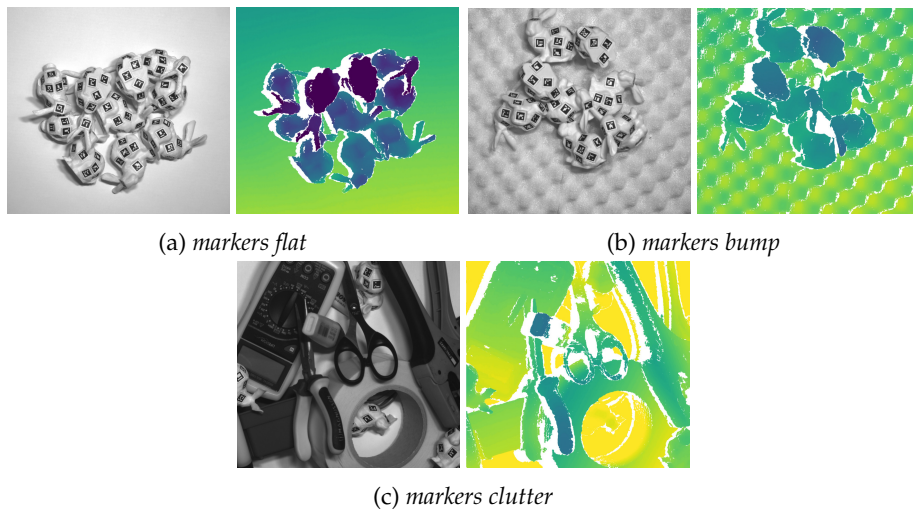


FIGURE 5.5 – Illustration des différents jeux de données annotés réels produits pour nos expérimentations. **Gauche** : image d’intensité, **droite** : image de profondeur.

### 5.1.2.2 Jeux de données synthétiques

Bien que l’approche précédente permette de générer efficacement des jeux de données annotés, celle-ci demeure intrusive, en ce qu’elle requière de légères modifications de la surface de l’objet afin de permettre de positionner les marqueurs de manière fiable. De plus, ces marqueurs sont visibles dans les images d’intensité, ce qui limite l’utilisabilité d’une telle approche pour évaluer des méthodes d’estimation de pose s’appuyant sur cette modalité. L’usage de données synthétiques permet de contourner ces difficultés, car alors il est possible de générer simplement une vérité terrain idéale de manière non intrusive. Il est en effet possible de générer des jeux de données synthétiques relativement représentatifs de scénarios en environnement contrôlé tels que des vracs d’objets au sein d’une installation industrielle, et nous exploitons ce moyen dans le cadre de nos travaux.

**Simulation physique** Nous générons des scènes de vracs d’objet au moyen d’outils que nous avons développés sous forme d’un plug-in intégrable à Blender ([Blender Online Community, 2017](#)) – un logiciel libre proposant des outils de modélisation, d’animation, de simulation physique et de rendu. Afin de générer une scène de vrac, nous simulons la chute successive d’instances d’objet au sein d’un bac, à l’aide des outils intégrés de simulation physique rigide s’appuyant sur la bibliothèque Bullet Physics ([Coumans, 2017](#)). Ces instances sont lâchées d’une hauteur donnée dans une position et une orientation aléatoire afin d’introduire une certaine variabilité dans les simulations. A titre indicatif, la simulation illustrée figure 5.6 de la chute de 80 instances d’un objet modélisé par un maillage de 800 triangles nécessite environ une heure de calcul monocœur sur un ordinateur portable équipé d’un processeur Intel Core i7-4702HQ cadencé à 2.20GHz, ce qui correspond

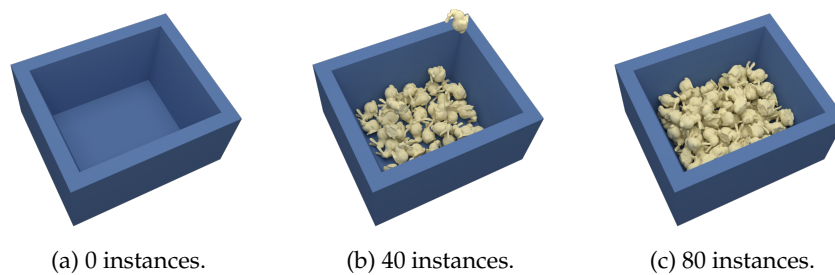


FIGURE 5.6 – Synthèse de scènes de vrac, par simulation physique de lâchers successifs d'instances d'objet dans un bac.

à la génération d'une scène toutes les 10 minutes environ, en exploitant les capacités de parallélisme de ce processeur.

**Simulation de capteur** Une fois une scène synthétique générée, il s'agit de simuler l'acquisition de données de celle-ci au moyen d'un capteur de profondeur. La synthèse d'images photoréaliste a été largement étudiée depuis notamment la formalisation de l'équation du rendu (Kajiya, 1986) et l'on dispose maintenant d'outils permettant de générer des images RGB de synthèse plausibles, tels que le moteur de rendu par lancer de rayons *Cycles* intégré à Blender que nous utilisons dans le cadre de nos travaux. Il s'agit là d'outils capables de produire des images d'un réalisme saisissant, pourvu qu'on consacre un effort important à la modélisation des matériaux, de l'éclairage des éléments de la scène, et des propriétés du capteur. Dans le cadre de nos expériences, nous sommes cependant d'avantage intéressés par la simulation d'acquisition d'images de profondeur. S'agissant de données géométriques, il est facile de produire des données de profondeur synthétiques idéales, cependant la simulation réaliste de capteur de profondeur est plus complexe. En effet, alors que les différentes caméras 2D produisent pour la plupart des données assez similaires, les caractéristiques d'une acquisition de données de profondeur sont assez dépendantes du capteur considéré ainsi que des différents traitements ayant été appliqués. Un capteur temps-de-vol souffre ainsi d'artéfacts de reconstruction différents d'un capteur par triangulation; le premier étant sujet notamment à des problématiques d'interférences dues à des réflexions multiples, tandis que le second souffre typiquement de problématiques d'appariements<sup>4</sup>. Pour un même capteur, les caractéristiques des images obtenues peuvent de plus être assez différentes suivant les traitements effectués afin d'en extraire de l'information de profondeur, ainsi qu'illustré figure 5.7. Aussi, bien que des simulateurs de capteurs 3D (Gschwandtner et al., 2011) et des modèles de bruits (Handa et al., 2014) aient été proposés dans la littérature pour certains capteurs spécifiques, ceux-ci restent relativement dépendants d'un capteur particulier et leur adaptation à une autre configuration n'est pas triviale. Afin de dépasser ces limitations, nous utilisons une approche consistant à

4. Entre projetés d'un même point 3D selon différents points de vue dans le cas d'un capteur stéréoscopique, ainsi qu'avec le rayon de la source éclairant ce point dans le cas d'utilisation de projection de lumière structurée (p. ex. projection de franges).

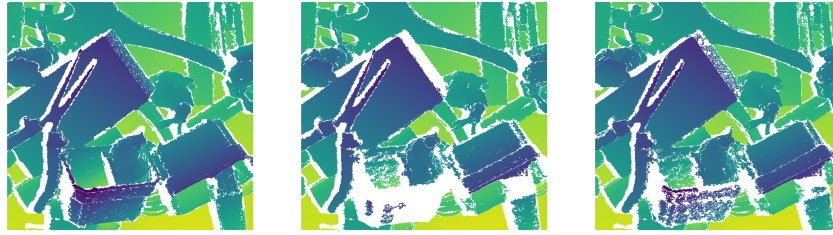
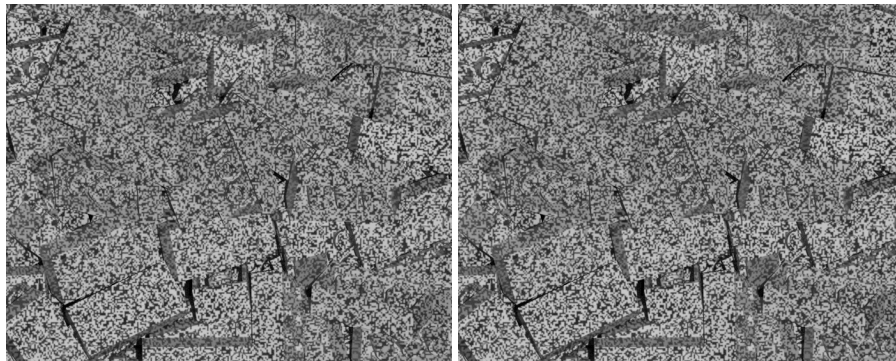


FIGURE 5.7 – Variabilité des images de profondeur reconstruites suivant la méthode de reconstruction et la paramétrage employé. Exemple avec la méthode d'appariement stéréoscopique SGBM de la bibliothèque OpenCV pour différents paramétrages.

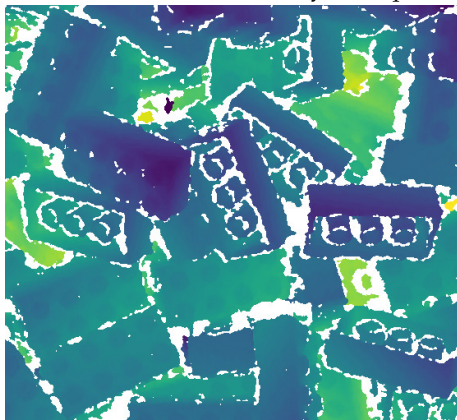
simuler le comportement complet d'un capteur de profondeur. Nous nous focalisons sur le cas d'un capteur stéréoscopique binoculaire utilisant un projecteur de motif comme celui évoqué section 5.1.2.1, car il s'agit d'une technologie couramment utilisée chez Siléane. Celle-ci permet des acquisitions rapides mais produit des données relativement bruitées comparé à d'autres technologies comme la triangulation LASER, aussi il s'agit là d'un scénario d'évaluation présentant un certain challenge. Nous éclairons la scène synthétique au moyen d'un projecteur de motif modélisé par une source de lumière peu étendue et une grille opaque constituée d'un masque binaire pseudo-aléatoire, de manière à reproduire l'éclairage texturant utilisé par un capteur 3D réel. Nous imageons alors la scène suivant le point de vue de deux caméras synthétiques placées au dessus du vrac, et procédons à la reconstruction 3D d'une image de profondeur à partir de ces dernières par appariement stéréo selon le même algorithme que celui employé par le capteur réel. Les images de profondeur ainsi obtenues s'avèrent alors visuellement réalistes, présentant des caractéristiques proches de celles obtenues avec un capteur réel sans pour autant nécessiter de modéliser le bruit de ce dernier. La figure 5.8 illustre des données typiques générées suivant cette méthode. D'autres capteurs basés sur les propriétés de l'optique géométrique tels que le *Microsoft Kinect v1* ou encore des scanners LASER ou multifranges pourraient être simulés suivant un procédé similaire, pourvu d'avoir accès aux algorithmes utilisés par ces derniers pour transformer leurs données brutes d'entrée en données de profondeur. La synthèse d'une vue RGBD ainsi que de toutes les annotations y afférentes à partir d'images de résolution 640x512 requière entre 6 et 7 minutes sur l'ordinateur portable évoqué plus haut, sans accélération GPU ou optimisation spécifique du paramétrage de rendu. La question du réalisme de ces images, et de la pertinence de leur utilisation sera discutée plus avant section 5.2.5.

### 5.1.3 Quantification des performances

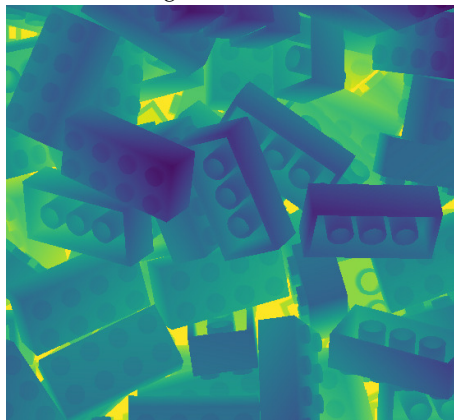
Une fois des données annotées à disposition se pose la question de l'évaluation de résultats obtenus sur celles-ci. Dans cette section, nous nous intéressons donc à la manière de quantifier les performances d'une méthode de détection et d'estimation de pose sur un jeu de données annoté.



(a) Rendu synthétique stéréo avec éclairage texturé.



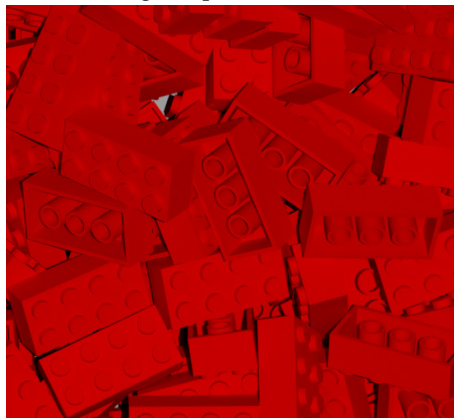
(b) Image de profondeur bruitée.



(c) Image de profondeur idéale.



(d) Segmentation des instances.



(e) Rendu RGB

FIGURE 5.8 – Synthèse de données RGBD (b et e) pour l'évaluation. (a) On réalise un rendu stéréoscopique de la scène avec un éclairage projetant un motif texturant, (b) que l'on utilise en entrée d'un algorithme de reconstruction 3D afin de synthétiser une image de profondeur bruitée. (c) L'usage de la simulation permet également de produire des données non bruitées, (d) ainsi que des annotations comme la segmentation des instances dans l'image.

**Extraction d'information** Notre problème peut être formulé comme un problème d'extraction d'information (*information retrieval* en anglais). Un tel problème consiste en effet typiquement à retourner les documents pertinents à une requête parmi une base de données. Les bases de données actuelles peuvent être extrêmement larges<sup>5</sup>, et les utilisateurs de moteurs de recherche ne sont souvent intéressés que par une liste limitée de résultats pertinents, sans rechercher nécessairement l'exhaustivité. Le problème de détection et d'estimation de pose d'instances d'objet partage ces caractéristiques. En effet, étant donné une acquisition 3D en entrée comme requête, il s'agit pour nous de retourner la pose des instances d'objets pertinentes vis-à-vis de celles-ci. L'espace de pose est infiniment grand<sup>6</sup>, et dans le cas de scènes de vrac, il s'agit moins de localiser l'ensemble des instances présentes dans la scène que d'en localiser un nombre suffisant pour la tâche à accomplir, par exemple une manipulation robotisée. Pour ces raisons, la démarche de quantification des performances que nous proposons ici s'inspire des différents travaux et standards du domaine de l'extraction d'information.

### 5.1.3.1 Classification binaire

Afin de mener à bien les évaluations, nous souhaitons être en mesure de classifier si une hypothèse de pose est valide ou non pour une scène donnée, et si une instance d'objet a été localisée ou non.

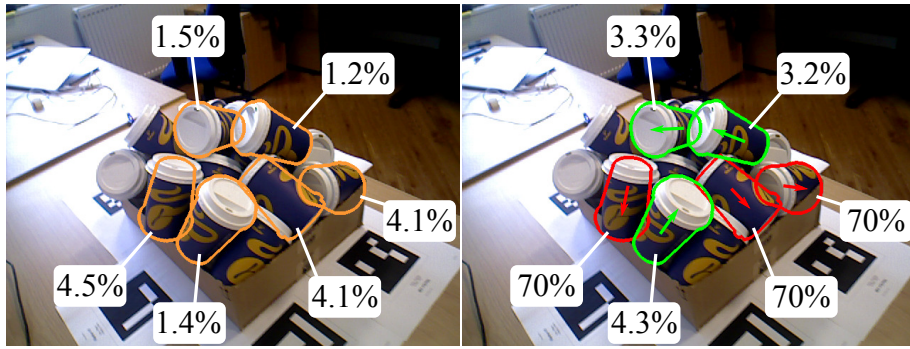
**Critère d'appariement** Nous définissons pour ce faire un critère  $m$  permettant de décider si une hypothèse de pose  $p$  ainsi qu'une instance de la scène de pose  $t$  peuvent être considérées comme appariées, à partir de la notion de distance entre ces dernières proposée au chapitre 3 :

$$m(p, t) \triangleq (d(p, t) < \delta). \quad (5.3)$$

Ainsi que décrit précédemment, cette distance permet de prendre en compte correctement les objets présentant des symétries et présente donc une alternative pertinente aux mesures employées dans l'état de l'art évoquées section 5.1.1.2, comme l'illustre la figure 5.9. La classe de symétrie considérée pour un objet est dépendante de l'application aussi celle-ci doit-elle être définie manuellement : à titre d'exemple, les gobelets à café de la figure 5.9 peuvent être considéré comme de révolution (c'est le cas dans nos expériences) s'il s'agit uniquement de les saisir, mais doivent être considérés sans invariance s'il s'agit de positionner correctement le trou de leur couvercle. On considère un seuil dur  $\delta \in \mathbb{R}^{+*}$  représentant la distance maximale tolérée entre une hypothèse de pose et la vérité terrain. Le choix d'un tel seuil est également très dépendant de l'application, des données et des dimensions de l'objet, mais afin de nous affranchir de ces considérations nous fixons dans le cadre de nos expériences la valeur de ce dernier arbitrairement à 10% du diamètre de la plus petite sphère englobant l'objet, centrée en le centre de gravité de la surface de ce dernier.

5. Plus de 4 milliards de pages web étaient indexées en 2016, selon <http://www.worldwidewebsize.com/> (Van den Bosch et al., 2016).

6. Au sens de « est un ensemble infini car continu ».



(a) Mesure de (Hinterstoisser et al., 2012b).

(b) Distance proposée.

FIGURE 5.9 – Mesure de dissimilarité (exprimée en % du diamètre  $D$  de l'objet) entre des hypothèses de pose (contours superposés à l'image) et les poses correspondantes de la vérité terrain. Dans cet exemple réalisé sur une scène du jeu de données de Doumanoglou et al. (2016), le critère largement répandu de Hinterstoisser et al. (2012b) adapté aux objets symétriques classifierait chaque hypothèse de pose comme valide (dissimilarité inférieure à  $10\% \cdot D$ ). A contrario, la distance proposée prend explicitement en compte les symétries de l'objet, ce qui permet de discriminer les vrais positifs (en vert) des faux positifs (hypothèses de pose à l'envers, en rouge).

À l'aide de ce critère, il est possible de réaliser les classifications évoquées précédemment. Pour ce faire, on note  $T$  l'ensemble des poses vraies des instances présentes dans la scène, obtenu par annotation, et  $P$  l'ensemble des hypothèses de pose retournées par la méthode de détection et d'estimation de pose d'objets évaluée.

**Instances d'intérêt** La pose de certaines instances peut être ambiguë étant donné une seule image d'entrée, et c'est notamment le cas dans le cadre de scènes de vrac où de nombreuses instances d'objet sont très voire totalement occultées. Aussi, nous considérons qu'il est possible que seul un sous-ensemble  $T_o \subset T$  des instances de la scène présentent un intérêt à être localisées, et dans le cadre de nos évaluations, nous considérerons typiquement  $T_o$  comme l'ensemble des instances  $t$  ayant un taux d'occultation  $o(t)$  inférieur à un seuil  $\delta_o \in [0, 1]$

$$T_o \triangleq \{t \in T | o(t) < \delta_o\}. \quad (5.4)$$

Nous définissons le taux d'occultation d'une instance comme la fraction du nombre de pixels visibles dans l'image de profondeur de cette instance, relativement au nombre de pixels de sa silhouette projetée sur le plan image sans occultation.

**Classification** En introduisant la notation

$$n_S(q) \triangleq \operatorname{argmin}_{r \in S} d(q, r) \quad (5.5)$$

afin de désigner la pose la plus proche parmi un ensemble  $S$  de la pose  $q$ , on définit alors les notions de *vrai positif* ( $TP$ ), *faux positif* ( $FP$ ) et *faux négatif* ( $FN$ ) comme suit :

$$\begin{aligned} TP &= \{(p, t) \in P \times T_o \mid m(p, t) \wedge p = n_P(t) \wedge t = n_T(p)\} \\ FP &= \{p \in P \mid \neg m(p, n_T(p)) \vee n_P(n_T(p)) \neq p\} \\ FN &= \{t \in T_o \mid \neg m(t, n_P(t)) \vee n_T(n_P(t)) \neq t\}. \end{aligned} \quad (5.6)$$

Un vrai positif est ainsi constitué d'une hypothèse de pose  $p$  appariée avec la pose « vraie » d'une instance de la scène d'intérêt, tandis qu'une hypothèse de pose située à une distance plus grande que le seuil d'appariement  $\delta$  des instances de la scène est considérée comme un faux positif. Les faux négatifs quant à eux correspondent aux instances d'intérêt de la scènes qui n'ont pas été localisées. Avec ces définitions, les hypothèses de pose faisant doublons sont considérées comme des faux positifs, tandis que celles appariées avec des instances dont la localisation n'est pas considérée comme d'intérêt (instances de pose parmi  $T \setminus T_o$ ) ne sont considérées dans aucune de ces catégories. La notion de *vrai négatif*, définie dans le cadre de problèmes de classification binaire n'est pas pertinente ici, du fait qu'il existe une infinité de poses qui, à raison, n'ont pas à être détectées.

À partir de ces notions, il est possible de définir diverses grandeurs statistiques permettant de quantifier les performances.

### 5.1.3.2 Précision et rappel

On définit à partir des notions de vrais positifs, faux positifs et faux négatifs, des grandeurs relatives offrant une certaine appréhension des performances globales obtenues. Celles-ci sont souvent exprimées au moyen de deux grandeurs complémentaires : la *précision* et le *rappel*.

La précision, cherche à quantifier la *pertinence* des résultats retournés, et est définie comme suit :

$$p = \frac{|TP|}{|FP| + |TP|}. \quad (5.7)$$

Une précision de 100% signifie ainsi que l'ensemble des résultats retournés sont corrects, tandis qu'une précision de 0 signifie qu'aucun de ceux-ci n'est correct.

Le rappel, également appelé parfois *sensibilité*, quantifie quant à lui l'*exhaustivité* des résultats, et est défini par

$$r = \frac{|TP|}{|FN| + |TP|}. \quad (5.8)$$

Un taux de rappel de 100% signifie que l'ensemble  $T_o$  des instances à localiser l'ont été, tandis qu'un rappel de 0 signifie qu'aucune n'a été correctement détectée.

**Nombre limité de résultat** Pour certaines applications, la détection et la localisation de l'ensemble des instances de la scène peut ne pas présenter

d'intérêt particulier. C'est notamment le cas dans le cadre du dévracage robotisé où il suffit typiquement de localiser une instance d'objet manipulable, et pour laquelle un nombre limité de résultats est donc suffisant.

On trouve dans la littérature existante en extraction d'information une mesure appelée *precision at k* ( $P@k$ ) (Craswell, 2009) synthétisant les notions de pertinence et d'exhaustivité des résultats pour ce cas d'application et consistant en la valeur de précision obtenue pour  $k \in \mathbb{N}^*$  résultats retournés.

Cependant, nous souhaitons pouvoir évaluer non seulement la bonne détection d'instances présentes dans une scène, mais également la bonne non détection d'instances lorsque celles-ci ne sont pas présentes<sup>7</sup>, ce qu'une telle métrique ne permet pas. Nous nous proposons donc de pallier cette difficulté en introduisant une notion alternative du rappel dans le cas où au plus  $k \in \mathbb{N}^*$  hypothèses de pose pourraient être formulées, que l'on définit ainsi :

$$r_{\leq k} = \frac{|TP|}{\min(n, |FN| + |TP|)}. \quad (5.9)$$

**Courbe précision-rappel** La précision et le rappel sont deux notions complémentaires et antagonistes. Il est aisé d'obtenir une précision de 100% en ne retournant aucun résultat, aux dépens d'un taux de rappel nul. Inversement, il serait possible d'atteindre un taux de rappel de 100% en *recouvrant* l'espace de pose de suffisamment d'hypothèses de manière à ce qu'il y ait au moins une hypothèse valide pour chaque instance présente dans la scène, aux dépens de la précision. Dès lors, une méthode d'extraction d'information peut généralement être réglée de manière à privilégier la précision devant le rappel ou inversement, et on représente donc ses performances sous forme d'une courbe précision-rappel comme illustré figure 5.10.

### 5.1.3.3 Indicateurs scalaires

Comme énoncé précédemment, précision et rappel sont deux notions indissociables et il est possible pour une même méthode de privilégier l'une à l'autre. Afin de permettre une comparaison chiffrée, il peut être utile de synthétiser les performances en une unique grandeur. Dans cette section nous décrivons quelques-uns de ces indicateurs, ainsi que leurs caractéristiques.

**Rappel à taux de précision fixé** Parmi les mesures les plus intuitives se trouve la mesure du taux de rappel  $r$  pour un taux de précision  $p$  fixé. Celle-ci peut se lire graphiquement (cf. figure 5.10), et exprime, en tolérant un taux d'erreur de  $(1 - p)$  dans les hypothèses de pose retournées, quelle est la fraction des instances il est possible de localiser parmi celles d'intérêt. La précision à taux de rappel fixé constitue la réciproque de cette mesure.

**Meilleur F-score** Le F-score (ou  $F_1$ -score (Zhang et Zhang, 2009)) vise à quantifier la performance pour un utilisateur qui attacherait autant d'import-

7. Cas d'une scène avec strictement moins de  $k$  instances à localiser.



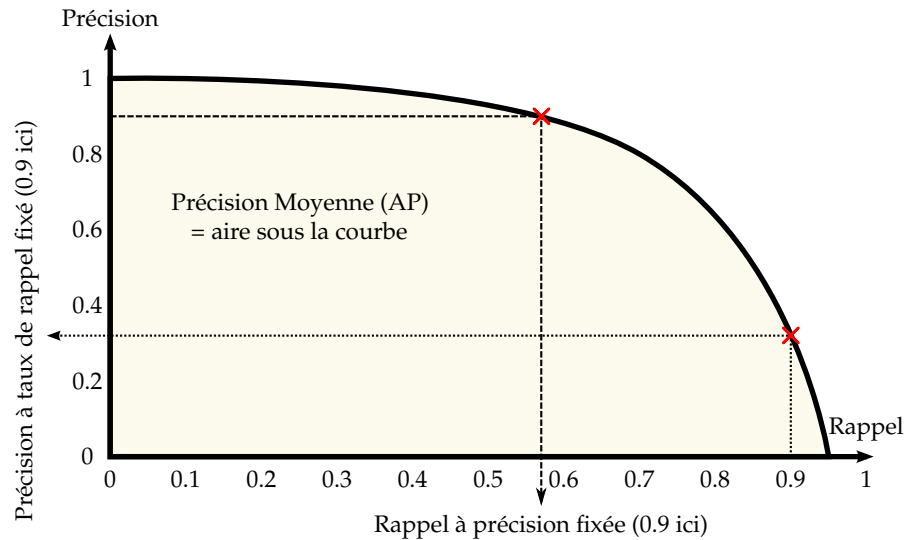


FIGURE 5.10 – Courbe précision-rappel, et illustration de différentes métriques de performance.

tance au rappel qu'à la précision, et est défini comme la moyenne géométrique de ces deux grandeurs :

$$F_1 = \frac{2}{\frac{1}{p} + \frac{1}{r}}. \quad (5.10)$$

Cette mesure est parfois utilisée dans la littérature en estimation de pose d'objet (Tejani et al., 2014; Doumanoglou et al., 2016), valant 0 dès lors que la précision ou le rappel est nul, et approchant 100% dans le cas de performances idéales. Cependant ainsi qu'évoqué précédemment il est toujours possible de jouer sur le paramétrage d'un algorithme pour privilégier la précision ou le rappel, ce qui conduit à des valeurs différentes de F-score. Aussi, la valeur présentée correspond généralement au meilleur F-score atteint en jouant sur le paramétrage.

**Précision Moyenne** Les mesures précédentes peuvent être qualifiées de *ponctuelles* en ce qu'elles quantifient les performances obtenues pour un paramétrage particulier de l'algorithme. La *Précision Moyenne* (*Average Precision* ou *AP*) est quant à elle une mesure globale, définie comme l'aire sous la courbe précision-rappel et correspondant à la précision moyenne obtenue pour chaque valeur de rappel

$$AP = \int_0^1 p(r) dr. \quad (5.11)$$

En pratique, l'intégrale est estimée en escalier pour chacune des valeurs de rappel  $(r_i)_{i=1\dots n}$  triées par ordre croissant ainsi

$$AP = \sum_{i=1}^n p(r_i) \cdot (r_i - r_{i-1}), \quad (5.12)$$

avec la convention  $r_0 = 0$ . Notez que certains auteurs définissent la Précision Moyenne de manière différente de celle présentée ici. A titre d'exemple, elle est définie dans le challenge de détection 2D et de reconnaissance d'objets Pascal VOC (Everingham et al., 2010) comme la moyenne de la précision interpolée pour 11 valeurs de rappel ( $\{0.1 \cdot k | k = 0, \dots, 10\}$ ), la précision interpolée au taux de rappel  $r$  étant définie par  $p_{inter}(r) = \max_{\tilde{r} \geq r} p(\tilde{r})$ .

Nous privilégions en général la Précision Moyenne aux autres mesures dans le cadre de nos expérimentations de par son caractère global. Il est néanmoins important de souligner qu'aucune mesure n'est parfaite, et que le choix de celle-ci est intimement lié à l'objectif applicatif visé.

#### 5.1.3.4 Mesures statistiques

Les métriques précédentes n'ont été définies que pour la quantification des performances réalisées pour une scène donnée. Afin d'obtenir une vue plus globale et robuste des performances d'une méthode de détection et d'estimation de pose, il est bon de considérer les performances obtenues sur plusieurs scènes (typiquement l'entièreté d'un jeu de données correspondant à un objet), voire sur plusieurs jeux de données distincts.

**Indicateur moyen** Étant donné un indicateur de performance  $B$ , et un ensemble de  $n \in \mathbb{N}^*$  scènes pour lesquelles on obtient des résultats classés  $(TP_i, FP_i, FN_i)_{i=1\dots n}$ , la littérature distingue typiquement deux approches pour définir un indicateur moyen  $\tilde{B}$  : la micro et la macro-moyenne.

**Macro-moyenne** La macro-moyenne consiste en la moyenne des indicateurs de performance pour chaque scène :

$$\tilde{B}_{\text{macro}} = \frac{1}{n} \sum_{i=1}^n B(|TP_i|, |FP_i|, |FN_i|). \quad (5.13)$$

**Micro-moyenne** La micro-moyenne revient à considérer l'ensemble des résultats retournés et des poses d'instances à trouver comme un tout, et à estimer la performance indépendamment des scènes :

$$\tilde{B}_{\text{micro}} = B\left(\sum_{i=1}^n |TP_i|, \sum_{i=1}^n |FP_i|, \sum_{i=1}^n |FN_i|\right). \quad (5.14)$$

Alors que la micro-moyenne est en quelque sorte pondérée par le nombre d'instances à trouver et le nombre de résultats retournés pour chaque scène, la macro-moyenne accorde une importance égale à chacune. Nous privilégions donc cette dernière approche pour nos évaluations, plus adaptée à notre cadre applicatif. Ce dernier consiste en effet typiquement à prendre

une image, localiser et manipuler une ou plusieurs instances et réitérer; aussi nous privilégions l'évaluation de la capacité de détecter et localiser des instances dans chaque scène devant celle de l'exhaustivité des résultats.

Pour un réglage donné de la méthode évaluée, on peut ainsi définir la précision et le rappel obtenus sur un jeu de données complet en moyennant les valeurs de précision et de rappel obtenues pour chaque scène. Cela permet ainsi de définir une courbe précision-rappel globale sur l'ensemble du jeu de données.

Lorsqu'il s'agit de synthétiser les performances obtenues pour plusieurs objets, il ne serait cependant pas des plus opportuns de procéder de la sorte. En effet, nous étudions la localisation d'instances d'un unique objet, aussi il est probable que les réglages permettant d'atteindre un certain équilibre entre précision et rappel pour un objet donné ne correspondent pas à ceux adaptés à un autre objet. Aussi, nous considérons typiquement en ce cas l'indicateur scalaire *Mean Average Precision*, qui consiste en la macro-moyenne de la Précision Moyenne obtenue sur chaque jeu.

## 5.2 Expérimentations sur différents jeux de données de vrac

Forts de la méthodologie décrite précédemment, nous expérimentons notre approche sur les deux jeux de données de Doumanoglou et al. (2016) ainsi que sur différents jeux de données de vrac annotés produits par nos soins, pour différents objets illustrés figure 5.11. Ces données sont mises à disposition de la communauté scientifique à l'adresse <http://rbregier.github.io/dataset2017>.

### 5.2.1 Analyse des performances de notre approche

Afin de mieux caractériser les performances de notre approche, nous en évaluons deux variantes :

- la première (*raw*) vise à évaluer de manière intrinsèque ses capacités de génération d'un ensemble de votes, et l'extraction de celui-ci d'hypothèses de pose pertinentes. Pour ce faire, nous extrayons les modes principaux de la distribution de votes générée par la forêt de décision<sup>8</sup>, et classons ceux-ci suivant la pseudo-densité de la distribution en ces modes, sans aucun filtrage. Nous procédons à un raffinement de pose des hypothèses correspondantes avant l'évaluation, de manière à évaluer la capacité à générer des hypothèses dans le bassin de convergence de poses de véritables instances.
- la seconde variante (*avec Post Processing*, ou *PP*) se veut plus représentative d'un cas d'utilisation pratique. Dans le cadre de celle-ci, nous n'extrayons qu'un nombre limité d'hypothèses de pose (20) de manière à limiter le temps de calcul et procédons aux étapes d'évaluation de la pertinence de ces dernières et de filtrage des hypothèses de pose décrites section 4.5.

8. Nous nous limitons aux 100 modes principaux pour des raisons de temps de calcul.



**coffee cup\***  
Révolution



**juice\***  
Sans symétrie



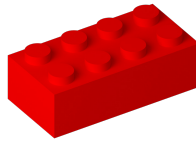
**markers**  
Sans symétrie

(\*) Jeux de données de Doumanoglou et al. (2016).

(a) Objets réels.



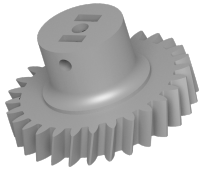
**bunny**  
Sans symétrie



**brick**  
Cyclique d'ordre 2



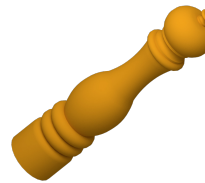
**candlestick**  
Révolution



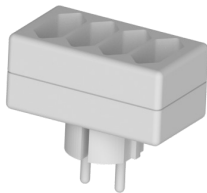
**gear**  
Révolution



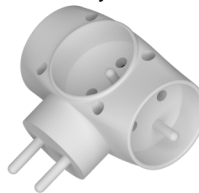
**markers**  
Sans symétrie



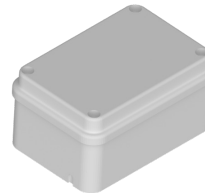
**pepper**  
Révolution



**tless 20**  
Cyclique d'ordre 2



**tless 22**  
Sans symétrie



**tless 29**  
Cyclique d'ordre 2

(b) Objets synthétiques.

FIGURE 5.11 – Objets considérés pour les évaluations quantitatives, avec précision de leur classe de symétrie.

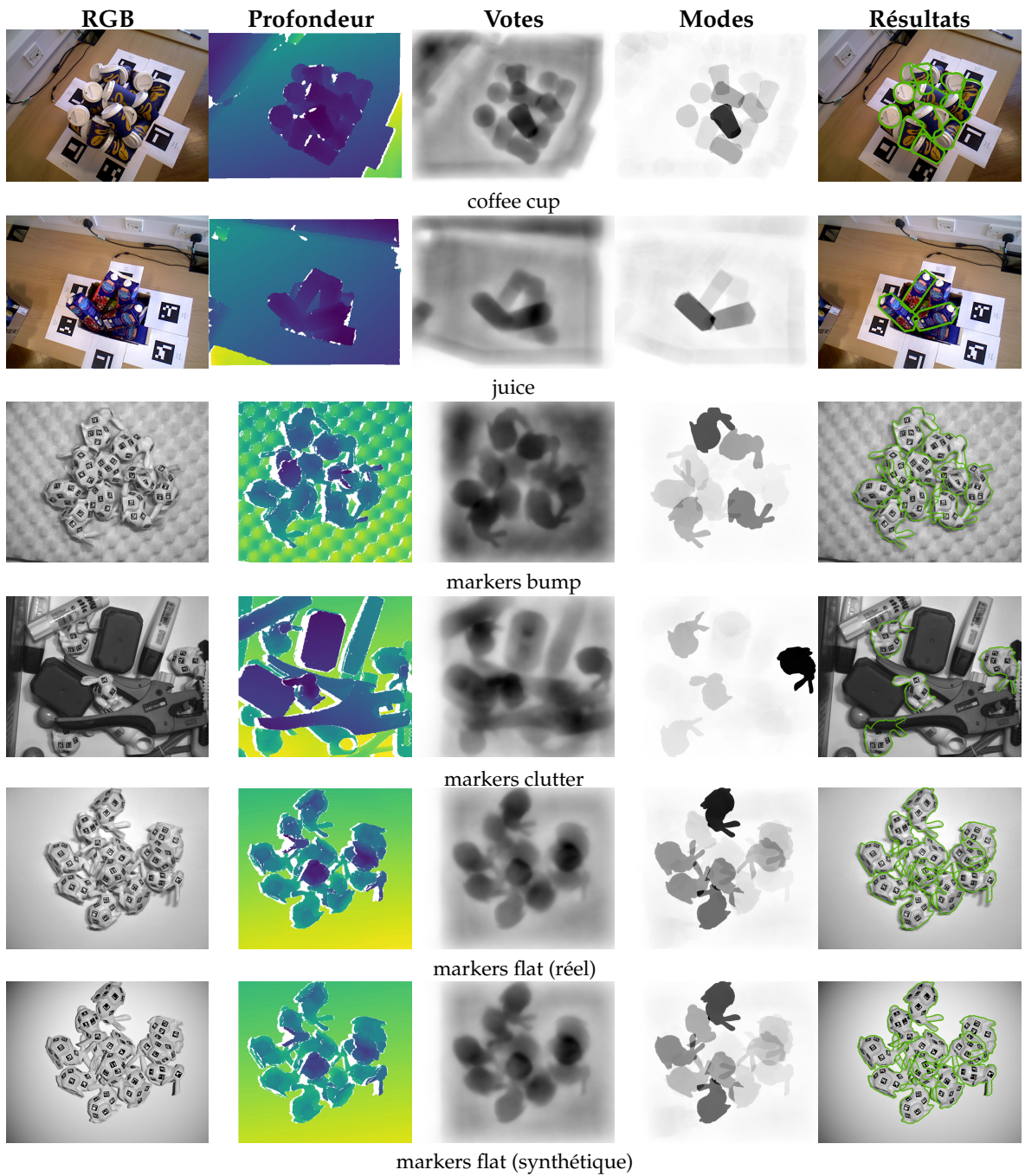
Les forêts de décision utilisées dans le cadre de ces expériences sont composées chacune de 5 arbres de décision, de profondeur maximale 20 et appris en testant 30 descripteurs faibles et 5 seuils par nœud. Nous ne réalisons aucun ajustement spécifique de paramètres selon le jeu de données considéré. L'ensemble des hyper-paramètres sont choisis suivant les heuristiques décrites chapitre 4 et se basant sur les informations géométriques fournies pour l'apprentissage, c'est-à-dire le modèle 3D de l'objet, la spécification de sa classe de symétrie, les paramètres intrinsèques typiques de la caméra ainsi que la plage typique de distance de l'objet à la scène. Bien que notre méthode soit conçue afin d'accepter en entrée un masque de segmentation des instances d'objet présentes dans l'image, et que celui-ci puisse être obtenu assez simplement dans le cas d'une scène d'instances en vrac dans un conteneur de forme connue comme c'est le cas ici, nous n'utilisons pas cette modalité afin de nous placer dans un contexte évaluatif plus ambitieux.

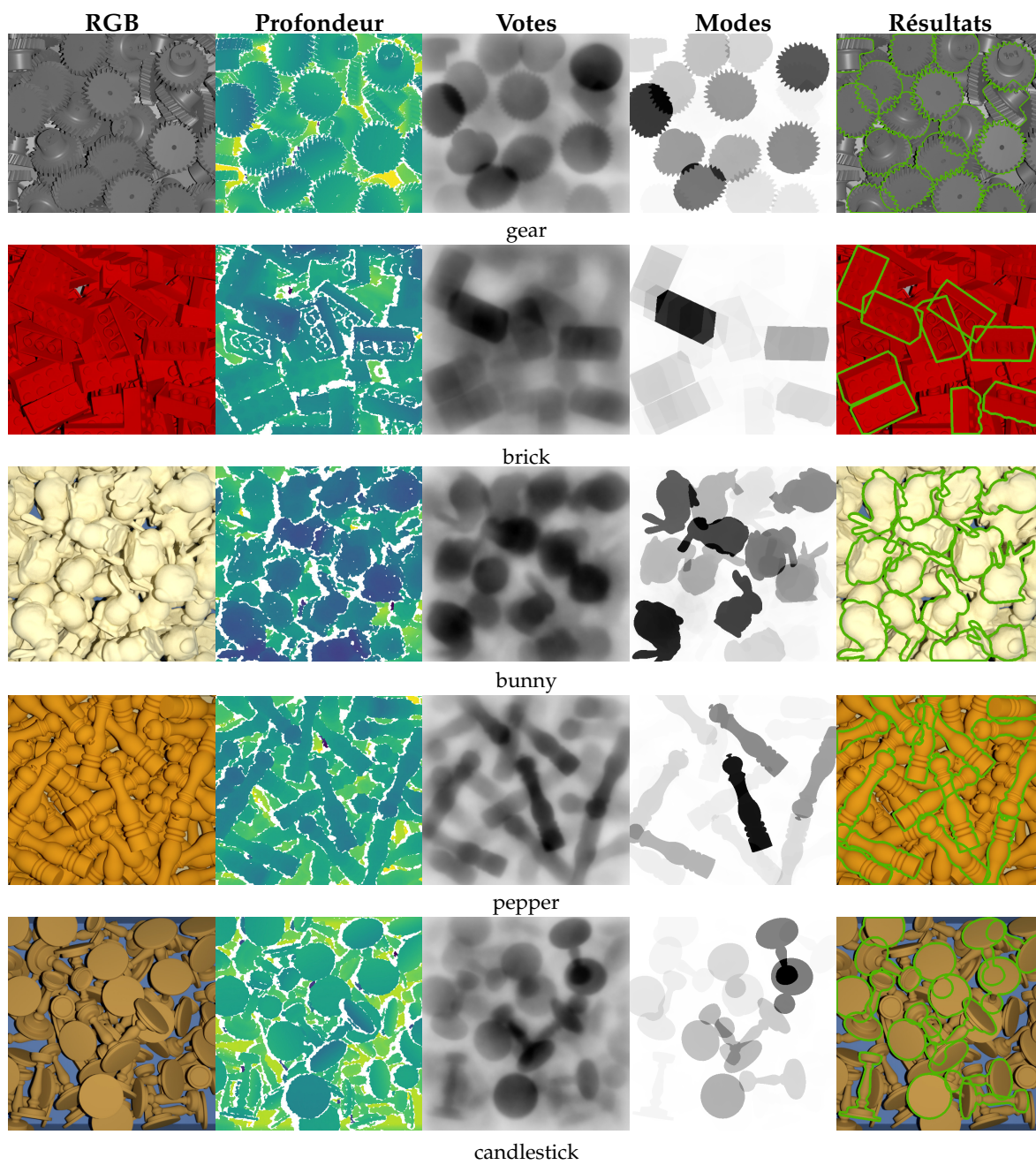
**Qualité des résultats** Le tableau 5.13 synthétise les performances mesurées sur les différents jeux de données pour ces deux variantes, et la figure 5.12 présente des exemples de résultats obtenus par la variante avec post-traitement (*PP*). Ainsi qu'évoqué dans la section méthodologie 5.1, nous nous fixons pour objectif la détection et l'estimation de pose des instances n'étant que modérément occultées. Nous considérons pour ce faire un seuil arbitraire de 50% de taux d'occultation, à l'exception des jeux *juice* et *coffee cup* de Doumanoglou et al. (2016) pour lesquels nous ne disposons pas des annotations nécessaires et pour lesquels nous considérons donc toutes les instances comme étant d'intérêt à localiser.

Globalement, les performances mesurées sont intéressantes et on observe que la distribution de votes générée, ainsi que les modes extraits de celle-ci constituent dans l'ensemble des hypothèses pertinentes, ainsi que représenté dans les 3<sup>èmes</sup> et 4<sup>èmes</sup> colonnes de la figure 5.12. Notamment, le mode principal détecté correspond dans la majorité des cas à une instance d'objet d'intérêt, comme en atteste le taux de rappel avec une précision de 100% pour au plus un résultat (R100), qui atteint pour notre approche brute les 100% sur la majorité des jeux de données, signifiant par là qu'il est possible sans erreur de localiser une instance d'objet dans chaque scène où il y en a une; sans générer de faux positifs dans les scènes où aucune instance n'est présente.

L'ajout d'une étape de post-traitement (*PP*) visant à filtrer les hypothèses de pose et les ordonner suivant une estimation de leur cohérence vis-à-vis des données présente quant à lui un effet variable. Si celui-ci permet d'augmenter les performances obtenues pour les objets *markers*, *juice*, et *brick*, il a cependant un effet extrêmement négatif sur l'objet *tless 22* et nuit également à la performance R100 pour l'objet *tless 29*.

Bien que cela s'avère pénalisant, il ne s'agit pas d'un aspect fondamentalement rédhibitoire. La technique d'estimation de la vraisemblance intrinsèque d'une hypothèse proposée section 4.5.1 a été choisie de manière relativement arbitraire – n'étant pas l'objet principal de nos travaux – et pourrait être significativement améliorée, notamment en adaptant cette dernière au jeu de données concerné de manière supervisée sur quelques





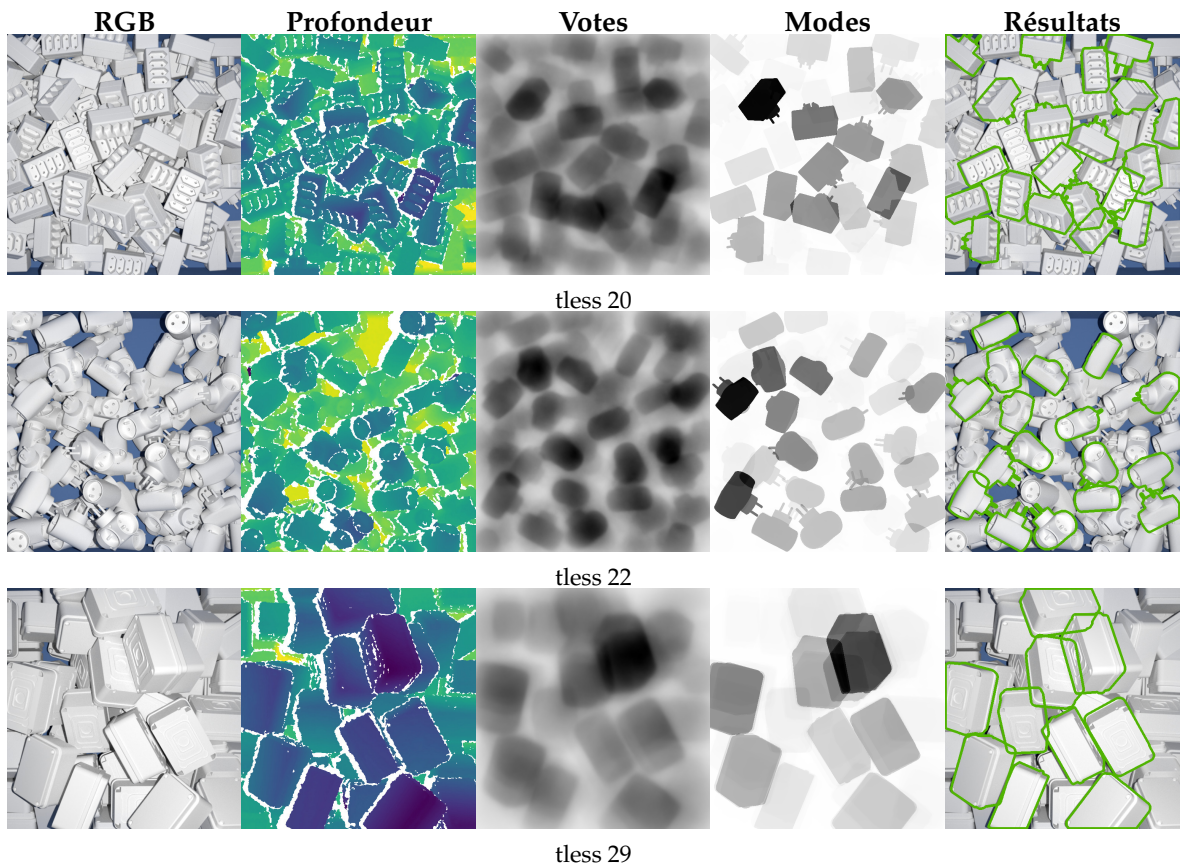


FIGURE 5.12 – Exemples de résultats retournés par notre méthode avec post-traitement (*PP*) sur les différents jeux de données de vrac. Le seuil de détection des résultats (représentés à droite) a été fixé de manière à maximiser le  $F_1$ -score de détection des instances moins de 50% occultées.



	Jeu de données	Approche brute ( <i>raw</i> )				Avec post-traitement ( <i>PP</i> )			
		AP	AP1	AP3	R100	AP	AP1	AP3	R100
Réal	markers bump	.98	1.00	1.00	1.00	<b>.99</b>	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>
	markers clutter	.88	.93	.87	.85	<b>.93</b>	<b>1.00</b>	<b>.94</b>	<b>1.00</b>
	markers flat	.94	1.00	.98	1.00	<b>.96</b>	<b>1.00</b>	<b>.99</b>	<b>1.00</b>
	juice	.09	.15	.07	.00	<b>.10</b>	<b>.19</b>	<b>.16</b>	<b>.00</b>
	coffee cup	.49	1.00	<b>1.00</b>	1.00	<b>.49</b>	<b>1.00</b>	.99	<b>1.00</b>
Synthétique	markers flat	.97	1.00	1.00	1.00	<b>.98</b>	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>
	tless 22	<b>.83</b>	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	.51	.80	.81	.01
	tless 20	<b>.91</b>	1.00	1.00	1.00	.62	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>
	tless 29	.56	<b>1.00</b>	.94	<b>1.00</b>	<b>.59</b>	.99	<b>.98</b>	.73
	brick	.46	.99	.90	.93	<b>.51</b>	<b>.99</b>	<b>.98</b>	<b>.99</b>
	gear	<b>.95</b>	1.00	1.00	1.00	.93	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>
	candlestick	<b>.73</b>	.98	.94	.94	.64	<b>1.00</b>	<b>.98</b>	<b>1.00</b>
	pepper	<b>.81</b>	1.00	1.00	1.00	.78	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>
bunny	<b>.88</b>	1.00	1.00	1.00	.86	<b>1.00</b>	<b>1.00</b>	<b>1.00</b>	

AP : Précision Moyenne.

AP $n$  ( $n \in \mathbb{N}^*$ ) : Précision Moyenne étant donné le renvoi d'au plus  $n$  résultats.

R100 : Rappel maximal étant donné le renvoi d'au plus un résultat pour une précision de 100%.

L'ensemble des instances d'intérêt considéré est ici réduit aux instances moins de 50% occultées, à l'exception des jeux *juice* et *coffee cup* de [Doumanoglou et al. \(2016\)](#) pour lesquels nous ne disposons pas d'information d'occultation.

TABLE 5.13 – Performances des deux variantes de notre approche sur les différents jeux de données.

images<sup>9</sup>. Les performances obtenues avec les approches brutes et avec post-traitement fournissent des bornes inférieures des performances qu'il est possible d'atteindre en optimisant ce calcul de vraisemblance. Celles-ci dépassent notamment les 99% de bonne détection d'une instance d'objet par scène (ou de bonne non détection lorsque l'objet est absent, exprimée par la mesure R100) pour l'ensemble des jeux de données à l'exception du jeu *juice* pour lequel cette borne inférieure est de 0% (nous revenons sur ce point dans la section suivante).

**Temps de calcul** La table 5.14 synthétise les durées d'exécution moyennes de notre méthode sur les différents jeux de données considérés. Les expériences ont été réalisées sur un ordinateur de bureau milieu de gamme *DELL Optiplex 9020* équipé d'un processeur *Intel Core i7-470 @ 3.60GHz*, sans accélérateur graphique dédié. Ces durées d'exécution, inférieures à la demi-seconde, sont compatibles avec la plupart des problématiques industrielles de débrassage rencontrées chez Siléane. Celles-ci peuvent être ajustées au besoin en utilisant une machine plus puissante ou en adaptant l'effort de recherche (nombre d'hypothèses de pose extraites, nombre d'itérations de raffinement, taille de la forêt de décision, etc.), et pourraient être améliorées significativement en procédant à un profilage plus fin et à une optimisation

9. Des approches plus gourmandes en données telles que les travaux de [Krull et al. \(2015\)](#) qui estiment la vraisemblance d'une hypothèse de pose au moyen d'un réseau de neurones sont également intéressantes à envisager, en ce qu'elles semblent montrer une certaine capacité de généralisation à de nouveaux objets.

TABLE 5.14 – Durée d’exécution moyenne de notre méthode ainsi que de ses principales étapes pour les différents jeux de données (en millisecondes), ainsi que l’écart-type associé, mesurés sur un ordinateur de bureau milieu de gamme *DELL Optiplex 9020* équipé d’un processeur *Intel Core i7-470 @ 3.60GHz*).

		Génération des points de référence	Vote de la forêt de décision	Extraction des 20 meilleures hypothèses de pose par Mean Shift	Evaluation et filtrage des hypothèses de pose	Total
Réel	markers bump	10 ± 7	63 ± 13	143 ± 20	127 ± 7	360 ± 23
	markers clutter	9 ± 7	50 ± 12	141 ± 29	121 ± 8	339 ± 35
	markers flat	10 ± 7	37 ± 11	162 ± 27	119 ± 7	341 ± 34
	juice	17 ± 10	88 ± 35	111 ± 31	51 ± 7	290 ± 51
	coffee cup	16 ± 10	132 ± 38	161 ± 34	36 ± 7	365 ± 25
Synthétique	markers flat simulation	9 ± 7	35 ± 11	161 ± 29	119 ± 8	338 ± 35
	bunny	13 ± 11	49 ± 15	159 ± 33	55 ± 8	294 ± 41
	tless 22	19 ± 9	134 ± 23	147 ± 19	49 ± 8	377 ± 28
	tless 20	19 ± 10	131 ± 23	294 ± 27	45 ± 6	547 ± 47
	tless 29	16 ± 10	52 ± 11	310 ± 38	65 ± 9	464 ± 38
	brick	16 ± 13	49 ± 17	86 ± 30	50 ± 18	229 ± 49
	gear	10 ± 7	69 ± 19	228 ± 27	120 ± 7	440 ± 20
	candlestick	16 ± 10	66 ± 20	72 ± 32	61 ± 7	223 ± 33
	pepper	17 ± 12	48 ± 14	59 ± 20	72 ± 8	205 ± 32
Moyenne		14 ± 10	72 ± 20	159 ± 29	78 ± 9	344 ± 36

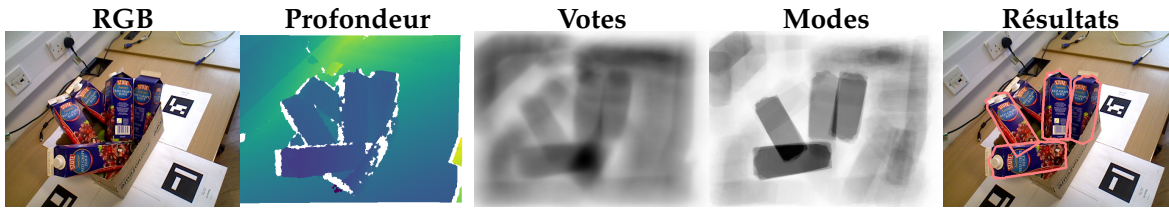
du code. L’approche utilisée ici est de plus de type *embarrassingly parallel*<sup>10</sup> ce qui permet d’envisager un portage de celle-ci sur du matériel adapté et plus efficace tel qu’un GPU.

### 5.2.2 Cas d’échecs

Dans cette section, nous abordons les limitations de notre approche observées lors de l’évaluation, et tâchons d’en expliquer les raisons.

**Échec sur le jeu *juice*** Les performances médiocres obtenues sur le jeu de données *juice* contrastent singulièrement avec celles obtenues sur les autres jeux de données. L’explication majeure de ce phénomène tient en la *quasi symétrie* de l’objet considéré, qui conduit à fréquemment retourner des résultats de pose orientés de manière incorrecte (cf. figure 5.15a). La résolution limitée des données utilisées rend en effet difficile de discerner ces

10. D’un parallélisme « embarrassant », suivant une expression anglo-saxonne établie.



(a) Erreur typique consistant en une mauvaise estimation de l'orientation de l'objet.



(b) Rendu synthétique RGBD de caractéristiques proche de celles des images réelles. La résolution limitée des images de profondeur rend difficile de discerner l'orientation de l'objet sans considération de sa texture.

Approche brute (raw)				Avec post-traitement (PP)			
AP	AP1	AP3	R100	AP	AP1	AP3	R100
.26	.68	.33	.00	.42	.97	.69	.93

(c) Performances obtenues en relaxant le critère d'évaluation de manière à tolérer la confusion des orientations représentées plus haut.

FIGURE 5.15 – Erreur typique d'orientation sur le jeu de données *juice*.

orientations uniquement à partir des données ou des contours de profondeur que nous utilisons dans notre approche. Afin de valider cette hypothèse, nous relaxons le critère d'évaluation de manière à ne pas différencier les 8 orientations de cet objet représentée figure 5.15b. Nous considérons pour ce faire que cet objet présente la même symétrie qu'un pavé droit à base carré<sup>11</sup>, dont le groupe de symétrie propre peut être exprimé

$$\left\{ \mathbf{R}_z^{k\pi/2} \mathbf{R}_x^{\delta\pi} \mid k \in \llbracket 0, 4 \rrbracket, \delta \in \{0, 1\} \right\} \quad (5.15)$$

dans le repère approprié. Le tableau 5.15c présente les performances obtenues selon ce critère. Celles-ci sont bonnes, bien que non idéales, avec notamment une Précision Moyenne de 97% en ce qui concerne la détection et l'estimation de pose d'une instance par scène (AP1), ce qui appuie cette interprétation.

11. Symétrie dite *ditéragonale dipyramidale* ( $D_{4h}$  suivant la notation de Schoenflies).

**Absence de segmentation du fond** L'absence de segmentation des instances durant cette évaluation viole une des hypothèses de notre méthode selon laquelle l'ensemble des points de référence considérés appartiennent à des instances d'objet. Cette absence conduit donc à la génération de faux positifs parmi les hypothèses de pose, mais peut également réduire le nombre d'instances détectées du fait que le post-traitement ne considère qu'un nombre limité de ces dernières, ainsi qu'illustré figure 5.16.

**Mauvaise convergence du raffinement d'hypothèse** La *densité* importante des scènes considérées – constituées d'instances diverses enchevêtrées – conduit également parfois le raffinement d'hypothèse de pose par Iterative Closest Point à diverger de la bonne solution, du fait de la confusion entre les points de l'instance considérée et ceux appartenant à d'autres instances, ainsi qu'illustré figure 5.17.

**Hésitation entre différentes hypothèses** Différentes poses d'un même objet peuvent conduire à des images de profondeur présentant une certaine similarité, conduisant la forêt de décision utilisée à parfois *hésiter* entre différentes hypothèses en générant une distribution de pose présentant plusieurs modes marqués pour une même instance d'objet. L'ambiguïté est alors levée en choisissant l'hypothèse la plus vraisemblable vis-à-vis des données suite aux étapes de validation et de filtrage décrites section 4.5. Cette estimation de vraisemblance est cependant relativement naïve, et les données ne sont pas toujours suffisantes afin de lever l'ambiguïté, même pour un humain. Aussi celle-ci conduit-elle parfois à des erreurs, ainsi qu'illustré figure 5.18.

### 5.2.3 Comparaison avec deux méthodes de référence

Nous évaluons également deux adaptations de méthodes de référence de l'état de l'art sur ce jeu de données, afin de pouvoir nous positionner vis à vis de celles-ci :

- la méthode de Drost et al. (2010) (*Point Pair Feature*, ou PPF), qui comme la nôtre est basée sur la génération d'un agrégat de votes de pose, afin d'en extraire des hypothèses pertinentes. Nous utilisons une implémentation de cette dernière développée par nos soins. Les auteurs ne décrivant pas précisément leurs techniques d'extraction d'hypothèses de pose, de raffinement et d'évaluation de celles-ci, nous procédons pour ces étapes suivant la même procédure que pour notre approche, décrite au chapitre 4.
- la méthode de Hinterstoisser et al. (2012b) (*LINEMOD+*) basée sur une recherche exhaustive de patrons 2D correspondant à différents points de vue de l'objet dans l'image et un post-traitement des détections. La méthode originale procède par filtrage des détections en vérifiant l'adéquation de l'hypothèse de pose avec la couleur observée dans l'image, ainsi qu'après raffinement de pose avec les données de profondeur. Notre cas d'utilisation principal se focalisant sur des scènes de vrac d'instances d'un unique objet non texturé, nous ne mettons néanmoins

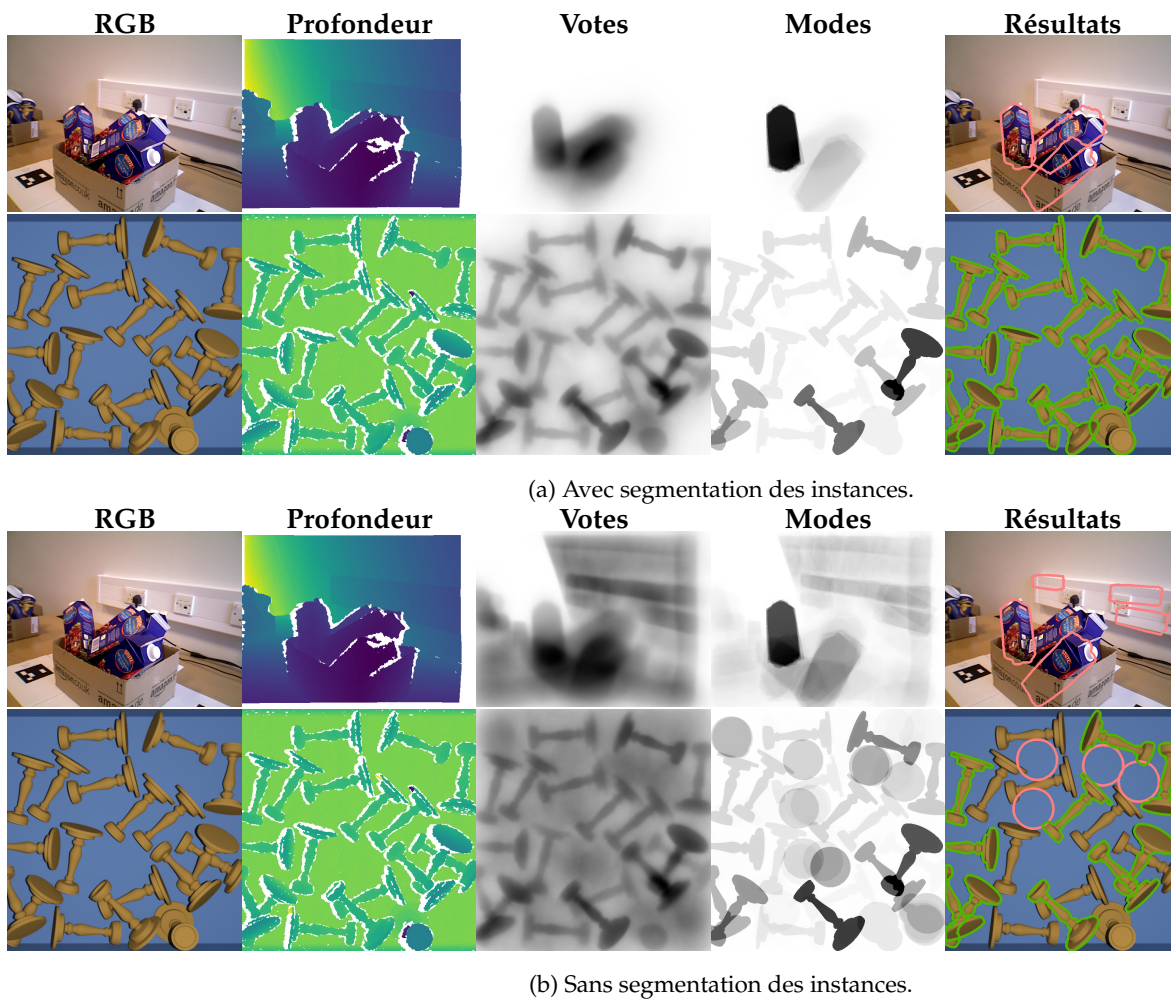


FIGURE 5.16 – Comparaison de résultats obtenus avec notre approche en supposant ou non une segmentation préalable des instances dans les images d'entrée. L'absence de segmentation préalable conduit à une distribution de votes présentant d'avantage de modes correspondant à des faux positifs, ce qui dans le cas de l'examen d'un nombre limité d'hypothèses de pose réduit le nombre d'instances localisées.

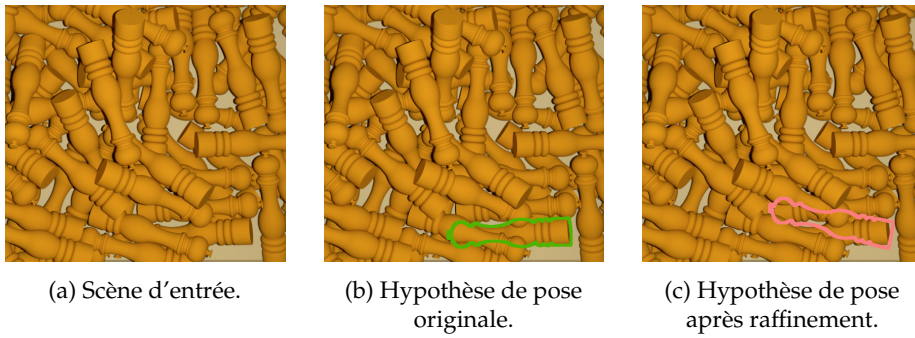


FIGURE 5.17 – Échec de raffinement de pose, pouvant diverger de la bonne solution.

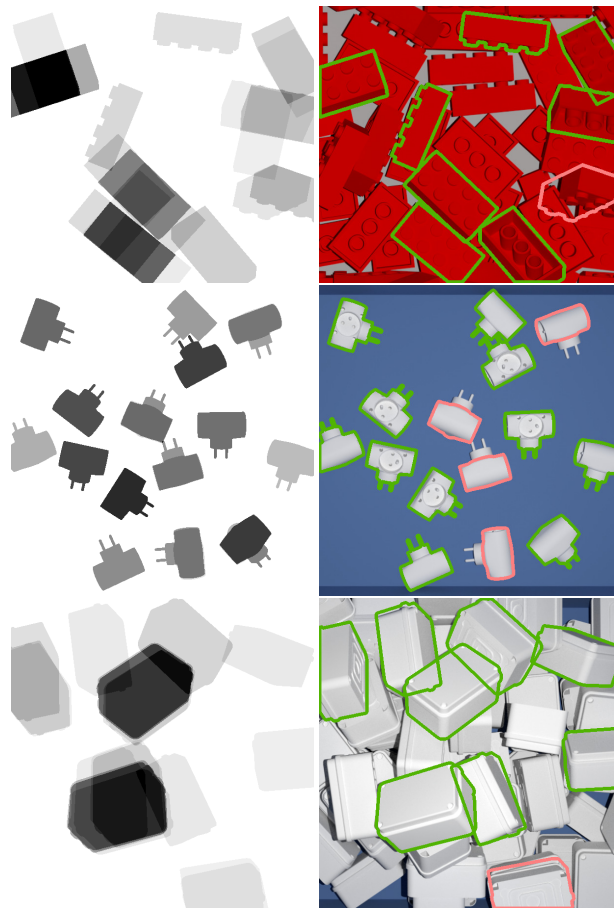


FIGURE 5.18 – Hésitation de la forêt de décision, qui génère plusieurs hypothèses de pose marquées pour une même instance. **À gauche** : votes déplacés via Mean Shift au niveau des modes de densité. **À droite** : hypothèses de pose retenues (à tort ou à raison) après estimation de la vraisemblance de celles-ci vis-à-vis des données.

pas en place de filtrage sur la couleur. L'approche de [Hinterstoisser et al. \(2012b\)](#) n'est de plus pas applicable en l'état pour la détection de plusieurs instances d'un même objet dans une même scène car elle ne prend pas en compte les doublons, en ce que différents patrons correspondant à des vues relativement similaires peuvent être détectés. Nous évaluons donc là encore une version modifiée. Nous convertissons les détections 2D de LINEMOD en hypothèses de pose<sup>12</sup>, en leur assignant la profondeur minimisant l'erreur quadratique entre les données de profondeur observées et celles du projeté de l'hypothèse de pose sur le plan image. Nous filtrons alors les doublons au moyen de la distance introduite chapitre 3, pour enfin procéder au raffinement d'hypothèses, à l'évaluation et au filtrage de celles-ci suivant la même procédure que pour les autres méthodes. Nous nous appuyons pour ce faire sur l'implémentation de LINEMOD de Stefan Holtzer présente dans la bibliothèque PCL ([Rusu et Cousins, 2011](#)).

Afin d'être tolérant aux imprécisions de positionnement des hypothèses de pose, nous procédons à une phase de raffinement plus conséquente que dans la section 5.2.1 de 30 itérations d'ICP et qui porte notamment la durée moyenne d'exécution de notre approche de 340ms à 1.5s par scène. Ce dernier n'a qu'un impact négligeable sur les performances de notre approche, mais apporte un gain de performance conséquent à *LINEMOD+* et *PPF*. Le matériel additionnel à notre publication ([Brégier et al., 2017a](#)) fournit de plus amples informations sur les détails d'implémentation employés, et le tableau 5.19 synthétise les résultats de ces évaluations.

---

12. En nous limitant aux 10000 premières détections pour des raisons de temps de calcul.

TABLE 5.19 – Performances de trois méthodes de détection et d’estimation de poses sur nos jeux de données d’évaluation. Nous considérons pour chacune quatre variantes exploitant ou non les propriétés de symétrie potentielles des objets (*sym* / *nosym*), et procédant ou non à une étape de post-traitement des 20 meilleures hypothèses de pose (*PP* / *raw*). La prise en compte des symétries améliore les performances de l’ensemble des méthodes (en gras), et notre méthode se compare favorablement vis-à-vis des deux autres (surlignage).

Jeu de données	Approche brute ( <i>raw</i> )						Avec post-traitement ( <i>PP</i> )						
	PPF		LINEMOD+		Notre méthode		PPF		LINEMOD+		Notre méthode		
	<i>nosym</i>	<i>sym</i>	<i>nosym</i>	<i>sym</i>	<i>nosym</i>	<i>sym</i>	<i>nosym</i>	<i>sym</i>	<i>nosym</i>	<i>sym</i>	<i>nosym</i>	<i>sym</i>	
Réel	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3	AP AP1 AP3
markers bump	.35 .66 .43	- - -	.85 1.00 .96	- - -	.98 1.00 1.00	- - -	.56 .97 .84	- - -	.91 1.00 .99	- - -	.99 1.00 1.00	- - -	
markers clutter	.34 .36 .31	- - -	.57 .67 .53	- - -	.88 .93 .87	- - -	.52 .70 .52	- - -	.68 .83 .69	- - -	.93 1.00 .94	- - -	
markers flat	.26 .54 .31	- - -	.83 .99 .97	- - -	.94 1.00 .98	- - -	.46 .94 .76	- - -	.90 1.00 .99	- - -	.96 1.00 .99	- - -	
juice	.04 .15 .07	- - -	.01 .01 .01	- - -	.09 .15 .07	- - -	.07 .29 .11	- - -	.06 .24 .10	- - -	.10 .19 .16	- - -	
coffee cup	.16 .76 .53	<b>.28 .96 .85</b>	.03 .37 .10	<b>.08 .37 .17</b>	.24 1.00 .57	<b>.49 1.00 1.00</b>	.23 .98 .90	<b>.30 1.00 .92</b>	.10 .95 .61	<b>.20 1.00 .93</b>	.34 1.00 .99	<b>.49 1.00 .99</b>	
Synthétique	markers flat	.29 .55 .36	- - -	.87 .99 .97	- - -	.97 1.00 1.00	- - -	.50 .94 .79	- - -	.91 .99 .99	- - -	.98 1.00 1.00	- - -
tless 22	.08 .52 .34	- - -	.19 .63 .54	- - -	.83 1.00 1.00	- - -	.12 .90 .77	- - -	.21 .81 .81	- - -	.51 .80 .81	- - -	
tless 20	.10 .49 .35	<b>.20 .82 .64</b>	.17 .81 .44	<b>.25 81 .75</b>	.54 1.00 .66	<b>.91 1.00 1.00</b>	.14 .92 .84	<b>.23 .98 .94</b>	.24 1.00 .97	<b>.31 1.00 .99</b>	.43 1.00 1.00	<b>.62 1.00 1.00</b>	
tless 29	.15 .69 .40	<b>.19 .76 .56</b>	.14 .71 .34	<b>.20 .71 .50</b>	.38 .99 .61	<b>.56 1.00 .94</b>	.21 .90 .76	<b>.23 .91 .79</b>	.20 .88 .84	<b>.26 .92 .86</b>	.50 1.00 1.00	<b>.59 .99 .98</b>	
brick	.05 .24 .13	<b>.08 .36 .23</b>	.20 .97 .47	<b>.31 .97 .76</b>	.29 .99 .54	<b>.46 .99 .90</b>	.10 .68 .47	<b>.13 .77 .59</b>	.32 .98 .95	<b>.39 .99 .97</b>	.39 .99 .97	<b>.51 .99 .98</b>	
gear	.24 .42 .30	<b>.63 .94 .89</b>	.15 .93 .31	<b>.44 .95 .84</b>	.41 1.00 .62	<b>.95 1.00 1.00</b>	.30 .81 .76	<b>.63 .99 .97</b>	.25 .99 .92	<b>.50 .99 .98</b>	.47 1.00 1.00	<b>.93 1.00 1.00</b>	
candlestick	.09 .32 .22	<b>.16 .60 .47</b>	.17 .86 .29	<b>.38 .92 .78</b>	.23 .89 .48	<b>.73 .98 .94</b>	.15 .85 .75	<b>.22 .85 .78</b>	.26 1.00 .96	<b>.49 1.00 1.00</b>	.25 .93 .89	<b>.64 1.00 .98</b>	
pepper	.04 .08 .06	<b>.06 .25 .13</b>	.03 .11 .05	<b>.04 .11 .08</b>	.34 1.00 .52	<b>.81 1.00 1.00</b>	.08 .68 .38	<b>.12 .85 .57</b>	.03 .13 .07	<b>.03 .14 .08</b>	.42 1.00 1.00	<b>.78 1.00 1.00</b>	
bunny	.29 .83 .66	- - -	.39 .97 .94	- - -	.88 1.00 1.00	- - -	.37 .99 .97	- - -	.45 .99 .98	- - -	.86 1.00 1.00	- - -	
Moyenne objets symétriques	.12 .43 .28	<b>.23 .67 .54</b>	.13 .68 .28	<b>.24 .69 .55</b>	.35 .98 .57	<b>.70 1.00 .97</b>	.17 .83 .70	<b>.27 .91 .80</b>	.20 .85 .76	<b>.31 .86 .83</b>	.40 .99 .98	<b>.65 1.00 .99</b>	
Moyenne globale	.18 .47 .32	<b>.23 .59 .45</b>	.33 .72 .49	<b>.39 .72 .63</b>	.57 .92 .71	<b>.75 .93 .91</b>	.27 .82 .69	<b>.32 .86 .74</b>	.39 .84 .78	<b>.45 .85 .81</b>	.58 .92 .91	<b>.71 .93 .92</b>	

AP ( $n \in \mathbb{N}^*$ ) : Précision Moyenne étant donné le renvoi d’au plus  $n$  résultats.

L’ensemble des instances d’intérêt considéré est ici réduit aux instances moins de 50% occultées, à l’exception des jeux *juice* et *coffee cup* de Doumanoglou et al. (2016) pour lesquels nous ne disposons pas de cette information.



Ainsi qu'on peut le constater, notre méthode se compare très favorablement aux deux autres, particulièrement dans sa version brute sans post-traitement (*raw*) où elle obtient systématiquement de meilleures performances. Ce résultat suggère que celle-ci génère des hypothèses de pose d'avantage pertinentes que les méthodes *LINEMOD+* et *PPF* testées. On observe notamment de manière qualitative que la distribution de votes générée par notre approche, basée sur de l'apprentissage, semble bien plus pertinente et moins bruitée que celle de *PPF*, qui repose sur des descripteurs conçus manuellement, ainsi qu'illustré figure 5.20.

**Précautions d'usage** Puisque nous ne considérons pas ici les méthodes originales des auteurs, les performances mesurées ne doivent pas être considérées comme représentatives de ces dernières. Nous utilisons notamment ici des implémentations non optimisées, aussi nous ne commenterons pas leurs temps de calcul, relativement importants. La démarche retenue consistant à suivre la même procédure de post-traitement pour l'ensemble des méthodes évaluées présente l'intérêt de permettre une comparaison des capacités de formulation d'hypothèses de pose des méthodes testées, et s'avère selon nous plus porteuse de sens qu'une comparaison des pipelines complets de chacune de celles-ci. Enfin, bien que les méthodes de [Drost et al. \(2010\)](#) et [Hinterstoisser et al. \(2012b\)](#) constituent des références de l'état de l'art, elles ne représentent pas le sommet de ce dernier. Diverses approches publiées depuis affichent en effet sur les jeux de données existants des performances supérieures, et le lecteur est renvoyé à la section 5.3 concernant le positionnement de notre méthode par rapport à celles-ci.

#### 5.2.4 Prise en compte des symétries

Une des thèses majeures de nos travaux concerne l'importance de la prise en compte des symétries propres des objets, et nous avons notamment consacré le chapitre 3 à celle-ci. Afin de la valider, nous mettons en œuvre deux variantes des différentes méthodes évaluées : l'une supposant que l'objet ne présente pas de symétrie propre (*nosym*), et l'autre prenant ces symétries en compte (*sym*). Nous utilisons la même collection de patrons 2D pour *LINEMOD+* dans les deux configurations afin de ne pas biaiser la comparaison<sup>13</sup>. Les résultats de cette comparaison sont disponibles tableau 5.19 et soutiennent notre hypothèse, en affichant pour les objets symétriques des performances moyennes supérieures pour l'ensemble des métriques dans le cas de la version *sym* comparé à la version *nosym*, notamment avec un gain de Précision Moyenne de 100% pour notre approche dans sa version brute, et 62% dans sa version avec post-traitement.

#### 5.2.5 Pertinence de données de synthèse pour l'évaluation

Notre méthodologie d'évaluation s'appuie fortement sur l'usage de jeux de données synthétiques, en raison de leur facilité de génération. Cela sou-

13. Il s'agit là d'une approche quelque peu artificielle car ces patrons constituent des points de vue de référence de l'objet, et il conviendrait dans le cas d'une application réelle de plutôt réduire la redondance de ceux-ci en prenant en compte les symétries afin de limiter le temps de calcul.

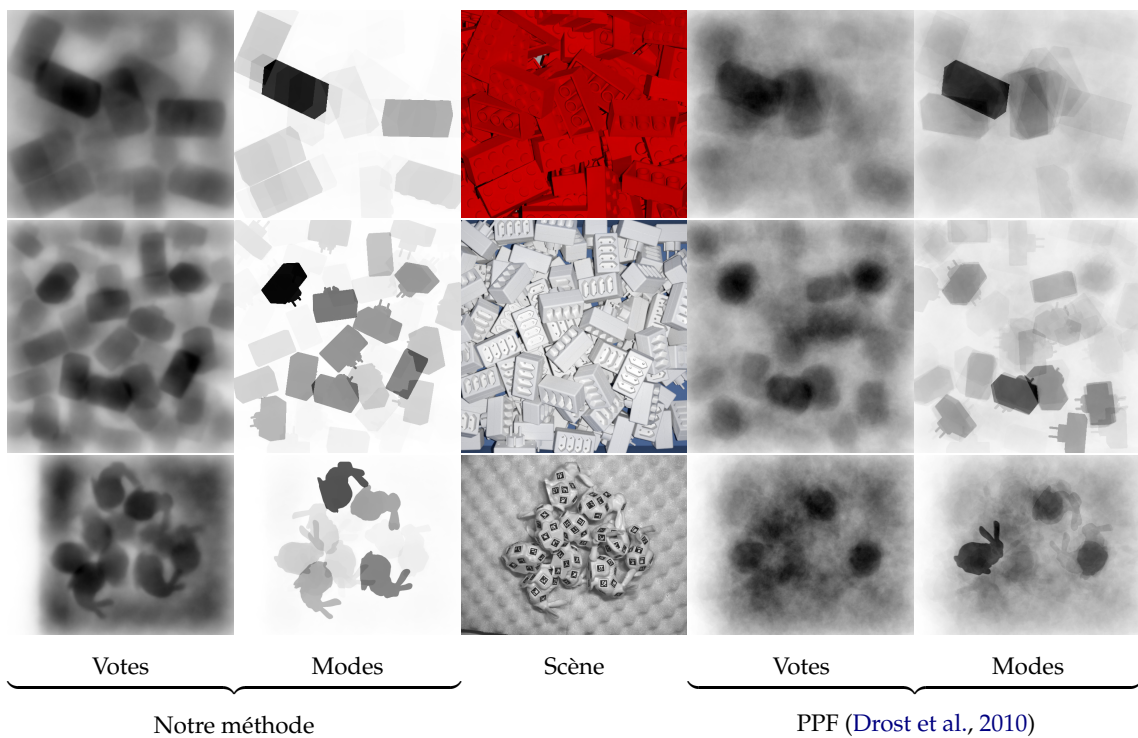


FIGURE 5.20 – Comparaison qualitative entre les distributions de votes générées par la méthode de [Drost et al. \(2010\)](#) et notre approche. La distribution de votes générée par notre méthode semble moins bruitée et présenter des modes plus marqués que celle de [Drost et al. \(2010\)](#).

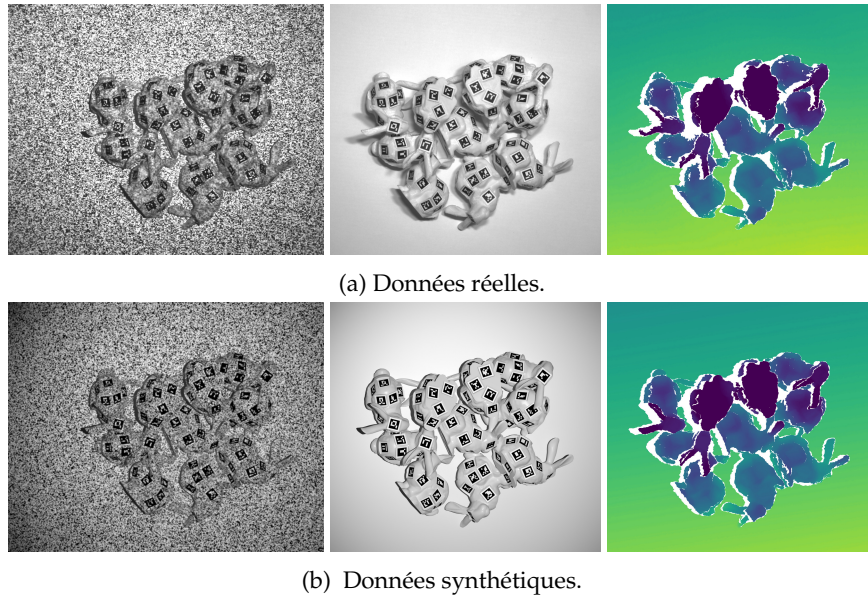


FIGURE 5.21 – Comparaison entre données réelles et données synthétiques représentant une scène similaire. Les données synthétiques sont générées sur la base des annotations de pose des données réelles, obtenues par détection de marqueurs fiduciaires. **Gauche** : image d'intensité avec éclairage texturé servant à la reconstruction 3D. **Centre** : image d'intensité. **Droite** : image de profondeur reconstruite.

lève donc la question de la pertinence de telles données pour l'évaluation.

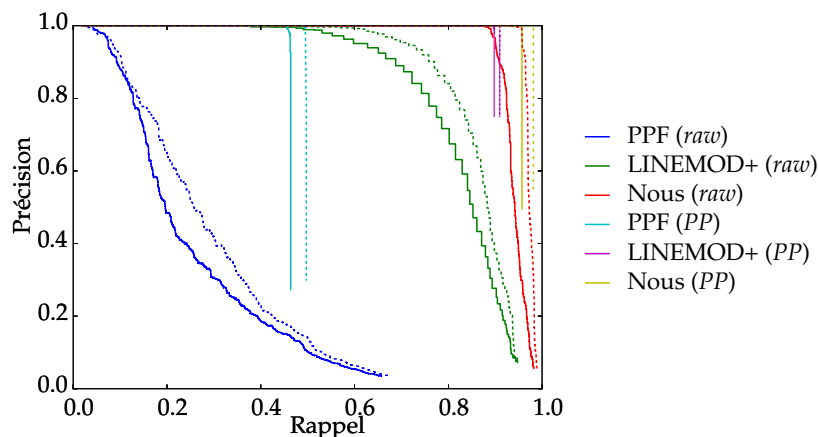
Afin d'aborder cette question, nous produisons à des fins de comparaison un jeu de données synthétique correspondant à des scènes similaires à celles d'un jeu de données réel (*markers flat*), composé de 308 scènes d'instances de lapins en vrac sur un fond plat. Pour chaque scène réelle, nous produisons automatiquement une scène virtuelle présentant des instances d'objets positionnées de manière similaire en exploitant les annotations de pose (obtenues suivant l'approche décrite section 5.1.2.1). Nous synthétisons alors une acquisition 3D de celle-ci à l'aide de la technique de simulation de capteur présentée section 5.1.2.2, en reproduisant les caractéristiques géométriques du capteur réel (résolution, focale, point principal, écartement entre les caméras notamment). La figure 5.21 présente un exemple de données synthétiques ainsi produites, associées aux données réelles correspondantes.

Nous utilisons dans nos simulations des modèles de matériaux et d'éclairage relativement grossiers : ici un matériau constitué d'une composante diffuse et d'un touche de composante réfléchive<sup>14</sup>, associé avec une source quasi-ponctuelle directionnelle de lumière. Aussi, les rendus synthétiques sont distinguables relativement facilement des images réelles pour un humain, et en cela les images de profondeur générées sont nécessairement différentes des données réelles. Dans notre expérience, les image de pro-

14. Nœud *glossy* de Blender Cycles (Blender Online Community, 2017).

AP	Approche brute ( <i>raw</i> )			Avec post-traitement ( <i>PP</i> )		
	PPF	LINEMOD+	Nous	PPF	LINEMOD+	Nous
Données réelles	0.26	0.83	0.94	0.46	0.90	0.96
Données synthétiques	0.29	0.87	0.97	0.50	0.91	0.98

(a) Précision Moyenne de détection et de localisation des instances occultées à moins de 50%.



(b) Courbes précision-rappel correspondantes obtenues sur le jeu de données réel (traits pleins) et le jeu de données synthétique (traits pointillés).

FIGURE 5.22 – Validation de l’usage de données synthétiques pour l’évaluation. Comparaison des performances obtenues sur un jeu de données réel et un jeu de données synthétique représentant les mêmes scènes. Bien que les performances obtenues soient systématiquement meilleures sur le jeu de données synthétique que sur le jeu réel, les différentes méthodes se comparent entre elles de manière similaire sur les deux jeux de données, corroborant l’usage de données synthétiques à des fins comparatives.

fondeur synthétiques sont ainsi légèrement de meilleure qualité que leurs homologues réelles, avec un taux de pixels pourvu d’information de profondeur de 96.6%, contre 96.2% pour les données réelles. Cette différence a un effet sur les performances mesurées. Nous évaluons en effet les performances des trois méthodes d’estimation de pose sur ces deux jeux de données, en testant pour chacune les deux variantes *raw* et *PP* décrites précédemment. La figure 5.22 présente les performances mesurées ; et nous observons que chaque méthode évaluée obtient de meilleures performances sur le jeu de données synthétique que sur le jeu de données réel.

Cependant, les courbes de performances obtenues sur les deux jeux de données sont relativement proches, et les méthodes testées se comparent de manière très similaires les unes aux autres sur les deux jeux de données, ce qui conduit à des conclusions identiques vis-à-vis des performances relatives des différentes méthodes. De plus, si les cartes de profondeur synthétiques ne sont pas identiques à celles réelles, elles n’en demeurent pas moins visuellement plausibles. Aussi, ces constats nous amènent à considérer nos

jeux de données synthétiques comme *suffisamment réalistes* pour des fins d'évaluation comparative.

### 5.3 Expérimentations sur le jeu de données LINE-MOD

Les limitations des méthodologies d'évaluation de la littérature nous ont conduit à proposer notre propre protocole d'évaluation adapté à la problématique du débrassage. Les protocoles existants n'en sont néanmoins pas dénués d'intérêt pour autant, ne serait-ce que de par leur utilisation comme référence du domaine. Aussi, afin de positionner nos travaux relativement à l'état de l'art nous évaluons notre approche sur le jeu de données LINEMOD de [Hinterstoisser et al. \(2012b\)](#) suivant la méthodologie proposée par les auteurs. Ce jeu de données est certainement le plus utilisé dans le cadre de publications académiques, et est constitué d'images RGBD acquises au moyen d'un capteur Kinect, représentant des scènes où divers objets colorés sont disposés sur une face donnée sur un bureau présentant un certain fouillis. Il cible le problème de *localisation* d'une instance d'objet particulière au sein d'une image, et est évaluée en terme de *taux de reconnaissance*, c.-à-d. de pourcentage de scènes pour lesquelles l'objet d'intérêt a été correctement localisé.

Lors de l'apprentissage, nous nous adaptons aux spécificités du jeu de données en introduisant un a priori concernant la distribution de pose relativement à la caméra. Suivant l'approche définie section 4.6.2, nous ne générons de données d'apprentissage que pour des poses où l'objet est vu « du dessus » et « tête vers le haut », en ne considérant pour se faire que des points de vue provenant d'un seul hémisphère, et en ne tolérant pas d'inclinaison supérieure à 45° de l'objet par rapport à la verticale de la caméra.

**LINEMOD Dataset** Le tableau 5.23 synthétise les résultats obtenus par notre méthode sur ce jeu de données comparativement à l'état de l'art. Ces résultats sont illustrés figure 5.24. De même que section 5.2, nous présentons les performances d'une version brute (*raw*) notre méthode – qui retourne le mode principal de la distribution de pose générée par la forêt de décision – et une version présentant un post-traitement (*PP*) consistant ici à extraire les 100 modes principaux de la distribution pour estimer leur score de vraisemblance et sélectionner le meilleur. Aucun raffinement de pose n'a été mis en place pour ces expériences du fait de la précision satisfaisante des hypothèses de pose retournées par notre méthode.

Notre approche s'avère compétitive sur ce jeu de données, affichant un taux de reconnaissance supérieur à 96% pour tous les objets sauf *glue*, et présentant un taux de reconnaissance moyen supérieur aux méthodes de l'état de l'art en estimation de pose au moment de la rédaction. Le taux de reconnaissance moindre (89.8%) obtenu pour *glue* s'explique par le manque de caractéristiques fortes de forme 3D pour cet objet, qui combiné à sa faible visibilité dans les données de profondeur (cf. figure 5.25) rend difficile sa localisation uniquement à partir de cette modalité. Cette explication

TABLE 5.23 – Taux de reconnaissance (%) obtenu sur le jeu de données de Hinterstoisser et al. (2012b), mesuré suivant la méthodologie de ceux-ci.

	Nous ( <i>raw</i> )	Nous ( <i>PP</i> )	Hinterstoisser et al. (2016)	Kehl et al. (2016)	Brachmann et al. (2014)*	Rios-Cabrera et Tuytelaars (2013)	Hinterstoisser et al. (2012b)	Drost et al. (2010)†
Ape	97.7	98.2	<b>98.5</b>	96.9	95.0	85.4	95.8	86.5
Bench vise	98.8	<b>99.9</b>	99.8	94.10	98.9	98.9	98.7	70.7
Camera	90.8	98.5	<b>99.3</b>	97.7	98.2	92.1	97.5	78.6
Watering can	94.1	<b>99.2</b>	98.7	95.2	96.3	84.4	95.4	80.2
Cat	98.8	<b>99.9</b>	99.9	97.4	99.1	90.6	99.3	85.4
Driller	95.5	<b>99.8</b>	93.4	96.2	94.3	99.7	93.6	87.3
Duck	97.8	98.0	<b>98.2</b>	97.3	94.2	92.7	95.9	46.0
Hole puncher	95.5	96.8	<b>98.1</b>	96.8	97.5	97.9	95.9	77.4
Iron	90.1	98.7	98.3	98.7	98.4	<b>98.8</b>	97.5	84.9
Lamp	94.1	96.1	96.0	96.2	<b>97.9</b>	97.6	97.7	93.3
Phone	92.6	98.5	<b>98.6</b>	92.8	88.3	86.1	93.3	80.7
Bowl	99.9	<b>100</b>		99.9	99.7		99.9	95.7
Cup	95.6	<b>99.8</b>		99.6	97.5		97.1	68.4
Box	99.6	<b>99.9</b>	98.8	99.9	99.8	91.1	99.8	97.0
Glue	78.9	89.8	75.4	78.6	96.3	87.9	91.8	57.2
Moyenne	94.6	<b>98.2</b>	96.4	95.8	92.6	96.8	96.6	79.3

\* : Résultats avec un apprentissage sur des données synthétiques sans apriori selon lequel l'objet reposerait sur un plan.

† : d'après les expérimentations de Hinterstoisser et al. (2012b).

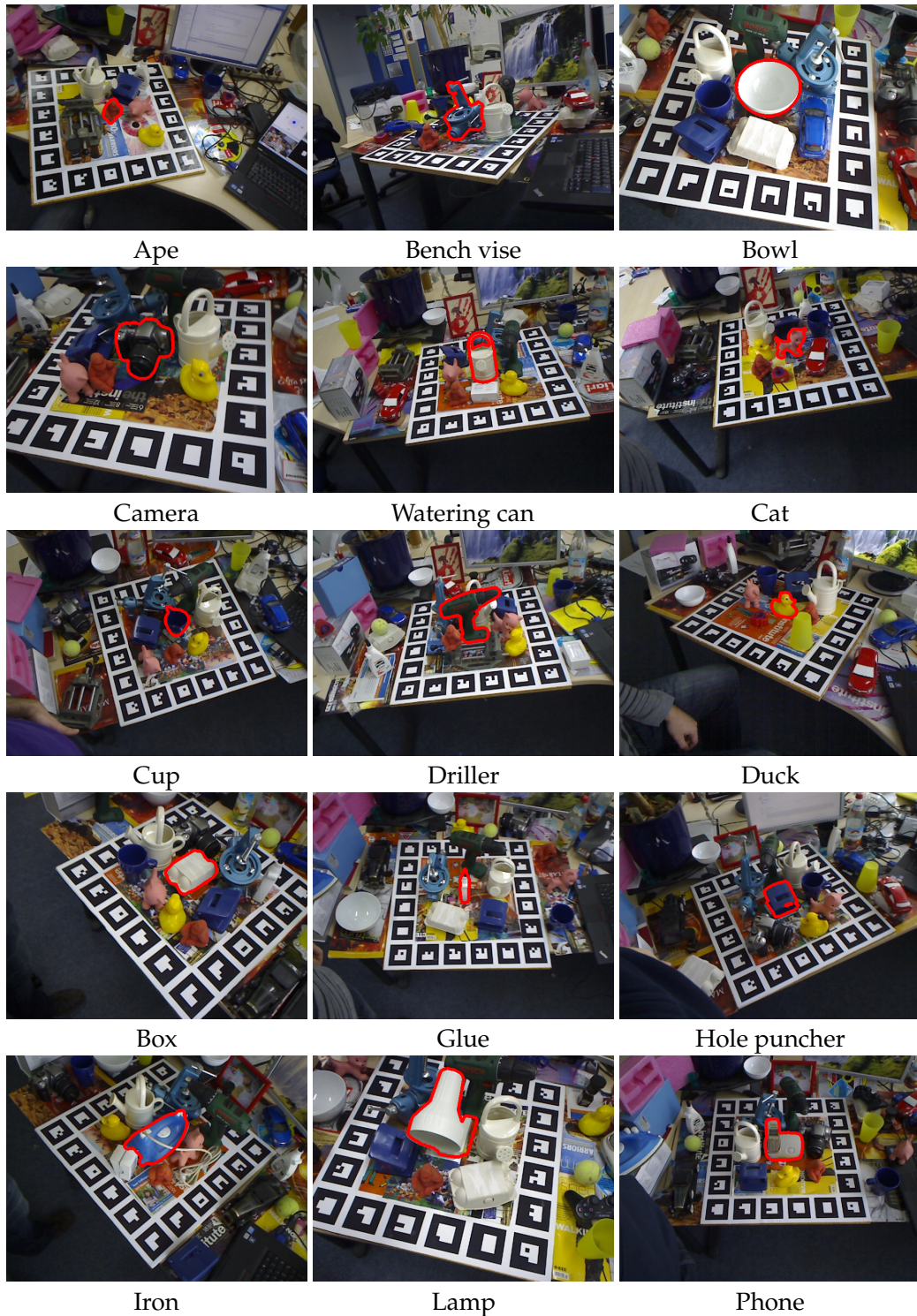


FIGURE 5.24 – Exemple de résultats de notre méthode pour les différents objets du jeu de données LINEMOD, sans aucun raffinement de pose.



FIGURE 5.25 – Le bruit et la faible visibilité de l’objet *glue* (tube de colle blanc au centre) au sein des données 3D du jeu LINEMOD rendent sa localisation difficile uniquement à partir de cette modalité (à droite). De gauche à droite : image RGB, nuage de points coloré, nuage de point (visualisation par *splatting* et *ambient occlusion*).



FIGURE 5.26 – Échecs de localisation typiques observés sur le jeu de données LINEMOD (l’objet à localiser est au centre de la planche de marqueurs).

est appuyée par les scores moindres également obtenus par [Drost et al. \(2010\)](#) et [Hinterstoisser et al. \(2016\)](#) pour cet objet, en ce que ces méthodes comme la nôtre n’exploitent pour générer des hypothèses de pose que l’information de profondeur, ici particulièrement pauvre. La figure 5.26 illustre des cas d’échecs typiques de notre approche sur ce jeu de données, attribuables à une confusion de l’objet d’intérêt avec une zone de fouillis présentant des caractéristiques de profondeur pouvant rappeler la forme de ce dernier. Cette confusion n’est pas étonnante en ce que notre approche a pour hypothèse que l’ensemble des points 3D de la scène appartiennent à des instances de l’objet d’intérêt. Cette hypothèse est largement violée ici puisque les images consistent en des scènes de fouillis divers dont seul une faible portion correspond à l’objet, et notre méthode n’a jamais été entraînée à distinguer l’objet du reste de l’image.

**Occlusion Dataset** Nous évaluons également notre méthode sur le jeu *Occlusion Dataset*, constitué d’une fraction du jeu de données LINEMOD ayant été réannoté par [Brachmann et al. \(2014\)](#) avec la pose de plusieurs objets parfois très occultés – voire même complètement non visibles. Cette occultation importante rend ce jeu de données plus difficile, et fait du taux de reconnaissance de 100% un objectif inatteignable en pratique. Ainsi que synthétisé dans le tableau 5.27, les performances obtenues par notre ap-



TABLE 5.27 – Taux de reconnaissance (%) obtenu pour les différents objets du jeu de données *Occlusion Dataset* (Hinterstoisser et al., 2012b; Brachmann et al., 2014), mesuré suivant la méthodologie de Hinterstoisser et al. (2012b).

	Nous ( <i>raw</i> )	Nous ( <i>PP</i> )	Hinterstoisser et al. (2016) <sup>‡</sup>	Krull et al. (2015) <sup>*</sup>	Brachmann et al. (2014) <sup>†</sup>	Hinterstoisser et al. (2012b) <sup>‡</sup>
Ape	65.0	65.4	<b>81.4</b>	77.9	62.6	49.8
Watering can	80.1	88.0	<b>94.7</b>	86.6	80.2	51.2
Cat	45.0	48.4	55.2	<b>55.6</b>	50.0	34.9
Driller	60.5	74.0	86.0	<b>93.6</b>	84.3	59.6
Duck	61.9	64.0	<b>79.7</b>	71.9	67.6	65.1
Hole puncher	85.1	88.3	<b>95.5</b>	94.8	89.9	67.2
Box	45.7	49.6	<b>65.6</b>	35.6	8.5	39.6
Glue	40.8	50.2	52.1	<b>67.9</b>	62.8	23.3
Moyenne	60.5	66.0	<b>76.3</b>	73.0	63.2	48.8

\* : moyenne obtenue suivant les différents apprentissages.

† : selon Krull et al. (2015).

‡ : selon Hinterstoisser et al. (2016).

proche sur ce jeu sont correctes au regard de l'état de l'art, mais restent néanmoins bien inférieures à celles annoncées par Hinterstoisser et al. (2016) et Krull et al. (2015). L'approche de Krull et al. (2015) exploite l'information de couleur contrairement à la nôtre, et utilise une étape de validation des hypothèses de pose spécifiquement entraînée pour être robuste aux occultations, ce qui peut expliquer sa supériorité. Hinterstoisser et al. (2016) utilisent quant eux une représentation des données sous forme de nuage de points. Si l'occultation partielle d'un objet engendre une absence partielle d'information pour ce dernier, elle n'introduit pas de difficulté particulière de distinction entre occulté et occulteur dans une telle représentation. Au contraire, notre approche dépendante du point de vue suppose implicitement la non occultation de l'objet, et tend à être d'avantage perturbée par les occultations, ce qui peut expliquer sa moindre performance sur ce jeu de données.

## 5.4 Caractérisation de notre forêt de décision

Dans cette section, nous cherchons à caractériser certaines propriétés de notre approche de génération d'hypothèses de pose à partir d'une forêt d'arbres de décision. Nous réalisons pour ce faire différentes expériences sur notre jeu de données de scènes de *bunny* avec la version brute (*raw*) de notre méthode introduite section 5.2.1, en faisant varier les paramètres

de cette forêt.

L'apprentissage d'un arbre de décision est un processus fondamentalement stochastique, aussi les performances obtenues peuvent-elles varier entre deux forêts de décision apprises avec des paramètres similaires. Afin d'être en mesure de discriminer entre observations statistiquement significatives et fluctuations aléatoires, nous procédons à plusieurs apprentissages indépendants pour chaque configuration.

En l'absence de précisions contraires, nous considérons ici une forêt constituée de 5 arbres de décision de profondeur maximale 20 dont les classifieurs faibles sont appris chacun à partir de l'évaluation de 30 descripteurs choisis aléatoirement. Les performances sont quant à elles quantifiées au moyen de la Précision Moyenne, en considérant pour tâche la détection et la localisation des instances moins de 50% occultées.

#### 5.4.1 Importance de la silhouette dans la détection et l'estimation de pose

Une des hypothèses ayant conduit au développement de notre méthode est que de l'information utile pour la détection et l'estimation de pose est stockée au niveau de la silhouette de l'objet, ou plus précisément dans le fait que les instances d'intérêt se détachent souvent du fond, en retrait par rapport à celles-ci.

Nous testons cette hypothèse en comparant les performances obtenues sur le jeu de données *bunny* avec deux variantes. La première consiste en la méthode présentée au chapitre 4, selon laquelle on réalise l'apprentissage de l'objet au moyen de vues synthétiques de l'objet sur un fond situé à une profondeur infinie (c.-à-d. suffisante pour saturer la sortie des descripteurs utilisés). L'attribution d'une profondeur au fond permet de spécifier lors de l'apprentissage que la silhouette de l'objet se détache généralement de ce dernier. La seconde variante n'attribue elle pas de profondeur définie au fond lors de l'apprentissage, de sorte qu'il n'est pas possible d'apprendre la silhouette de l'objet et que l'apprentissage doit alors uniquement se baser sur les variations de profondeur à l'intérieur de celle-ci. La figure 5.28 présente les performances obtenues avec 5 apprentissages différents pour chacune de ces deux variantes, sous forme de courbes précision-rappel.

Comme on peut l'observer figure 5.28a, les 5 apprentissages correspondant à la variante ayant la capacité d'apprendre la silhouette de l'objet affichent dans cette expérience une performance systématiquement supérieure aux 5 autres lorsque l'on considère une forêt composée d'un seul arbre de décision. L'écart de Précision Moyenne observé (de respectivement 0.77 pour la première variante contre 0.71 pour la seconde) peut être considéré comme significatif devant la variabilité due à l'apprentissage<sup>15</sup>. Ce résultat suggère que le pouvoir discriminant d'un arbre de décision est en effet plus grand lorsque ce dernier a appris à reconnaître le saut de profondeur typique observé au niveau de la silhouette de l'objet. Cette supériorité en terme de Précision Moyenne est également observée lors de l'utilisation d'une forêt de décision comportant d'avantage d'arbres (AP moyenne pour

15. Un test de permutation permet de rejeter l'hypothèse d'équivalence des deux méthodes avec une valeur-p symétrique de deux chances sur 5-parmi-10, soit environ 8%.

des forêts de 5 arbres de 0.83 contre 0.82), comme illustré figure 5.28b. Cependant, on constate également que la variante ne prenant pas la silhouette de l'objet en considération présente un taux de rappel maximum légèrement supérieur à l'autre ( $0.886 \pm 0.004$  contre  $0.879 \pm 0.002$ ).

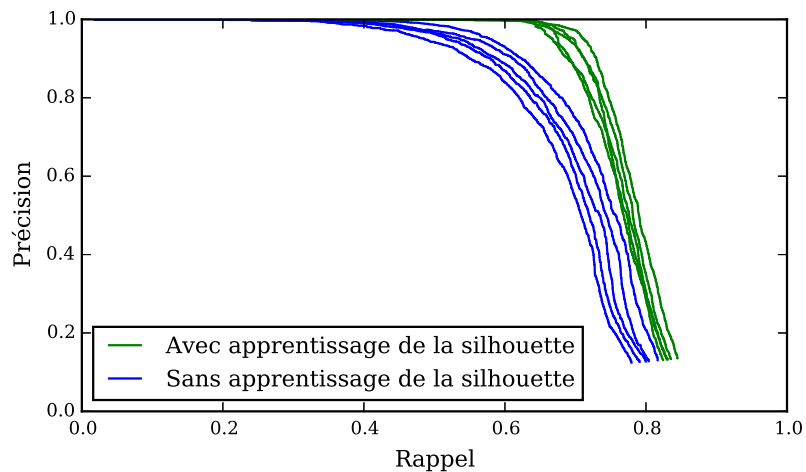
La figure 5.29 présente une comparaison qualitative de ces deux variantes qui permet de mieux appréhender ces résultats. Ainsi qu'illustré, la forêt de décision apprise avec l'information que la silhouette de l'objet se détache du fond (figure 5.29c) semble présenter un pouvoir discriminant plus important que l'autre (figure 5.29b), car elle produit une distribution de pose moins étalée d'où des modes correspondant à des instances d'objets ressortent plus clairement, ce qui est favorable à la précision des résultats. En revanche, celle-ci a plus de mal à localiser les instances d'objet se détachant peu du fond – telles que celles fléchées en rouge sur la figure, ce qui limite le taux de rappel atteignable. Cette limitation a néanmoins peu d'impact dans le cadre d'une application pratique de dévracage où on se satisfait de la localisation d'un nombre limité d'instances, et les résultats expérimentaux figure 5.28c suggèrent que l'apprentissage de la silhouette de l'objet améliore significativement les performances dans ce scénario.

## 5.4.2 Effort d'apprentissage

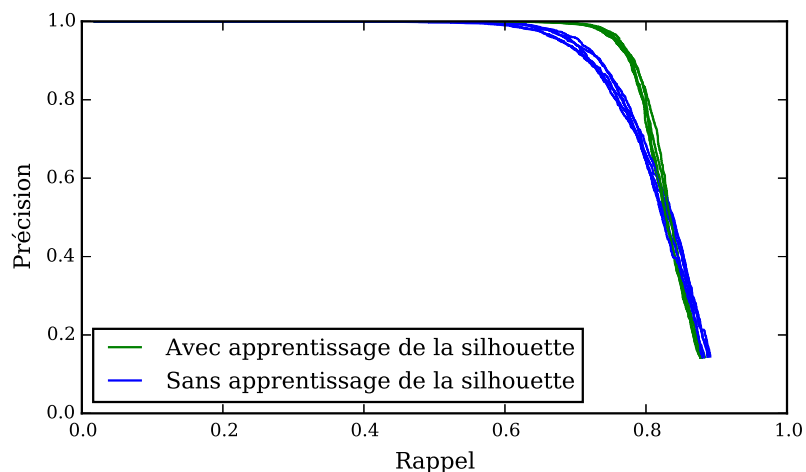
Ainsi que décrit section 4.6.3, notre procédure d'apprentissage revient à tester sur un ensemble d'exemples d'apprentissage différents classifieurs faibles choisis aléatoirement, afin de sélectionner celui apportant le plus grand gain d'information. L'objectif visé par cette démarche est d'atteindre de meilleures performances lors de la phase de test en détection et estimation de pose que ne l'aurait permis un choix purement aléatoire de classifieur.

Nous validons la pertinence de cette approche en étudiant l'influence sur les performances finales du nombre de descripteurs faibles testés pour chaque nœud lors de l'apprentissage d'un arbre de décision. Pour chaque descripteur, 5 valeurs de seuil distinctes sont envisagées afin de définir 5 classifieurs faibles. Le nombre de classifieurs caractérise en quelque sorte l'effort déployé lors de l'apprentissage, dont la durée est de complexité linéaire en cette grandeur (cf. figure 5.30).

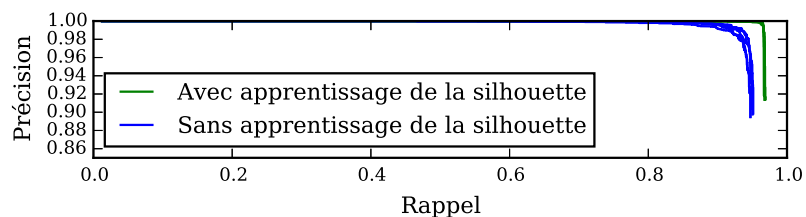
La figure 5.31 synthétise les résultats obtenus dans une configuration avec un seul arbre de décision, ainsi que dans le cas d'une forêt de 5 arbres. Pour des raisons de lisibilité, la performance d'un apprentissage est quantifiée au moyen d'une grandeur unique : la Précision Moyenne de détection et d'estimation de pose des instances moins de 50% occultées. Comme on l'observe sur la figure, l'augmentation de l'effort d'apprentissage induit une amélioration de la mesure de performance évaluée dans le cadre de notre expérimentation, ce qui valide la pertinence de notre méthode d'apprentissage. On observe également une diminution de la variabilité des performances obtenues entre différents apprentissages avec l'augmentation du nombre de descripteurs testés. Notamment, l'écart-type de Précision Moyenne obtenu avec des forêts d'un seul arbre passe de 5% en ne testant qu'un seul descripteur à 0.8% dans le cas d'un apprentissage où 30 descripteurs sont évalués par nœud. Ce phénomène est cohérent avec l'idée que l'augmentation de l'effort d'apprentissage conduit à la sélection d'arbres moins *aléatoires* et



(a) Avec 1 arbre de décision.

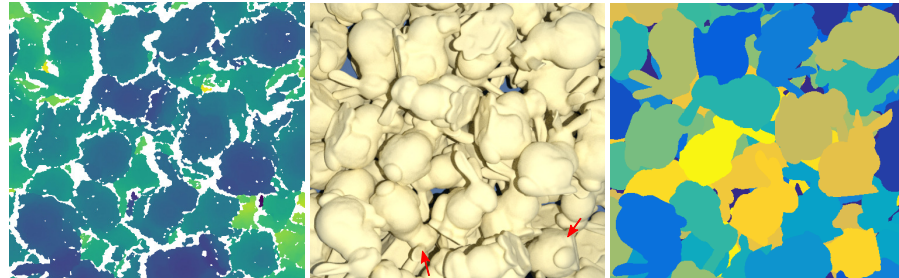


(b) Avec 5 arbres de décision.

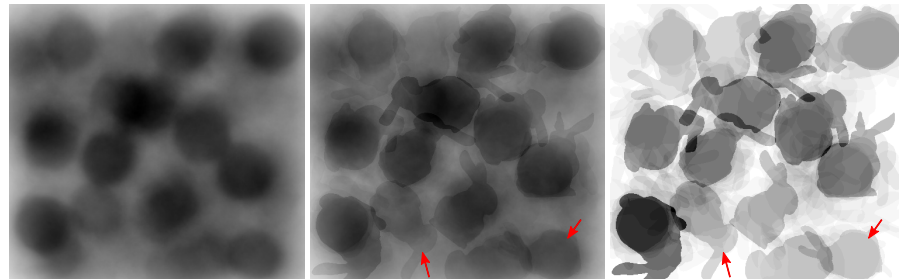


(c) Renvoi d'au plus 10 hypothèses de pose. 5 arbres de décision.

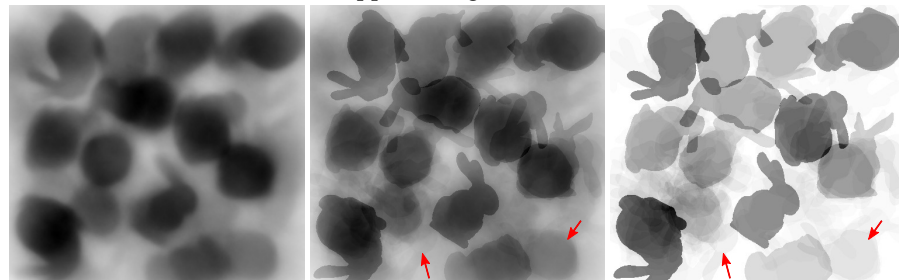
FIGURE 5.28 – Comparaison quantitative des performances obtenues, suivant que soit spécifié ou non dans les données d'apprentissage que la silhouette de l'objet se détache du fond. Performances de détection et d'estimation de pose des instances présentant moins de 50% d'occultation sur le jeu de données *bunny*. L'apprentissage a été réalisé 5 fois pour chaque variante avec des initialisations distinctes du générateur de nombre pseudo-aléatoire.



(a) Image de profondeur d'entrée, et images RGB et de segmentation associées à des fins de visualisation.



(b) Sans apprentissage de la silhouette.



(c) Avec apprentissage de la silhouette.

FIGURE 5.29 – Comparaison qualitative de variantes avec ou sans apprentissage de la silhouette de l'objet (exemples utilisant des forêts de 5 arbres de décision). (b,c) **À gauche** : distribution de pose générée par la forêt de décision. **Au centre** : agrégation de poses ayant convergé vers les modes de densité de la distribution initiale. **À droite** : 100 modes principaux de la distribution pondérés par leur pseudo-densité de probabilité. Les modes correspondant à des hypothèses de pose valides contrastent mieux avec le bruit de fond lorsque la forêt est en mesure d'apprendre la silhouette de l'objet (c) que lorsque ce n'est pas le cas (b), à l'exception des modes correspondant à des instances ne se détachant que peu du fond dans l'image de profondeur (flèches rouges).

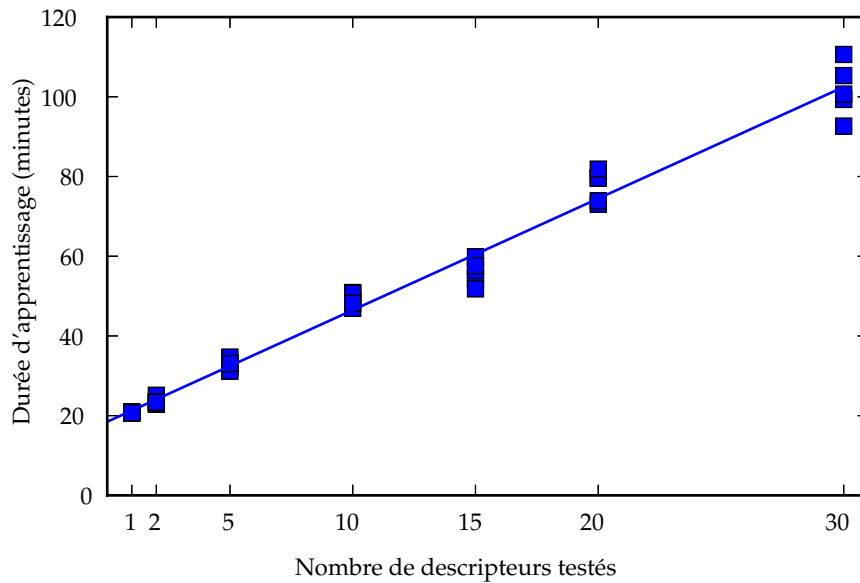


FIGURE 5.30 – Durée d'apprentissage d'un arbre en fonction du nombre de descripteurs évalués pour chaque nœud durant la phase d'apprentissage. La durée d'apprentissage présente une complexité linéaire en ce paramètre, qui exprime en quelque sorte l'effort d'apprentissage réalisé afin de sélectionner le classifieur le plus pertinent.

donc à des performances plus répétables. De manière empirique, le gain marginal de performance décroît rapidement avec l'augmentation de l'effort d'apprentissage, de sorte qu'il n'est pas nécessairement efficace d'y consacrer des ressources très importantes. En pratique, nous nous limitons typiquement à tester 30 descripteurs par nœud lors de nos apprentissages, afin de limiter la durée de ceux-ci à des valeurs raisonnables (typiquement 1h40 – 2h par arbre sur un ordinateur de bureau standard).

### 5.4.3 Profondeur et nombre d'arbres

On vérifie également que la taille de la forêt de décision est corrélée positivement avec la performance, comme illustré figure 5.32. La variance de Précision Moyenne obtenue avec différents apprentissages diminue avec l'augmentation de la taille de la forêt (facteur 3 en passant 1 à 5 arbres, et facteur 7 en passant de 1 à 12 arbres), ce qui constitue un point important en terme de fiabilité. Le gain marginal de performance décroît cependant rapidement avec l'augmentation du nombre d'arbres, de sorte qu'une forêt de seulement 5 arbres permet en pratique d'assurer de bonnes performances tout en limitant la durée de l'apprentissage et les besoins en calcul et en mémoire lors de l'exécution à des niveaux modérés.

On vérifie de même que les performances augmentent avec la profondeur des arbres, notion qui traduit en quelque sorte le pouvoir discriminant de ces derniers (cf. figure 5.33). Nous sommes cependant contraints de limiter

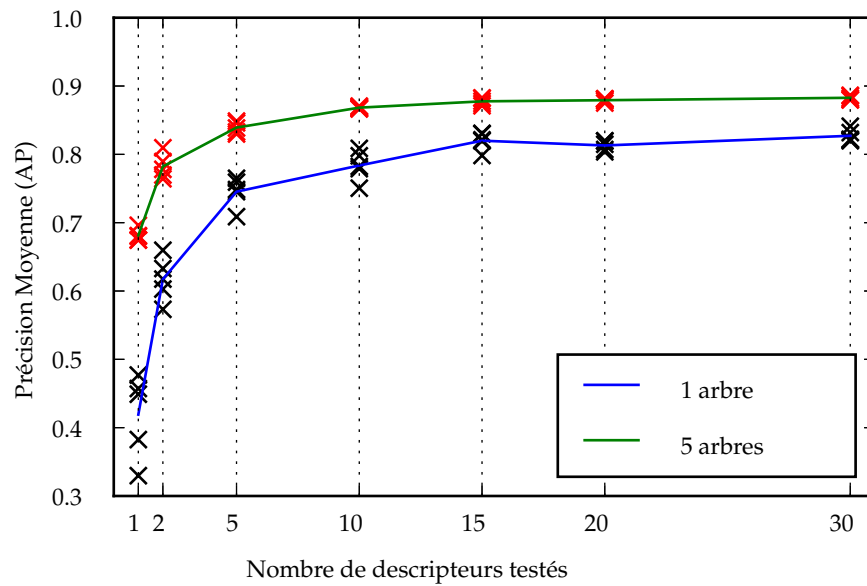


FIGURE 5.31 – Évaluation quantitative des performances obtenues selon le nombre de descripteurs faibles évalués pour chaque nœud durant la phase d'apprentissage des arbres de décision. Pour chaque configuration, on réalise 5 apprentissages avec des graines aléatoires différentes de manière à prendre en compte le caractère aléatoire de l'apprentissage d'une forêt de décision. L'augmentation du nombre de test permet d'augmenter la pertinence des classifieurs faibles sélectionnés suivant le critère de maximum de gain d'information, ce qui permet d'améliorer les performances. Cette augmentation de l'effort d'apprentissage réduit également le caractère aléatoire des arbres de décision obtenus, ce qui réduit la variance de performance pour un même paramétrage.

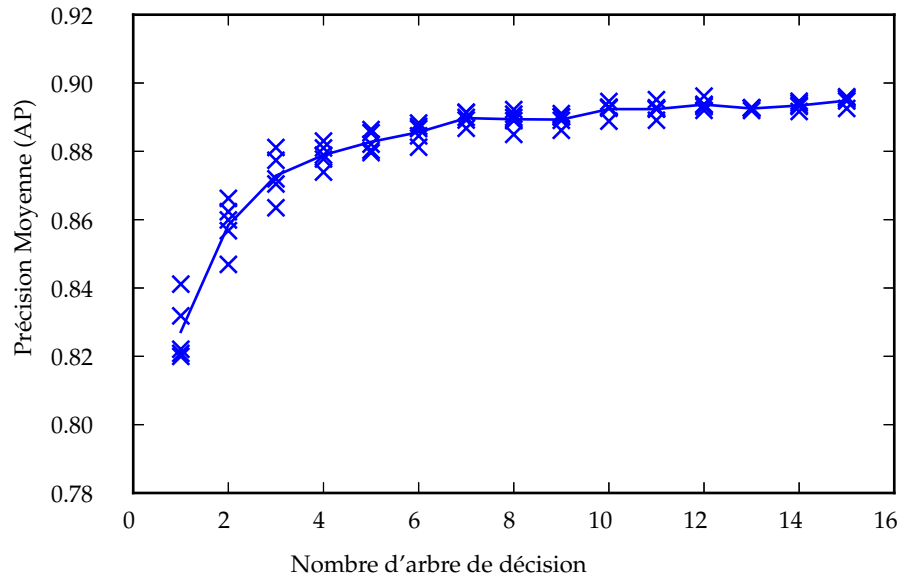


FIGURE 5.32 – Évolution des performances selon le nombre d'arbres utilisés dans la forêt de décision.

la profondeur des arbres de décision à 20 en pratique, de par la complexité mémoire exponentielle de leur apprentissage.

## 5.5 Influence de l'occultation

Notre méthodologie d'évaluation offre également l'occasion d'étudier l'impact de l'occultation partielle des instances d'objet sur les performances, ce que nous synthétisons figure 5.34 pour le jeu de données *bunny*.

Comme représenté, notre approche présente des performances quasi idéales pour la détection et l'estimation de pose des instances de *bunny* peu occultées (occultation inférieure à 10%) – tant en terme de précision que de rappel. Ces performances se dégradent progressivement avec l'augmentation de l'occultation des instances recherchées. Elles demeurent bonnes pour des occultations modérées (inférieure à 30%), mais chutent sensiblement au delà de 40% d'occultation – tout particulièrement en terme de précision. Notre approche présente alors des difficultés à discriminer les bonnes hypothèses de pose des mauvaises. Cela est assez compréhensible<sup>16</sup>, dans la mesure où l'interprétation de données ne représentant qu'une fraction limitée d'objet peut facilement être ambiguë.



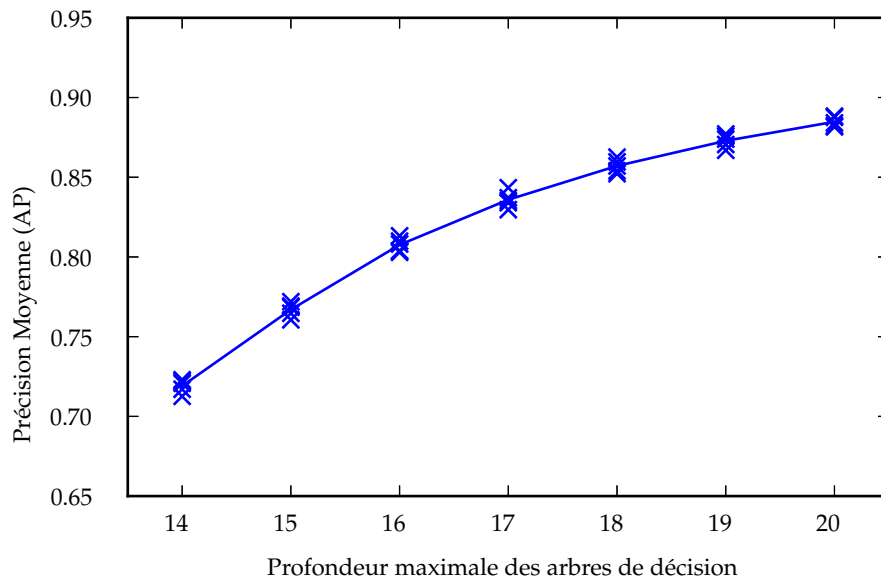


FIGURE 5.33 – Évolution des performances selon la profondeur maximale des arbres de décision.

## 5.6 Robustesse au bruit

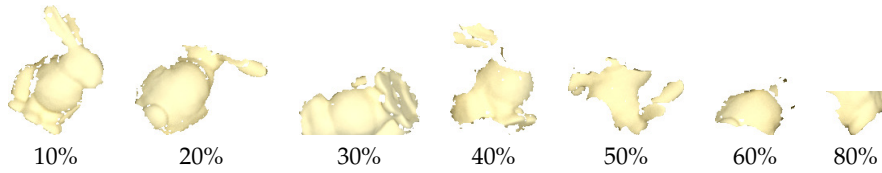
Nous évaluons enfin la robustesse au bruit de notre méthode. Bien que les évaluations précédentes aient déjà été réalisées sur des données présentant un bruit correspondant à un capteur stéréoscopique particulier, nous souhaitons ici étudier plus en détail l'influence du bruit sur les performances. Nous procédons pour ce faire au moyen d'images de profondeur synthétiques idéales du jeu de données *bunny*, auxquelles nous ajoutons artificiellement un bruit dont nous maîtrisons les propriétés.

Par commodité, nous nous limitons à un modèle de bruit simple composé de deux processus aléatoires :

- Un bruit blanc gaussien centré, d'écart-type noté  $\sigma$ , qui vient s'ajouter aux valeurs de profondeur.
- Un bruit blanc binaire, pouvant supprimer l'information de profondeur en un pixel avec une probabilité  $p_s$ .

Ce modèle est peu représentatif du bruit observable en sortie de capteurs réels, car ceux-ci incluent généralement une étape de post-traitement permettant de boucher les trous et lisser les données des images de profondeur. Il s'agit cependant d'un test de robustesse intéressant. La figure 5.35 présente une illustration des données ainsi générées, tandis que nous représentons figure 5.36 l'impact de ces deux types de bruits sur les performances (le lecteur est référé à l'annexe E pour une évaluation exhaustive des combinaisons de ces deux sources de bruits).

16. Ce qui n'exclut pas des possibilités d'amélioration pour autant.



(a) Illustration d'instances présentant différents taux d'occultation.

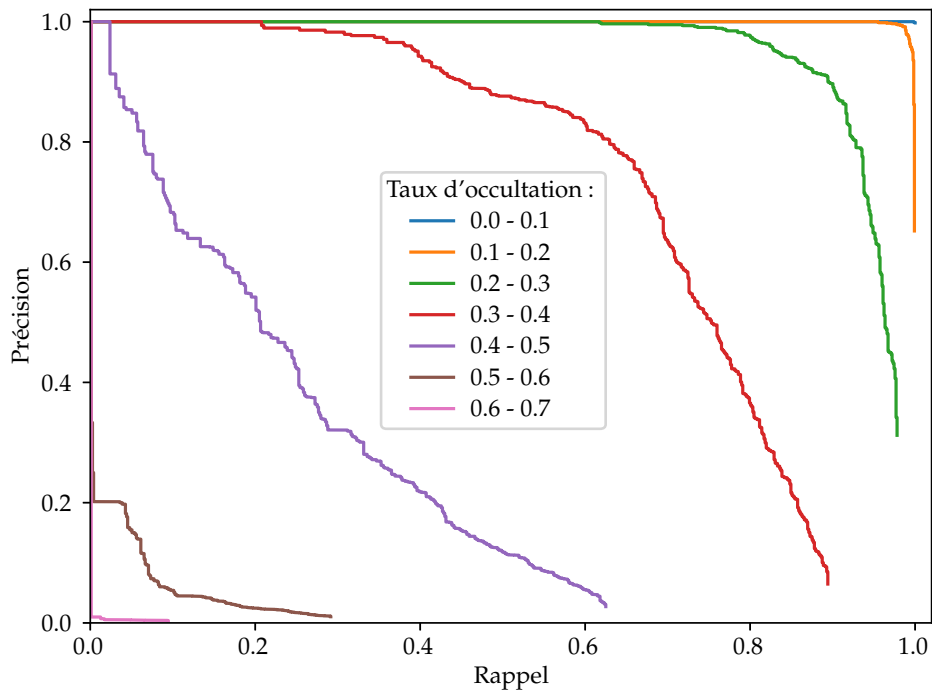
(b) Courbes précision-rappel (micro-moyennes) obtenues sur le jeu de données *bunny* pour différentes plages de taux d'occultation.

FIGURE 5.34 – Influence de l'occultation des instances d'objet sur les performances.

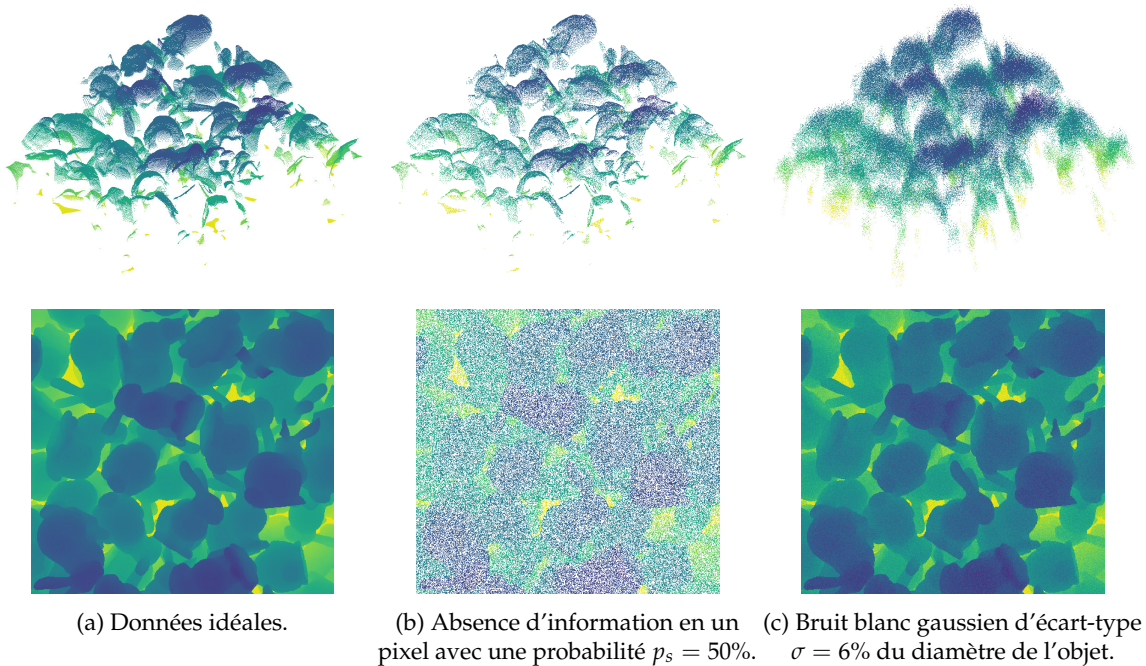


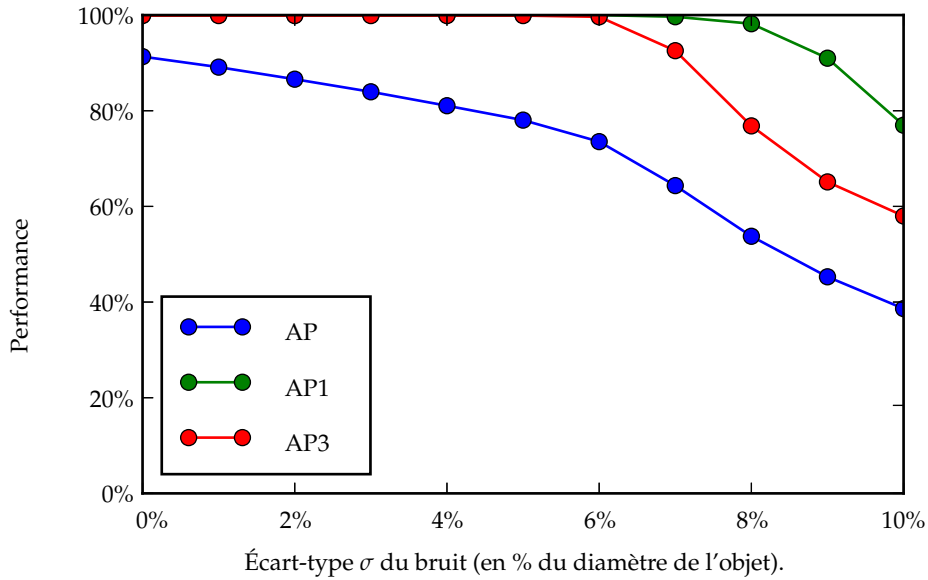
FIGURE 5.35 – Illustrations des modèles de bruit synthétiques appliqués aux données de profondeur.

Bien que l'on observe naturellement une diminution des performances avec l'augmentation du niveau de bruit, notre méthode s'avère relativement robuste à celui-ci. Elle est notamment encore en mesure d'atteindre une Précision Moyenne pour la détection d'une unique instance d'objet ( $AP1$ ) de 96.9% dans des scènes où l'information de profondeur est absente pour  $p_s = 20\%$  des pixels et bruitée avec un écart-type  $\sigma$  correspondant à 6% du diamètre de la sphère englobante de l'objet pour les autres. Il s'agit pourtant d'un niveau de bruit extrêmement important comparé à celui typiquement rencontré dans une application réelle, comme l'illustre la figure 5.37. Cette robustesse est attribuable à la technique de fusion des différents votes utilisée, formulée de sorte à être robuste aux valeurs aberrantes, et dont la figure 5.38 illustre le comportement.

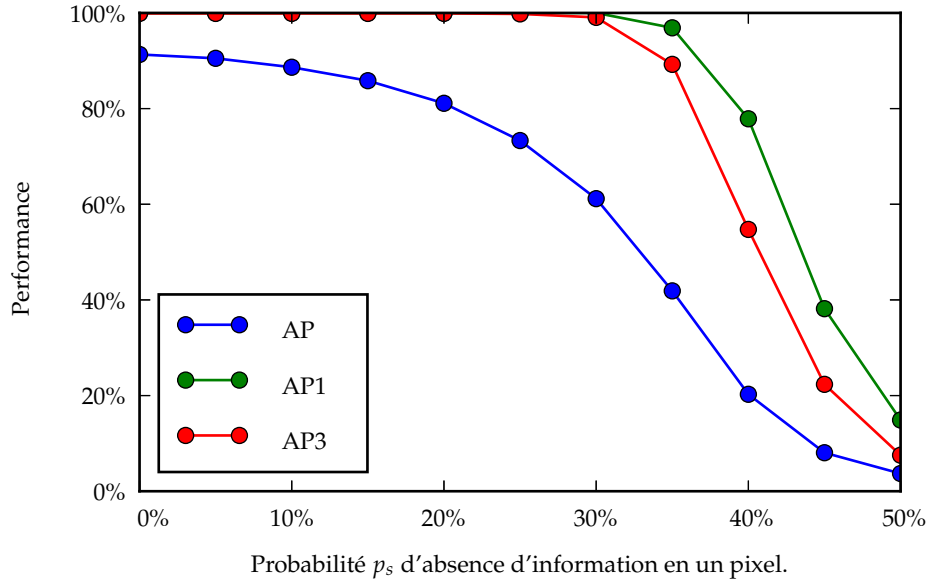
## 5.7 Synthèse

Le présent chapitre apporte un éclairage expérimental sur notre approche de détection et d'estimation de pose d'objet rigide. Nous y avons étudié les caractéristiques de la méthode introduite au chapitre 4, en testant notamment les hypothèses principales sur laquelle elle repose, que sont l'importance de la prise en compte des symétries des objets, et la présence d'information précieuse pour la reconnaissance dans la silhouette de l'objet.

Notre méthode se compare favorablement à l'état de l'art sur le jeu de données de référence LINEMOD, et nous avons également validé les bonnes



(a) Influence du bruit d'estimation de profondeur sur les performances.



(b) Influence de l'absence d'information sur les performances.

FIGURE 5.36 – Évaluation de la robustesse au bruit selon différentes métriques (Précision Moyenne (AP) et Précision Moyenne étant donné le renvoi d'au plus  $k$  résultats (AP $k$ ), pour  $k \in \{1, 3\}$ ).

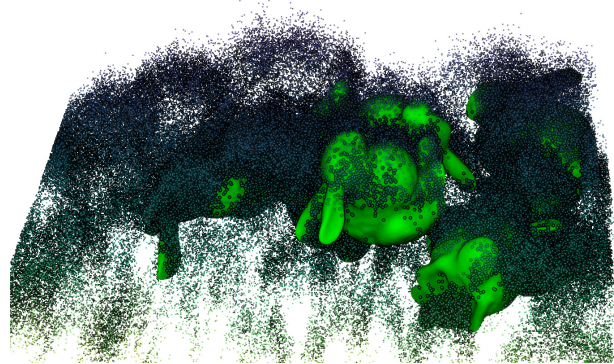


FIGURE 5.37 – Même en présence de niveaux de bruits élevés (ici  $p_s = 20\%$ ,  $\sigma = 6\%$  du diamètre de l'objet), notre méthode parvient à générer des hypothèses de pose pertinentes.

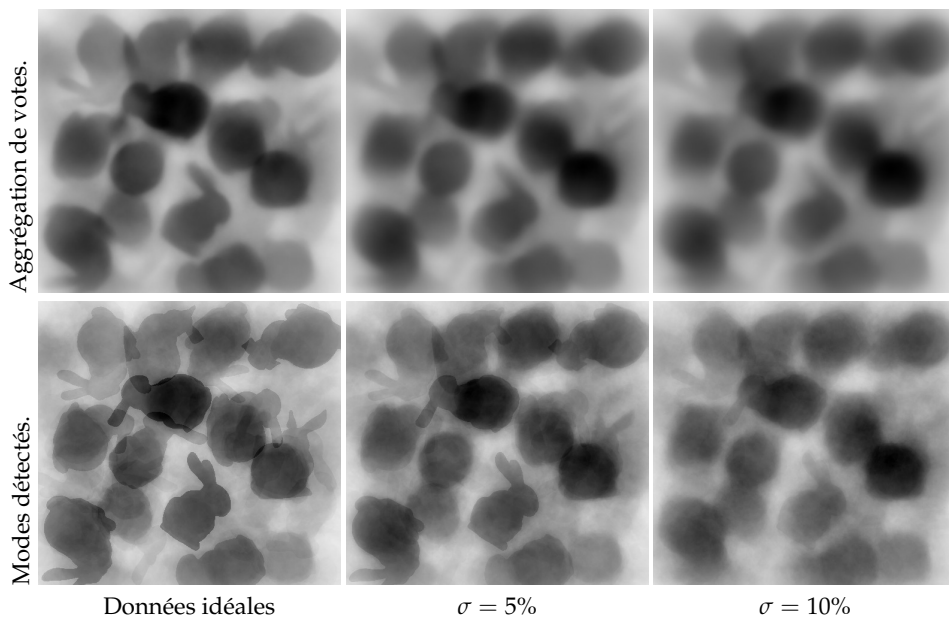


FIGURE 5.38 – Effet qualitatif du bruit. L'augmentation du niveau de bruit  $\sigma$  augmente la dispersion des votes du fait de l'incertitude de position des points de référence extraits, ainsi que la propension de votes aberrants, ce qui se traduit respectivement par une visualisation plus floue et moins contrastée de la distribution de probabilité de pose (en haut). La procédure de détection de modes (en bas) est en mesure de compenser pour partie cette dispersion (silhouettes nettes des instances d'objet se détachant du fond) mais voit le rapport signal-sur-bruit de ces modes se dégrader progressivement (diminution du contraste).

performances de cette dernière dans un scénario de dévracage, au moyen d'une méthodologie d'évaluation adaptée ainsi que de jeux de données introduits pour l'occasion.

Les évaluations menées mettent enfin en évidence la robustesse de notre approche, pour laquelle l'augmentation de l'effort d'apprentissage et du nombre d'arbres de décision conduit systématiquement à une augmentation des performances et où la fusion d'un grand nombre d'estimateurs introduit une résistance intrinsèque au bruit, sans apprentissage préalable de ce dernier.

## Chapitre 6

# Conclusion

### 6.1 Résumé

Nous étudions dans cette thèse le problème de la détection et de l'estimation de pose d'instances d'un objet rigide à partir de données visuelles. Nous traitons plus spécifiquement ce sujet au travers du prisme d'une application de dévracage robotisé, où la détection et la localisation de pièces en vrac constitue un préalable important à leur manipulation automatique. Cette dernière peut présenter des finalités aussi variées que des opérations d'usinage, d'assemblage, d'emballage, de contrôle qualité, de préparation de commandes, etc.

Nos efforts se focalisent sur le cas d'objets non texturés, qui représentent un scénario de grande importance pratique pour l'industrie, et nous développons dans cette optique une approche de détection et d'estimation de pose exploitant des données de profondeur. Celle-ci est formulée en termes probabilistes, où il s'agit d'estimer une distribution représentant la probabilité de présence d'une instance d'objet au voisinage de chaque pose. Cette estimation est réalisée par agrégation d'hypothèses, produites pour un ensemble d'estimateurs locaux au moyen d'une forêt de décision. Les poses correspondant aux maxima locaux de cette distribution sont alors extraites au moyen d'une procédure spécifique de Mean-Shift, et constituent des hypothèses pertinentes concernant la pose d'instances d'objet présentes dans la scène, qui sont retournées à la suite d'une procédure de raffinement et de filtrage.

Cette approche constitue selon nous un compromis intéressant au regard de l'état de l'art. Sa rapidité d'exécution – de l'ordre de 0.5s dans le cadre de nos expériences sur un ordinateur moyenne gamme – est suffisante pour nombre d'applications industrielles et pourrait être largement améliorée au prix de quelques optimisations. La technique d'apprentissage par forêt de décision utilisée se révèle de plus relativement simple à mettre en place, ne nécessitant que la donnée d'un modèle 3D de l'objet, et s'avère suffisamment robuste pour être employée sur des données bruitées sans apprentissage préalable de ce bruit.

Un soin tout particulier a été apporté à la généralité de notre méthode. Nous développons pour ce faire un cadre théorique rigoureux formalisant

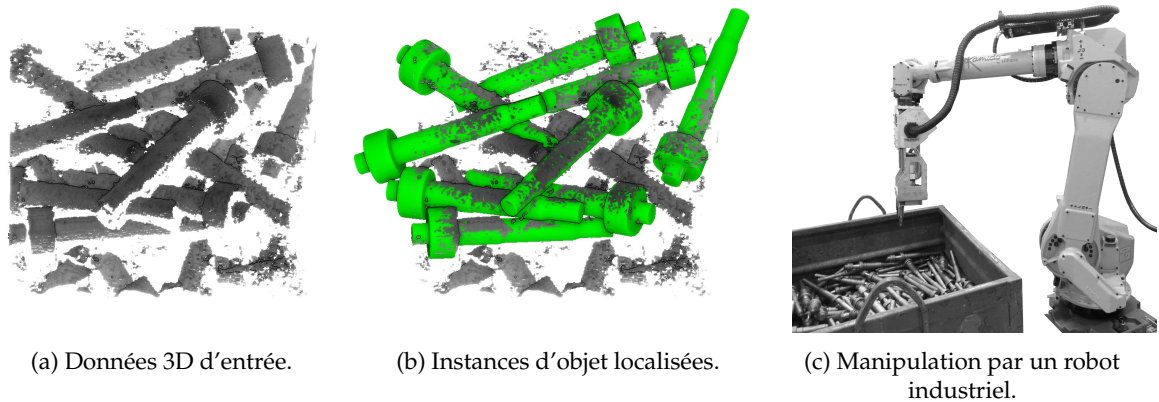


FIGURE 6.1 – Exemple d'application de dévracage de pièces de fonderie mise en œuvre au moyen de notre méthode de détection et d'estimation de pose d'objets.

la notion de pose pour tout objet rigide, y compris les objets symétriques, usuellement laissés de côté dans la littérature. Cette notion est complétée par la définition d'une distance entre poses d'un même objet. Celle-ci possède une interprétation géométrique simple, tout en présentant des propriétés proches d'une distance euclidienne qui permettent de réaliser diverses opérations de manière efficaces.

Ces développements sont évalués sur différents jeux de données réels et synthétiques de scènes de vrac produits par nos soins, après avoir validé la pertinence de l'usage de données synthétiques et suivant un protocole résolvant diverses limitations de ceux mis en œuvre dans la littérature. La comparaison de notre approche avec l'état de l'art, sur nos jeux de données comme sur celui de référence LINEMOD permet de montrer la viabilité de notre approche.

## 6.2 Contributions

Plutôt que d'ouvrir de nouveaux champs de recherche, nos travaux cherchaient avant tout à répondre à une problématique vieille de plus de 30 ans, et nous espérons que ceux-ci pourront contribuer au progrès du domaine.

Notre formalisation de la notion de pose d'objet rigide et la distance qui y est associée fournissent ainsi un cadre utile afin de manipuler des poses d'objets rigides arbitraires, indépendamment de leur classe de symétrie et sans nécessiter de choix arbitraires de paramètres.

La méthode de détection et d'estimation de pose adaptée à la problématique du dévracage robotisé développée dans ce manuscrit ne fait pas l'objet d'une publication scientifique pour des raisons de propriété intellectuelle. Celle-ci est cependant d'ores et déjà mise en œuvre avec succès chez Siléane pour différentes applications de dévracage industriel comme l'illustre la figure 6.1.



Enfin, nous espérons que notre proposition d'une nouvelle méthodologie d'évaluation – associée à d'autres initiatives telles qu'ITODD (Drost et al., 2017) – aidera à dépasser certains travers des méthodologies de référence, dont les limitations et le manque de représentativité risquent d'orienter l'effort de recherche dans une mauvaise direction.

Le problème abordé dans cette thèse ne reste cependant qu'une minuscule fraction d'un problème plus vaste qu'est l'analyse et l'interprétation automatique de scène, où tout reste à faire.

# Annexes

## A Méthodes de calcul pour un maillage triangulaire

Le centre de masse, l'aire et la matrice de covariance de la surface d'un objet définie par un maillage triangulaire  $\mathcal{S} = \cup_i \mathcal{T}(a_i, b_i, c_i)$  – où  $\mathcal{T}(a, b, c)$  est un triangle défini par trois sommets  $a, b, c \in \mathbb{R}^3$  – peuvent être estimés simplement à partir des contributions des différents triangles.

Soit  $\mathcal{T}(a, b, c)$  un triangle donné. Son aire peut être estimée à partir du produit vectoriel de deux de ses arêtes :

$$S_{a,b,c} = \frac{\|(\mathbf{b} - \mathbf{a}) \times (\mathbf{c} - \mathbf{a})\|}{2}, \quad (1)$$

son centre de masse par :

$$\mathbf{o}_{a,b,c} = \frac{\mathbf{a} + \mathbf{b} + \mathbf{c}}{3}, \quad (2)$$

et sa matrice de covariance non centrée par :

$$\sigma_{a,b,c} = \frac{S_{a,b,c}}{12} \left( 9\mathbf{o}_{a,b,c}\mathbf{o}_{a,b,c}^\top + \mathbf{a}\mathbf{a}^\top + \mathbf{b}\mathbf{b}^\top + \mathbf{c}\mathbf{c}^\top \right). \quad (3)$$

De ses résultats, on déduit l'expression de l'aire totale du maillage :

$$S = \sum_i S_{a_i, b_i, c_i}, \quad (4)$$

son centre de masse :

$$\mathbf{o} = \sum_i S_{a_i, b_i, c_i} \mathbf{o}_{a_i, b_i, c_i}, \quad (5)$$

et sa matrice de covariance normalisée, pour peu que le centre de masse du maillage soit choisi pour origine du repère objet :

$$\Lambda^2 = \frac{1}{S} \sum_i \sigma_{a_i, b_i, c_i}. \quad (6)$$

## B Simplification de l'expression de la distance proposée dans le cas d'un objet de révolution sans invariance par rotoréflexion

Le terme d'orientation de la distance proposée s'exprime dans le cas d'un objet de révolution sans invariance par rotoréflexion

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = \min_{\phi_1, \phi_2} \|\mathbf{R}_2 \mathbf{R}_z^{\phi_2} \Lambda - \mathbf{R}_1 \mathbf{R}_z^{\phi_1} \Lambda\|_F^2. \quad (7)$$

La norme de Frobenius étant invariante aux rotations, l'égalité (7) peut être réécrite en introduisant la rotation relative  $\mathbf{R} \triangleq \mathbf{R}_1^{-1} \mathbf{R}_2$  :

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = \min_{\phi_1, \phi_2} \|\mathbf{R}_z^{-\phi_1} \mathbf{R} \mathbf{R}_z^{\phi_2} \Lambda - \Lambda\|_F^2. \quad (8)$$

On paramétrise alors  $\mathbf{R}$  au moyen d'angles d'Euler  $(\tilde{\psi}, \theta, \tilde{\phi}) \in \mathbb{R}^3$  de telle sorte que  $\mathbf{R} = \mathbf{R}_z^{\tilde{\psi}} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\tilde{\phi}}$ , en considérant les rotations élémentaires suivantes :

$$\forall \alpha \in \mathbb{R}, \mathbf{R}_z^\alpha \triangleq \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \mathbf{R}_x^\alpha \triangleq \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}. \quad (9)$$

En injectant ces paramètres dans l'expression précédente, le changement de variables  $\psi \leftarrow \tilde{\psi} - \phi_1$  et  $\phi \leftarrow \tilde{\phi} + \phi_2$  conduit à l'égalité

$$\begin{aligned} d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) &= \min_{\phi_1, \phi_2} \|\mathbf{R}_z^{-\phi_1} \mathbf{R}_z^{\tilde{\psi}} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\tilde{\phi}} \mathbf{R}_z^{\phi_2} \Lambda - \Lambda\|_F^2 \\ &= \min_{\psi, \phi} \|\mathbf{R}_z^{\psi} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\phi} \Lambda - \Lambda\|_F^2. \end{aligned} \quad (10)$$

La structure particulière de  $\Lambda$  (équation (3.33) page 74) permet de décomposer le terme à minimiser en deux parties :

$$\begin{aligned} \|\mathbf{R}_z^{\psi} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\phi} \Lambda - \Lambda\|_F^2 &= \lambda_z^2 \underbrace{\|\mathbf{R}_z^{\psi} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\phi} \mathbf{e}_z - \mathbf{e}_z\|^2}_{a_{\psi, \phi}} \\ &\quad + \lambda_r^2 \underbrace{(\|\mathbf{R}_z^{\psi} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\phi} \mathbf{e}_x - \mathbf{e}_x\|^2 + \|\mathbf{R}_z^{\psi} \mathbf{R}_x^{\theta} \mathbf{R}_z^{\phi} \mathbf{e}_y - \mathbf{e}_y\|^2)}_{b_{\psi, \phi}}. \end{aligned} \quad (11)$$

En développant ces termes au moyen des rotations élémentaires (9), on aboutit aux égalités suivantes :

$$\begin{cases} a_{\psi, \phi} = 2(1 - \cos(\theta)) \\ b_{\psi, \phi} = 4 - 2 \cos(\psi + \phi)(1 + \cos(\theta)). \end{cases} \quad (12)$$

Le premier terme est indépendant de  $\psi$  et  $\phi$ . Le second peut quant à lui être minimisé facilement relativement à ces deux paramètres, et admet un minimum qui se trouve être égal au premier terme :

$$\min_{\psi, \phi} b_{\psi, \phi} = 2(1 - \cos(\theta)). \quad (13)$$

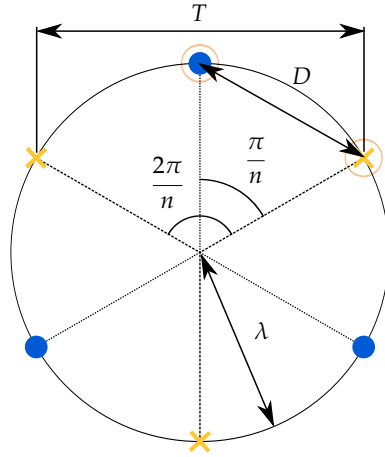


FIGURE 2 – Cas limite de cohérence dans le cas d'un objet 2D cyclique (ici d'ordre  $n = 3$ ).

On dispose ainsi d'une solution analytique à la distance entre deux poses. Cependant, le fait de passer par une décomposition sous forme d'angle d'Euler serait peu pratique et peu efficace, aussi nous préférons exploiter la propriété suivante

$$\begin{aligned} 2(1 - \cos(\theta)) &= \|\mathbf{R}e_z - e_z\|^2 \\ &= \|\mathbf{R}_2e_z - \mathbf{R}_1e_z\|^2 \end{aligned} \quad (14)$$

de manière à exprimer le terme d'orientation de notre distance en fonction de la distance entre les axes de révolution de l'objet dans les deux poses :

$$d_{\text{rot}}^2(\mathcal{P}_1, \mathcal{P}_2) = (\lambda_r^2 + \lambda_z^2) \|\mathbf{R}_2e_z - \mathbf{R}_1e_z\|^2. \quad (15)$$

## C Condition de cohérence pour un doublet de représentants de poses d'un objet 2D cyclique

La figure 2 illustre pour un objet 2D cyclique d'ordre  $n \in \mathbb{N}, n \geq 2$ , le cas limite de cohérence d'un doublet de représentants (entourés sur la figure) de deux poses. Ces poses sont représentées chacune par  $n$  points espacés de  $2\pi/n$  sur un cercle de rayon  $\lambda \in \mathbb{R}_+^*$  (respectivement en bleu et jaune sur la figure), en ne considérant que l'orientation de l'objet (le cas limite de cohérence étant obtenu pour des poses de positions identiques).

La distance minimale  $T$  entre deux représentants d'une même pose s'exprime d'après le théorème d'Al-Kashi (loi des cosinus)

$$T = \sqrt{2}\lambda \sqrt{1 - \cos\left(\frac{2\pi}{n}\right)} = 2\lambda \sin\left(\frac{\pi}{n}\right). \quad (16)$$

La distance maximale (strictement)  $D$  entre deux représentants de poses permettant de garantir leur cohérence mutuelle peut être exprimée de même

$$\begin{aligned} D &= 2\lambda \sin\left(\frac{\pi}{2n}\right) \\ &= \frac{\sin(\pi/(2n))}{\sin(\pi/n)} T. \end{aligned} \quad (17)$$

Cette distance vaut  $\sqrt{2}/2T \approx 0.71T$  dans le cas  $n = 2$ , mais tend vers  $T/2$  – la borne présentée dans la proposition 15 – lorsque  $n$  tend vers l’infini.

## D Détail de calcul : distance minimale entre représentants

Dans cette section, nous détaillons le calcul de la distance minimale  $T$  entre représentants d’une même pose (voir définition 4) pour les objets de notre exemple applicatif.

Le *lapin* et le *chandelier* n’admettent qu’un représentant par pose, aussi  $T = +\infty$  pour ceux-ci par convention.

Le cas de la *fusée* nécessite en revanche quelques calculs. Par simplicité, nous considérons un repère objet dont l’axe  $e_z$  correspond à l’axe de symétrie de la fusée. Dans ce repère, le groupe de symétrie propre de la fusée peut être exprimé

$$G = \{I, R_z^{2\pi/3}, R_z^{-2\pi/3}\} \quad (18)$$

et la racine carré de la matrice de covariance est de la forme

$$\Lambda = \text{diag}(\lambda_r, \lambda_r, \lambda_z). \quad (19)$$

Afin de simplifier d’avantage le calcul de  $T$ , nous considérons un représentant particulier  $p$  (spécifié ci-dessous) de la pose de référence  $\mathcal{P}_0$ . Les représentants  $\mathcal{R}(\mathcal{P}_0)$  de cette pose sont

$$\left\{ \underbrace{\begin{pmatrix} \text{vec}(\Lambda) \\ \mathbf{0}_3 \end{pmatrix}}_p, \begin{pmatrix} \text{vec}(R_z^{2\pi/3}\Lambda) \\ \mathbf{0}_3 \end{pmatrix}, \begin{pmatrix} \text{vec}(R_z^{-2\pi/3}\Lambda) \\ \mathbf{0}_3 \end{pmatrix} \right\} \subset \mathbb{R}^{12}. \quad (20)$$

Il est alors possible d’évaluer  $T$  :

$$\begin{aligned} T &= \min_{q \in \mathcal{R}(\mathcal{P}_0), q \neq p} \|q - p\| \\ &= \min \|R_z^{\pm 2\pi/3}\Lambda - \Lambda\|_F \\ &= \sqrt{6}\lambda_r. \end{aligned} \quad (21)$$

Le seuil  $\frac{T}{4}$  de la proposition 16 correspond donc dans le cas de la fusée à la valeur  $\frac{\sqrt{3}}{2}\lambda_r$ .

## E Robustesse au bruit

Évaluations des performances sur les scènes du jeu de données *bunny*, en fonction du niveau de bruit :

### Précision Moyenne (AP)

		Écart-type $\sigma$ du bruit (en % du diamètre de l'objet)										
		0%	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Probabilité $p_s$ de pixel sans information	0%	91.3%	89.1%	86.6%	84.0%	81.1%	78.0%	73.5%	64.3%	53.8%	45.3%	38.6%
	5%	90.5%	88.2%	85.6%	82.6%	79.4%	75.9%	69.8%	59.9%	49.5%	41.3%	33.7%
	10%	88.6%	86.7%	83.7%	80.4%	76.8%	72.6%	63.9%	53.3%	44.2%	37.1%	31.5%
	15%	85.8%	83.7%	80.7%	77.4%	72.5%	66.9%	56.6%	47.3%	39.1%	31.4%	25.8%
	20%	81.1%	78.8%	75.7%	71.5%	65.7%	58.2%	48.6%	39.4%	30.7%	23.3%	18.0%
	25%	73.3%	71.5%	68.2%	62.8%	56.3%	46.9%	37.6%	28.5%	21.6%	15.7%	10.5%
	30%	61.2%	59.0%	55.3%	49.3%	42.0%	33.6%	24.8%	16.7%	10.5%	8.9%	6.1%
	35%	41.9%	41.4%	37.8%	31.9%	24.9%	18.3%	12.1%	8.0%	5.7%	4.3%	3.8%
	40%	20.3%	20.7%	19.7%	16.5%	11.3%	9.0%	6.5%	4.4%	3.0%	2.6%	2.6%
	45%	8.1%	8.5%	7.2%	6.4%	4.5%	4.1%	2.9%	2.6%	2.4%	2.2%	2.1%
	50%	3.7%	4.0%	3.2%	3.0%	3.0%	2.6%	2.3%	2.1%	2.1%	1.9%	1.9%

### Précision Moyenne pour la détection d'au plus une instance d'objet (AP1)

		Écart-type $\sigma$ du bruit (en % du diamètre de l'objet)										
		0%	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Probabilité $p_s$ de pixel sans information	0%	100%	100%	100%	100%	100%	100%	100%	99.7%	98.2%	91.0%	77.0%
	5%	100%	100%	100%	100%	100%	100%	100%	99.6%	93.2%	84.7%	71.1%
	10%	100%	100%	100%	100%	100%	100%	100%	96.6%	91.9%	83.9%	74.1%
	15%	100%	100%	100%	100%	100%	100%	100%	95.5%	88.6%	77.4%	68.3%
	20%	100%	100%	100%	100%	100%	100%	96.9%	89.4%	78.9%	65.1%	59.3%
	25%	100%	100%	100%	100%	99.7%	98.5%	94.3%	80.2%	66.4%	58.8%	38.5%
	30%	100%	99.1%	100%	99.4%	98.1%	89.0%	79.9%	63.7%	45.6%	38.0%	26.7%
	35%	96.9%	95.2%	96.3%	92.9%	81.2%	74.5%	55.2%	36.7%	27.0%	14.6%	14.7%
	40%	77.9%	77.2%	74.8%	70.3%	52.6%	41.0%	31.2%	17.1%	11.8%	8.3%	8.5%
	45%	38.2%	39.2%	30.0%	26.6%	19.5%	12.5%	9.4%	8.9%	6.4%	4.6%	3.6%
	50%	14.9%	14.3%	11.9%	9.2%	8.0%	6.9%	6.2%	4.3%	4.0%	2.2%	2.6%

### Précision Moyenne pour la détection d'au plus 3 instances d'objet (AP3)

		Écart-type $\sigma$ du bruit (en % du diamètre de l'objet)										
		0%	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Probabilité $p_s$ de pixel sans information	0%	99.9%	99.9%	99.9%	99.9%	99.9%	99.9%	99.6%	92.6%	76.8%	65.1%	58.0%
	5%	99.9%	99.9%	99.9%	99.9%	99.9%	99.9%	98.7%	90.2%	72.7%	64.2%	52.2%
	10%	99.9%	99.9%	99.9%	99.9%	99.9%	99.7%	95.8%	83.6%	71.7%	60.5%	54.6%
	15%	99.9%	99.9%	99.9%	99.9%	99.7%	98.8%	92.2%	80.5%	69.3%	57.3%	48.8%
	20%	99.9%	99.9%	99.7%	99.7%	99.1%	96.7%	88.5%	74.8%	61.3%	48.1%	39.7%
	25%	99.8%	99.9%	99.8%	99.5%	98.1%	91.6%	77.6%	62.5%	51.2%	40.9%	26.6%
	30%	99.1%	98.7%	97.8%	95.2%	90.5%	74.5%	61.2%	43.7%	29.1%	23.9%	16.1%
	35%	89.2%	87.4%	85.6%	77.2%	61.4%	48.9%	34.5%	21.8%	15.1%	9.5%	8.3%
	40%	54.7%	55.7%	54.2%	44.6%	30.8%	24.0%	16.9%	10.6%	6.8%	4.6%	4.7%
	45%	22.3%	22.8%	18.8%	15.9%	10.7%	8.1%	5.6%	5.1%	4.0%	3.0%	2.7%
	50%	7.5%	8.7%	6.5%	5.6%	5.3%	4.7%	3.6%	2.9%	2.8%	2.3%	2.2%

# Bibliographie

- W. Abbeloos et T. Goedemé. Point pair feature based object detection for random bin picking. In *Computer and Robot Vision (CRV), 2016 13th Conference on*, pages 432–439. IEEE, 2016. 143
- A. Agrawal, Y. Sun, J. Barnwell, et R. Raskar. Vision-guided robot system for picking objects by casting shadows. *The International Journal of Robotics Research*, 29(2-3) :155–173, 2010. 26, 27, 35, 36
- A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. Rusu, et 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. doi:10.1109/ICCVW.2011.6130296. 36
- A. Aldoma, F. Tombari, R. B. Rusu, et M. Vincze. OUR-CVFH-oriented, unique and repeatable clustered viewpoint feature histogram for object recognition and 6dof pose estimation. In *Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium*, pages 113–122. Springer, 2012. 36
- A. Aldoma, T. Fäulhammer, et M. Vincze. Automation of “ground truth” annotation for multi-view RGB-D object instance recognition datasets. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, page 5016–5023. IEEE, 2014. 144, 145, 146
- A. Aldoma, F. Tombari, L. Di Stefano, et M. Vincze. A global hypothesis verification framework for 3D object recognition in clutter. *IEEE transactions on pattern analysis and machine intelligence*, 38(7) :1383–1396, 2016. 50
- J. Angeles. Is there a characteristic length of a rigid-body displacement? *Mechanism and Machine Theory*, 41(8) :884–896, 2006. 61
- H. Araújo, R. L. Carceroni, et C. M. Brown. A fully projective formulation to improve the accuracy of lowe’s Pose-Estimation algorithm. *Computer Vision and Image Understanding*, 70(2) :227–238, 1998. ISSN 1077-3142. doi:10.1006/cviu.1997.0632. 49
- U. Asif, M. Bennamoun, et F. Sohel. Model-free segmentation and grasp selection of unknown stacked objects. In *European Conference on Computer Vision*, pages 659–674. Springer, 2014. 13, 15

- V. Balntas, A. Doumanoglou, C. Sahin, J. Sock, R. Kouskouridas, et T. Kim. Pose guided RGBD feature learning for 3D object pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 3856–3864, 2017. 36, 39
- O. Barinova, V. Lempitsky, et P. Kholi. On detection of multiple object instances using Hough transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9) :1773–1784, Sept. 2012. ISSN 0162-8828. doi:10.1109/TPAMI.2012.79. 50, 105
- H. Bay, T. Tuytelaars, et L. Van Gool. Surf : Speeded up robust features. In *Computer Vision—ECCV 2006*, page 404–417. Springer, 2006. 43
- C. Belta et V. Kumar. An SVD-based projection method for interpolation on SE (3). *Robotics and Automation, IEEE Transactions on*, 18(3) :334–345, 2002. 91
- P. J. Besl et R. C. Jain. Three-dimensional object recognition. *ACM Comput. Surv.*, 17(1) :75–145, Mar. 1985. ISSN 0360-0300. doi:10.1145/4078.4081. 21, 22, 41
- P. J. Besl et N. D. McKay. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2) :239–256, 1992. ISSN 0162-8828. doi:10.1109/34.121791. 48, 96, 120
- T. Birdal et S. Ilic. Point pair features based object detection and pose estimation revisited. In *3D Vision (3DV), 2015 International Conference on*, page 527–535. IEEE, 2015. 47, 49
- Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Blender Institute, Amsterdam, 2017. URL <http://www.blender.org>. 150, 176
- R. C. Bolles et P. Horaud. 3DPO : A Three- Dimensional Part Orientation System. *The International Journal of Robotics Research*, 5(3) :3–26, Sept. 1986. ISSN 0278-3649. doi:10.1177/027836498600500301. 28, 40, 49
- U. Bonde, V. Badrinarayanan, et R. Cipolla. Robust instance recognition in presence of occlusion and clutter. In *European Conference on Computer Vision*, page 520–535. Springer, 2014. 39, 144, 145, 146
- K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, et D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 30
- E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, et C. Rother. Learning 6d object pose estimation using 3d object coordinates. In *ECCV (2)*, pages 536–551, 2014. 30, 44, 52, 130, 144, 179, 181, 182
- E. Brachmann, F. Michel, A. Krull, M. Y. Yang, S. Gumhold, et C. Rother. Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 44



- R. Brégier, F. Devernay, L. Leyrit, et J. L. Crowley. Symmetry aware evaluation of 3D object detection and pose estimation in scenes of many parts in bulk. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017a. 18, 142, 172
- R. Brégier, F. Devernay, L. Leyrit, et J. L. Crowley. Defining the pose of any 3d rigid object and an associated distance. *International Journal of Computer Vision*, Nov 2017b. ISSN 1573-1405. doi:10.1007/s11263-017-1052-4. 18, 56
- E. Brown, N. Rodenberg, J. Amend, A. Mozeika, E. Steltz, M. R. Zakin, H. Lipson, et H. M. Jaeger. Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences*, 107 (44) :18809–18814, 2010. 11
- D. Buchholz, S. Winkelbach, et F. M. Wahl. RANSAM for industrial Bin-Picking. In *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pages 1–6, June 2010. 44, 45, 143
- Z. Cao, Y. Sheikh, et N. K. Banerjee. Real-time scalable 6DOF pose estimation for textureless objects. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, page 2441–2448. IEEE, 2016. 34, 39
- R. T. Chin et C. R. Dyer. Model-based recognition in robot vision. *ACM Computing Surveys (CSUR)*, 18(1) :67–108, 1986. 41
- G. S. Chirikjian. Partial Bi-Invariance of SE (3) metrics. *Journal of Computing and Information Science in Engineering*, 15(1) :011008, 2015. 61
- G. S. Chirikjian et S. Zhou. Metrics on motion and deformation of solid models. *Journal of Mechanical Design*, 120(2) :252–261, 1998. 63, 64
- C. Choi et H. I. Christensen. 3D pose estimation of daily objects using an RGB-D camera. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, page 3342–3349. IEEE, 2012. 47
- C. Choi, Y. Taguchi, O. Tuzel, M. Liu, et S. Ramalingam. Voting-based pose estimation for robotic assembly using a 3D sensor. In *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1724–1731, 2012. doi:10.1109/ICRA.2012.6225371. 47
- M. S. Costa et L. G. Shapiro. 3D object recognition and pose with relational indexing. *Computer Vision and Image Understanding*, 79(3) :364–407, 2000. 41
- E. Coumans. *Bullet Physics SDK : real-time collision detection and multi-physics simulation for VR, games, visual effects, robotics, machine learning etc.*, 2017. URL <http://www.bulletphysics.org>. 128, 150
- N. Craswell. Precision at n. In L. Liu et M. T. Özsu, editors, *Encyclopedia of Database Systems*, pages 2127–2128. Springer US, Boston, MA, 2009. ISBN 978-0-387-39940-9. doi:10.1007/978-0-387-39940-9\_484. 157
- A. Criminisi et J. Shotton. *Decision forests for computer vision and medical image analysis*. Springer Science & Business Media, 2013. 112

- A. Crivellaro, M. Rad, Y. Verdie, K. Moo Yi, P. Fua, et V. Lepetit. A novel representation of parts for accurate 3D object detection and tracking in monocular images. In *Proceedings of the IEEE International Conference on Computer Vision*, page 4391–4399, 2015. 39
- I. A. Şucan et S. Chitta. *MoveIt!*, 2017. URL <http://moveit.ros.org>. 12
- W. Curtis, A. Janin, et K. Zikan. A note on averaging rotations. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, pages 377–385, Sept. 1993. doi:10.1109/VRAIS.1993.380755. 83, 96
- D. Damen, P. Bunnun, A. Calway, et W. W. Mayol-Cuevas. Real-time learning and detection of 3D texture-less objects : A scalable approach. In *BMVC*, pages 1–12, 2012. 41
- R. Detry, E. Baseski, M. Popovic, Y. Touati, N. Kruger, O. Kroemer, J. Peters, et J. Piater. Learning object-specific grasp affordance densities. In *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on*, page 1–7. IEEE, 2009. 15
- R. Di Gregorio. A novel point of view to define the distance between two rigid-body poses. In *Advances in Robot Kinematics : Analysis and Design*, page 361–369. Springer, 2008. 63
- R. Diankov. *Automated construction of robotic manipulation programs*. PhD thesis, Carnegie Mellon University, 2010. 12
- Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, et T. Hirai. Fast graspability evaluation on single depth maps for bin picking with general grippers. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1997–2004, 2014. doi:10.1109/ICRA.2014.6907124. 13
- A. Doumanoglou, R. Kouskouridas, S. Malassiotis, et T.-K. Kim. Recovering 6D object pose and predicting next-best-view in the crowd. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 26, 47, 48, 52, 57, 155, 158, 160, 161, 162, 166, 173
- B. Drost et S. Ilic. 3D object detection and localization using multimodal point pair features. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, page 9–16. IEEE, 2012. 47, 49
- B. Drost, M. Ulrich, N. Navab, et S. Ilic. Model globally, match locally : Efficient and robust 3D object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, page 998–1005. IEEE, 2010. 28, 29, 45, 48, 51, 52, 92, 93, 94, 145, 147, 169, 174, 175, 179, 181
- B. Drost, M. Ulrich, P. Bergmann, P. Härtinger, et C. Steger. Introducing MVTec ITODD-A dataset for 3D object recognition in industry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 2200–2208, 2017. 142, 143, 198
- R. O. Duda et P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1) :11–15, 1972. 20, 45

- J. K. Eberhardter et B. Ravani. Local metrics for rigid body displacements. *Journal of Mechanical Design*, 126(5) :805–812, Oct. 2004. ISSN 1050-0472. doi:[10.1115/1.1767816](https://doi.org/10.1115/1.1767816). 64
- K. R. Etzel et J. M. McCarthy. A metric for spatial displacement using biquaternions on SO(4). In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 4, page 3185–3190. IEEE, 1996. 61
- M. Everingham, L. Van Gool, C. K. Williams, J. Winn, et A. Zisserman. The Pascal visual object classes (VOC) challenge. *International journal of computer vision*, 88(2) :303–338, 2010. 159
- G. Fanelli, J. Gall, et L. Van Gool. Real time head pose estimation with random regression forests. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 617–624, June 2011. doi:[10.1109/CVPR.2011.5995458](https://doi.org/10.1109/CVPR.2011.5995458). 57, 92, 134
- M. A. Fischler et R. C. Bolles. Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6) :381–395, 1981. ISSN 0001-0782. doi:[10.1145/358669.358692](https://doi.org/10.1145/358669.358692). 44
- A. W. Fitzgibbon. Robust registration of 2D and 3D point sets. *Image and Vision Computing*, 21(13) :1145–1153, 2003. 49
- S. Fuchs, S. Haddadin, M. Keller, S. Parusel, A. Kolb, et M. Suppa. Cooperative bin-picking with time-of-flight camera and impedance controlled dlr lightweight robot iii. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, page 4862–4867. IEEE, 2010. 11
- K. Fukunaga et L. D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *Information Theory, IEEE Transactions on*, 21(1) :32–40, 1975. 95
- J. Gall, A. Yao, N. Razavi, L. Van Gool, et V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11) :2188–2202, Nov. 2011. ISSN 0162-8828. doi:[10.1109/TPAMI.2011.70](https://doi.org/10.1109/TPAMI.2011.70). 47, 111
- S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, et M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6) :2280–2292, 2014. 148
- R. Girshick, J. Shotton, P. Kohli, A. Criminisi, et A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, page 415–422. IEEE, 2011. 47
- R. Girshick, J. Donahue, T. Darrell, et J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 54

- C. Gramkow. On averaging rotations. *Journal of Mathematical Imaging and Vision*, 15(1-2) :7–16, 2001. 82, 84, 85
- S. Granger et X. Pennec. Multi-scale EM-ICP : a fast and robust approach for surface registration. In *Computer Vision—ECCV 2002*, page 418–432. Springer, 2002. 48
- M. Grard, R. Brégier, F. Sella, E. Dellandréa, et L. Chen. Object segmentation in depth maps with one user click and a synthetically trained fully convolutional network. In *Springer Proceedings in Advanced Robotics*, à paraître. URL <https://arxiv.org/abs/1801.01281>. 14, 15, 18
- M. Gschwandtner, R. Kwitt, A. Uhl, et W. Pree. BlenSor : Blender sensor simulation toolbox. In *Advances in Visual Computing*, page 199–208. Springer, 2011. 151
- K. C. Gupta. Measures of positional error for a rigid body. *Journal of mechanical design*, 119(3) :346–348, 1997. 62
- S. Gupta, P. Arbeláez, R. Girshick, et J. Malik. Aligning 3D models to RGB-D images of cluttered scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4731–4740, 2015. 35, 37
- A. Handa, T. Whelan, J. McDonald, et A. J. Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *Robotics and automation (ICRA), 2014 IEEE international conference on*, page 1524–1531. IEEE, 2014. 151
- C. Harris et M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, pages 10–5244. Citeseer, 1988. 43
- R. Hartley et A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 24
- K. He, G. Gkioxari, P. Dollár, et R. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 14, 15
- M. Heber, M. Rüther, et H. Bischof. Catadioptric multiview pose estimation for robotic pick and place. In *VISAPP (1)*, pages 423–426, 2010. 26, 35
- S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, et V. Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 858–865. IEEE, 2011. 28, 127
- S. Hinterstoisser, C. Cagniart, S. Ilic, P. Sturm, N. Navab, P. Fua, et V. Lepetit. Gradient Response Maps for Real-Time Detection of Textureless Objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5) :876–888, 2012a. ISSN 0162-8828. doi:10.1109/TPAMI.2011.206. 33, 39, 127
- S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, et N. Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012b. 28, 30, 33, 38, 49, 52, 53, 57, 63, 132, 144, 145, 146, 147, 155, 169, 172, 174, 178, 179, 182

- S. Hinterstoisser, V. Lepetit, N. Rajkumar, et K. Konolige. Going further with point pair features. In *European Conference on Computer Vision*, page 834–848. Springer, 2016. 47, 50, 53, 179, 181, 182
- H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2) :328–341, 2008. 148
- T. Hodaň, X. Zabulis, M. Lourakis, Š. Obdržálek, et J. Matas. Detection and fine 3D pose estimation of texture-less objects in RGB-D images. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 4421–4428. IEEE, 2015. 34, 49
- T. Hodaň, J. Matas, et Š. Obdržálek. On evaluation of 6D object pose estimation. In *European Conference on Computer Vision Workshops (ECCVW) 2016*, 2016. 144, 146
- T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, et X. Zabulis. T-LESS : An RGB-D dataset for 6D pose estimation of texture-less objects. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 880–888. IEEE, 2017. 28, 143, 144, 145, 146
- D. Holz, M. Nieuwenhuisen, D. Droschel, J. Stückler, A. Berner, J. Li, R. Klein, et S. Behnke. Active recognition and manipulation for mobile robot bin picking. In *Gearing Up and Accelerating Cross-fertilization between Academic and Industrial Robotics Research in Europe*, page 133–153. Springer, 2014. 41
- S. Holzer, J. Shotton, et P. Kohli. Learning to efficiently detect repeatable interest points in depth data. In *European Conference on Computer Vision*, pages 200–213. Springer, 2012. 43
- B. K. Horn et K. Ikeuchi. Picking parts out of a bin. Technical report, Massachusetts Inst. of Tech. Cambridge Artificial Intelligence Lab, 1983. 10, 36, 143
- P. V. Hough. Machine analysis of bubble chamber pictures. In *International conference on high energy accelerators and instrumentation*, volume 73, 1959. 20, 45
- P. V. C. Hough. Method and means for recognizing complex patterns, Dec. 1962. US Patent 3,069,654. 45
- K. Hsiao, S. Chitta, M. Ciocarlie, et E. G. Jones. Contact-reactive grasping of objects with partial shape information. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, page 1228–1235. IEEE, 2010. 11
- K. Ikeuchi, H. K. Nishihara, B. K. P. Horn, P. Sobalvarro, et S. Nagata. Determining grasp configurations using photometric stereo and the PRISM binocular stereo system. *The International Journal of Robotics Research*, 5(1) :46–65, Mar. 1986. ISSN 0278-3649, 1741-3176. doi:10.1177/027836498600500103. 14

- E. Jang, S. Vijayanarasimhan, P. Pastor, J. Ibarz, et S. Levine. End-to-End learning of semantic grasping. In *Conference on Robot Learning*, July 2017. arXiv : 1707.01932. 15
- Y. Jiang, S. Moseson, et A. Saxena. Efficient grasping from RGBD images : Learning using a new rectangle representation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, page 3304–3311. IEEE, 2011. 13
- A. Johnson et M. Hebert. Recognizing objects by matching oriented points. In *, 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997. Proceedings*, pages 684–689, 1997. doi:10.1109/CVPR.1997.609400. 43, 44
- A. E. Johnson et M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on pattern analysis and machine intelligence*, 21(5) :433–449, 1999. 28, 29, 43, 147
- T. Jost et H. Hügli. A multi-resolution ICP with heuristic closest point search for fast and robust 3D registration of range images. In *3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. Fourth International Conference on*, page 427–433. IEEE, 2003. 48, 123
- J. T. Kajiya. The rendering equation. In *ACM Siggraph Computer Graphics*, volume 20, pages 143–150. ACM, 1986. 151
- A. C. Kak et L. Edwards. Experimental state of the art in 3D object recognition and localization using range data. In *Proceedings of the workshop on vision for robots in IROS1995*, page 45–54, 1995. 41
- K. Kazerounian et J. Rastegar. Object norms : A class of coordinate and metric independent norms for displacements. *Flexible Mechanisms, Dynamics, and Analysis. ASME DE-Vol, 47* :271–275, 1992. 63, 64
- W. Kehl, F. Tombari, N. Navab, S. Ilic, et V. Lepetit. Hashmod : A hashing method for scalable 3D object detection. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 36.1–36.12. BMVA Press, September 2015. ISBN 1-901725-53-7. doi:10.5244/C.29.36. 34
- W. Kehl, F. Milletari, F. Tombari, S. Ilic, et N. Navab. Deep learning of local RGB-D patches for 3D object detection and 6D pose estimation. In *European Conference on Computer Vision*, pages 205–220. Springer, 2016. 47, 48, 179
- W. Kehl, F. Manhardt, F. Tombari, S. Ilic, et N. Navab. SSD-6D : making RGB-Based 3D detection and 6D pose estimation great again. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 1521–1529, 2017. 24, 28, 29, 30, 37, 39, 49, 52
- A. Kendall, M. Grimes, et R. Cipolla. PoseNet : A convolutional network for real-time 6-DOF camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2938–2946, 2015. 38, 56, 62

- E. Kim et G. Medioni. 3D object recognition in range images using visibility context. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, page 3800–3807. IEEE, 2011. 47
- L. F. Kozachenko et N. N. Leonenko. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, 23(2) :9–16, 1987. 135
- K. Kronander et A. Billard. Learning compliant manipulation through kinesthetic and tactile human-robot interaction. *IEEE Transactions on Haptics*, 7(3) :367–380, 2014. 12
- A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, et C. Rother. Learning Analysis-by-Synthesis for 6D pose estimation in RGB-D images. In *Proceedings of the IEEE International Conference on Computer Vision*, page 954–962, 2015. 44, 49, 166, 182
- P. M. Larochelle, A. P. Murray, et J. Angeles. A distance metric for finite sets of rigid-body displacements via the polar decomposition. *Journal of Mechanical Design*, 129(8) :883–886, 2007. 61
- I. Lenz, H. Lee, et A. Saxena. Deep learning for detecting robotic grasps. *CoRR*, 2013. 13, 14
- B. León, S. Ulbrich, R. Diankov, G. Puche, M. Przybylski, A. Morales, T. Asfour, S. Moio, J. Bohg, J. Kuffner, et al. Opengrasp : a toolkit for robot grasping simulation. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, page 109–120. Springer, 2010. 15
- V. Lepetit, P. Lagger, et P. Fua. Randomized trees for real-time keypoint recognition. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 775–781. IEEE, 2005. 43
- V. Lepetit, F. Moreno-Noguer, et P. Fua. EPnP : An Accurate O(n) Solution to the PnP Problem. *Int J Comput Vis*, 81(2) :155, Feb. 2009. ISSN 0920-5691, 1573-1405. doi:10.1007/s11263-008-0152-6. 44
- S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, et D. Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, page 0278364917710318, 2016. 13
- Q. Lin et J. W. Burdick. Objective and frame-invariant kinematic metric functions for rigid bodies. *The International Journal of Robotics Research*, 19(6) :612–625, 2000. 61, 63, 91
- T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, et C. L. Zitnick. Microsoft COCO : Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 28, 29
- M. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, et R. Chellappa. Fast object localization and pose estimation in heavy clutter for robotic bin picking. *The International Journal of Robotics Research*, 31(8) :951–973, 2012. 32, 33, 143

- W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, et A. C. Berg. SSD : Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 37, 38, 54
- D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial intelligence*, 31(3) :355–395, 1987. 49
- D. G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, page 1150–1157. Ieee, 1999. 43
- I. Lysenkov et V. Rabaud. Pose estimation of rigid transparent objects in transparent clutter. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, page 162–169. IEEE, 2013. 27
- J. Mahler et K. Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In *PMLR*, pages 515–524, Oct. 2017. 13
- J. M. R. Martinez et J. Duffy. On the metrics of rigid body displacements for infinite and finite bodies. *Journal of Mechanical Design*, 117(1) :41–47, 1995. 63
- R. V. Martinez, J. L. Branch, C. R. Fish, L. Jin, R. F. Shepherd, R. Nunes, Z. Suo, et G. M. Whitesides. Robotic tentacles with Three-Dimensional mobility based on flexible elastomers. *Advanced Materials*, 25(2) :205–212, 2013. 11
- F. Massa, B. C. Russell, et M. Aubry. Deep Exemplar 2D-3D Detection by Adapting From Real to Rendered Views. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6024–6033, 2016. 30
- R. A. Matthews. Tumbling toast, murphy’s law and the fundamental constants. *European Journal of Physics*, 16(4) :172, 1995. 131
- R. A. Matthews. Testing murphy’s law : urban myths as a source of school science projects. *School science review*, 82 :23–32, 2001. 131
- A. S. Mian, M. Bennamoun, et R. Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(10) :1584–1601, 2006. 44, 47, 144, 145
- A. T. Miller et P. K. Allen. Graspit! a versatile simulator for robotic grasping. *Robotics & Automation Magazine, IEEE*, 11(4) :110–122, 2004. 15
- O. Morel. *Environnement actif pour la reconstruction tridimensionnelle de surfaces métalliques spéculaires par imagerie polarimétrique*. PhD thesis, Université de Bourgogne, 2005. 27
- M. Muja et D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP (1)*, page 331–340, 2009. 36, 71



- J. L. Mundy, C. Huang, J. Liu, W. Hoffman, D. A. Forsyth, C. A. Rothwell, A. Zisserman, S. Utcke, et O. Bournez. MORSE : a 3D object recognition system based on geometric invariants. In *Proc. DARPA Image Understanding Workshop*, page 1393–1402, 1994. 40
- M. Nieuwenhuisen, D. Droschel, D. Holz, J. Stuckler, A. Berner, J. Li, R. Klein, et S. Behnke. Mobile bin picking with an anthropomorphic service robot. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, page 2327–2334. IEEE, 2013. 41
- D. Novotny, D. Larlus, et A. Vedaldi. Learning 3D object categories by looking around them. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 28, 38
- S. Nowozin. Improved information gain estimates for decision tree induction. In *Proceedings of the 29th International Conference on International Conference on Machine Learning*, pages 571–578. Omnipress, 2012. 135
- M. Oberweger, P. Wohlhart, et V. Lepetit. Training a feedback loop for hand pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3316–3324, 2015. 30, 38, 49
- M. Oshima et Y. Shirai. Object recognition using Three-Dimensional information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5 (4) :353–361, July 1983. ISSN 0162-8828. doi:10.1109/TPAMI.1983.4767405. 41
- M. Ozuysal, M. Calonder, V. Lepetit, et P. Fua. Fast keypoint recognition using random ferns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(3) :448–461, 2010. 43, 112
- F. C. Park. Distance metrics on the rigid-body motions with applications to mechanism design. *Journal of Mechanical Design*, 117(1) :48–54, 1995. 61, 62
- I. K. Park, M. Germann, M. D. Breitenstein, et H. Pfister. Fast and automatic object pose estimation for range images on the GPU. *Machine Vision and Applications*, 21(5) :749–766, Aug. 2010. ISSN 0932-8092, 1432-1769. doi:10.1007/s00138-009-0209-8. 28, 32, 49, 131
- B. Pelletier. Kernel density estimation on riemannian manifolds. *Statistics & Probability Letters*, 73(3) :297–304, July 2005. ISSN 01677152. doi:10.1016/j.spl.2005.04.004. 96
- X. Pennec. Computing the Mean of Geometric Features Application to the Mean Rotation. report, INRIA, Mar. 1998. 91
- L. Pinto et A. Gupta. Supersizing self-supervision : Learning to grasp from 50K tries and 700 robot hours. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3406–3413, May 2016. doi:10.1109/ICRA.2016.7487517. 13
- A. Purwar et Q. J. Ge. Reconciling distance metric methods for rigid body displacements. In *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, page 1295–1304. American Society of Mechanical Engineers, 2009. 61

- J. R. Quinlan. Induction of decision trees. *Machine learning*, 1(1) :81–106, 1986. [134](#)
- M. Rad et V. Lepetit. BB8 : A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. [24](#), [29](#), [30](#), [35](#), [39](#), [49](#), [50](#), [52](#)
- M. Rad, P. M. Roth, et V. Lepetit. ALCN : Adaptive Local Contrast Normalization for Robust Object Detection and 3D Pose Estimation. In *Proc. British Machine Vision Conf. (BMVC)*, 2017. [30](#)
- K. Rahardja et A. Kosaka. Vision-based bin-picking : Recognition and localization of multiple complex objects using simple visual cues. In *Intelligent Robots and Systems' 96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 3, page 1448–1457. IEEE, 1996. [40](#)
- J. Redmon et A. Angelova. Real-time grasp detection using convolutional neural networks. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, page 1316–1322. IEEE, 2015. [13](#)
- J. Redmon, S. Divvala, R. Girshick, et A. Farhadi. You Only Look Once : Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. [54](#)
- S. Ren, K. He, R. Girshick, et J. Sun. Faster R-CNN : Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. [54](#)
- R. Rios-Cabrera et T. Tuytelaars. Discriminatively trained templates for 3D object detection : A real time scalable approach. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, page 2048–2055. IEEE, 2013. [34](#), [179](#)
- J. J. Rodrigues, J. Kim, M. Furukawa, J. Xavier, P. Aguiar, et T. Kanade. 6D pose estimation of textureless shiny objects using random ferns for bin-picking. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, page 3334–3341. IEEE, 2012. [47](#), [48](#), [49](#), [92](#), [143](#)
- E. Rublee, V. Rabaud, K. Konolige, et G. Bradski. ORB : An efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE international conference on*, pages 2564–2571. IEEE, 2011. [42](#), [43](#)
- R. B. Rusu et S. Cousins. 3D is here : Point cloud library (PCL). In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1–4. IEEE, 2011. [172](#)
- R. B. Rusu, N. Blodow, et M. Beetz. Fast point feature histograms (FPFH) for 3D registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, page 3212–3217. IEEE, 2009. [43](#)
- R. B. Rusu, G. Bradski, R. Thibaux, et J. Hsu. Fast 3D recognition and pose using the viewpoint feature histogram. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2155–2162. IEEE, 2010. [35](#), [36](#)

- A. Schmitz, U. Pattacini, F. Nori, L. Natale, G. Metta, et G. Sandini. Design, realization and sensorization of the dexterous iCub hand. In *Humanoid Robots (Humanoids), 2010 10th IEEE-RAS International Conference on*, pages 186–191. IEEE, 2010. 11
- P. H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1) :1–10, 1966. 81
- P. Sermanet, K. Xu, et S. Levine. Unsupervised perceptual rewards for imitation learning. In *Proceedings of Robotics : Science and Systems*, 2017. arXiv : 1612.06699. 15
- I. Sharf, A. Wolf, et M. Rubin. Arithmetic and geometric solutions for average rigid-body rotation. *Mechanism and Machine Theory*, 45(9) :1239–1251, Sept. 2010. ISSN 0094114X. doi:10.1016/j.mechmachtheory.2010.05.002. 82
- J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, et al. Efficient human pose estimation from single depth images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12) :2821–2840, 2013a. 29, 54, 139
- J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, et R. Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1) :116–124, 2013b. 112, 114
- H. Singh, N. Misra, V. Hnizdo, A. Fedorowicz, et E. Demchuk. Nearest neighbor estimates of entropy. *American journal of mathematical and management sciences*, 23(3-4) :301–321, 2003. 135
- S. Song et J. Xiao. Sliding shapes for 3D object detection in depth images. In D. Fleet, T. Pajdla, B. Schiele, et T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, number 8694 in Lecture Notes in Computer Science, pages 634–651. Springer International Publishing, Sept. 2014. ISBN 978-3-319-10598-7 978-3-319-10599-4. doi:10.1007/978-3-319-10599-4\_41. 34
- R. Suárez, J. Cornella, et M. R. Garzón. *Grasp quality measures*. Institut d’Organització i Control de Sistemes Industrials, 2006. 15
- R. Subbarao et P. Meer. Nonlinear mean shift for clustering over analytic manifolds. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, page 1168–1175. IEEE, 2006. 92
- I. Sucan, M. Moll, et L. Kavraki. The open motion planning library. *IEEE Robotics Automation Magazine*, 19(4) :72–82, 2012. ISSN 1070-9932. doi:10.1109/MRA.2012.2205651. 56
- A. Tejani, D. Tang, R. Kouskouridas, et T.-K. Kim. Latent-class hough forests for 3d object detection and pose estimation. *ECCV (6)*, 8694 :462–477, 2014. 30, 38, 40, 47, 48, 57, 92, 105, 134, 144, 145, 146, 147, 158
- H. Tjaden, U. Schwanecke, et E. Schömer. Real-Time monocular segmentation and pose tracking of multiple objects. In *European Conference on Computer Vision*, page 423–438. Springer, 2016. 56

- F. Tombari, S. Salti, et L. Di Stefano. Unique signatures of histograms for local surface description. In *European conference on computer vision*, page 356–369. Springer, 2010. 43
- O. Tuzel, R. Subbarao, et P. Meer. Simultaneous multiple 3D motion estimation via mode finding on Lie groups. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, page 18–25. IEEE, 2005. 92
- M. Ulrich, C. Wiedemann, et C. Steger. Combining Scale-Space and Similarity-Based aspect graphs for fast 3D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10) :1902–1914, Oct. 2012. ISSN 0162-8828. doi:10.1109/TPAMI.2011.266. 32, 33
- S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 13(4) :376–380, 1991. 44, 81
- N. Vahrenkamp, M. Kröhnert, S. Ulbrich, T. Asfour, G. Metta, R. Dillmann, et G. Sandini. Simox : A robotics toolbox for simulation, motion and grasp planning. In *Intelligent Autonomous Systems 12*, page 585–594. Springer, 2013. 15
- B. K. Vainsthein. *Fundamentals of Crystals*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1994. ISBN 978-3-642-08153-8 978-3-662-02975-6. 59
- A. Van den Bosch, T. Bogers, et M. De Kunder. Estimating search engine index size variability : a 9-year longitudinal study. *Scientometrics*, 107(2) : 839–856, May 2016. ISSN 0138-9130, 1588-2861. doi:10.1007/s11192-016-1863-z. 154
- P. Viola et M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, page I–511. IEEE, 2001. 20
- W. Wang, L. Chen, D. Chen, S. Li, et K. Kuhlntenz. Fast object recognition and 6D pose estimation using viewpoint oriented color-shape histogram. In *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, page 1–6. IEEE, 2013. 35, 36
- J. Weisz et P. K. Allen. Pose error robust grasping from contact wrench space metrics. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, page 557–562. IEEE, 2012. 15
- P. Wohlhart et V. Lepetit. Learning descriptors for object recognition and 3D pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 3109–3118, 2015. 36, 37, 52
- J. Yang, H. Li, et Y. Jia. Go-ICP : Solving 3D registration efficiently and globally optimally. In *Proceedings of the IEEE International Conference on Computer Vision*, page 1457–1464, 2013. 49

- M. Zefran et V. Kumar. Planning of smooth motions on SE (3). In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, page 121–126. IEEE, 1996. 63, 91
- E. Zhang et Y. Zhang. F-measure. In L. Liu et M. T. Özsu, editors, *Encyclopedia of Database Systems*, pages 1147–1147. Springer US, Boston, MA, 2009. ISBN 978-0-387-39940-9. doi:[10.1007/978-0-387-39940-9\\_483](https://doi.org/10.1007/978-0-387-39940-9_483). 157
- Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11) :1330–1334, 2000. 24