Visuo-Haptic recognition of daily-life objects : a contribution to the data scarcity problem

Zineb Abderrahmane

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En SYAM – Systèmes Automatiques et Microélectroniques

École doctorale I2S – Information, Structure, Systèmes

Unité de recherche UMR 5506

Visuo-Haptic Recognition of Daily-Life Objects: a contribution to the Data Scarcity Problem

Présentée par Zineb ABDERRAHMANE

Le 29 Novembre 2018

Sous la direction de Andrea CHERUBINI

Devant le jury composé de

Véronique PERDEREAU, Professeur, ISIR, Paris, France	Présidente de Jury
Yoichi MIYAWAKI, Professeur, UEC, Tokyo, Japan	Rapporteur
Lorenzo NATALE, Chercheur, IIT, Gênes, Italie	Rapporteur
Gordon CHENG, Professeur, TUM, Munich, Allemagne	Examinateur
André CROSNIER, Professeur, LIRMM, Montpellier, France	Co-encadrant
Gowrishankar GANESH, CR CNRS HDR, LIRMM, Montpellier, France	Co-encadrant
Andrea CHERUBINI, MCF HDR, LIRMM, Montpellier, France	Directeur de thèse

UNIVERSITÉ DE MONTPELLIER

Acknowledgment

This thesis was fully funded by the Algerian ministry of Higher Education and Scientific Research. A big thank you to all the people in charge of this scholarship in the Algerian ministry of Higher Education and Scientific Research, Ecole national Superieur d'Informatique (ESI) and the Algerian consulate in Montpellier.
I had the chance to work under the supervision of Pr. André Crosnier, Dr. Andrea Cherubini and Dr. Gowrishankar Ganesh. I would like to thank them all for their help, advises, critics and especially for their great flexibility with me. It had been an honor to work with them.
Many thanks to my family, especially my mother Naima Teldjoune, my father Moussa, my sisters Fairouz, Khadidja and Meriem, my brothers Mohamed Ali and Omar, my nephew Abdelhafed and my niece Nada for their great support and for believing in me.
I would like to thank Funny group girls, especially my cousin Nina, for sharing this journey with me. Our trips, shared meals, tiny parties made this journey unforgettable.
A special thanks to my fiance Tarek for always bringing the smile to my face.
Finally, thank you to every friend, family member and every colleague who contributed to the success of this thesis.

Dedication

To my grandfather Moussa Teldjoune.

Résumé

Reconnaissance visio-haptique des objets de la vie quotidienne à partir de peu de données d'entraînement

Il est important pour les robots de pouvoir reconnaître les objets rencontrés dans la vie quotidienne afin d'assurer leur autonomie. De nos jours, les robots sont équipés de capteurs sophistiqués permettant d'imiter le sens humain du toucher. C'est ce que permet aux robots interagissant avec les objets de percevoir les propriétés (telles que la texture, la rigidité et la matière) nécessaires pour leur reconnaissance. Le but de cette thèse est d'exploiter les données haptiques issues de l'interaction robot-objet afin de reconnaître les objets de la vie quotidienne, et cela en utilisant les algorithmes d'apprentissage automatique. Le problème qui se pose est la difficulté de collecter suffisamment de données haptiques afin d'entraîner les algorithmes d'apprentissage supervisé sur tous les objets que le robot doit reconnaître. En effet, les objets de la vie quotidienne sont nombreux et l'interaction physique entre le robot et chaque objet pour la collection des données prend beaucoup de temps et d'efforts. Pour traiter ce problème, un système de reconnaissance haptique permettant de reconnaître des objets à partir d'aucune, d'une seule, ou de plusieurs données d'entraînement est proposé . Ensuite, la vision est integrée afin d'améliorer la reconnaissance d'objets lorsque le robot est équipé de caméras.

Mots-clés

- Classification
- Reconnaissance haptique
- Reconnaissance visuelle
- Zero-Shot Learning

Abstract

Visuo-Haptic recognition of daily-life objects: a contribution to the data scarcity problem

Recognizing surrounding objects is an important skill for the autonomy of robots performing in daily-life. Nowadays robots are equipped with sophisticated sensors imitating the human sense of touch. This allows the recognition of an object based on information ensuing from robot-object physical interaction. Such information can include the object texture, compliance and material. This thesis exploits haptic data to perform haptic recognition of daily-life objects using machine learning techniques. The main challenge faced is the difficulty of collecting a fair amount of haptic training data for all daily-life objects. This is due to the continuously growing number of objects and to the effort and time needed by the robot to physically interact with each object for data collection. This thesis solves this problem by developing a haptic recognition framework capable of performing Zero-shot, One-shot and Multi-shot Learning. This framework is extended by integrating vision to enhance the robot's recognition performance, whenever such sense is available.

Keywords

- Classification
- Haptic recognition
- Visual recognition
- Zero-Shot Learning

Rattachement

Équipe de Recherche

IDH - Interactive Digital Humans

Laboratoire

LIRMM - Laboratoire d'Informatique, Robotique et Micro-électronique de Montpellier

Adresse

Campus St Priest - 860 rue St Priest
Bâtiment 5
34095 Montpellier cedex 5

Résumé Étendu

Introduction

L'intégration des robots dans les environnements domestiques nécessite le développement des robots autonomes qui peuvent percevoir et interagir avec leur entourage. Dans ce but, les robots domestiques ont été équipés de différents types de capteurs qui récoltent des données numériques sur les différents objets rencontrés. Ces données peuvent être exploitées, entre autres, afin de reconnaître l'identité de ces objets. Dans cette thèse, nous étudions le problème de la reconnaissance des objets de la vie quotidienne à partir de ces données sensorielles. Parmi les différentes modalités de données, nous nous intéressons aux données haptiques ; ce sont les données issues de l'interaction physique entre un objet et une main robotique, généralement équipée de capteurs tactiles.

Reconnaissance haptique

La reconnaissance haptique a fait l'objet de plusieurs études [71] utilisant différentes plateformes robotiques et jeux d'objets. Les approches récentes exploitent l'avancement des techniques d'apprentissage automatique pour entraîner un classificateur à prédire l'identité d'un objet manipulé, à partir des données haptiques collectées. Ces données ont des spécificités qui doivent être prises en considération lors du développement d'un système de reconnaissance haptique. Premièrement, le format des données dépendent fortement de la plateforme robotique et des capteurs tactiles utilisés, ce qui rend la généralisation de la méthode développée pour d'autres plateformes pas évidente. En outre, la récolte des données nécessite l'exploration des différentes parties de l'objet en utilisant la main robotique. Cette exploration haptique doit être efficacement effectuée afin de maximiser l'information obtenue sur l'objet, tout en évitant d'endommager le robot et l'objet. En plus, elle prend beaucoup de temps puisque quelques capteurs nécessitent de maintenir le contact avec l'objet pendant une période donnée pour une lecture plus robuste.
Cette thèse s'intéresse au problème de la difficulté de récolter des données haptiques d'entraînement pour tous les objets que le robot doit reconnaître dans la vie quotidienne. Le grand nombre d'objets dans les environnements domestiques, ainsi que le temps et l'effort requis pour explorer chaque objet ont rendu l'entraînement du robot n'est possible que pour un nombre restreint d'objets. A cet effet, il est important d'optimiser la

reconnaissance des objets n'ayant pas de données d'entraînement, ce qui est connu sous le nom *Zero-Shot Learning (ZSL)*, i.e. l'apprentissage à zéro tir.

Zero-Shot Learning

Le ZSL consiste à entraîner le robot sur un certain ensemble d'objets, et de le tester sur de nouveaux objets non inclus dans l'ensemble d'entraînement. Puisque on n'a pas de données d'entraînement pour les objets de test, leur reconnaissance doit être réalisée en exploitant les données collectées à partir des objets d'entraînement, ainsi que la relation entre ces derniers et les nouveaux objets de test.
Cette thèse propose les solutions suivantes pour la reconnaissance haptique des nouveaux objets, en intégrant la vision quand elle est disponible :

ZSL haptique

La première solution consiste à concevoir un système de reconnaissance haptique capable de reconnaître de nouveaux objets selon leurs attributs. Sachant qu'un attribut est une propriété physique ou sémantique de l'objet (e.g. long, rond, noir, métallique), on peut facilement décrire un nouveau objet en utilisant ses attributs. Cette description peut être fournie par un être humain ou automatiquement extraite à partir des bases de données sémantiques telles que WordNet. En décrivant les objets d'entraînement et de test en utilisant le même ensemble d'attributs, on peut utiliser les données disponibles pour entraîner un classificateur binaire par attribut. Ce classificateur apprend à prévoir la présence de l'attribut dans un objet. Ensuit, pendant la phase de test, les données collectées à partir d'un objet inconnu sont introduites aux classificateurs des attributs afin de prédire la présence ou l'absence de chaque attribut dans l'objet en question. Finalement, l'objet est classifié en tant que l'objet de test ayant les attributs les plus similaires.
Ce système est implémenté et adapté sur deux plateformes robotiques différentes. La première consiste à un préhenseur avec deux doigts équipés de capteurs BioTacs. La deuxième plateforme consiste à la main robotique Shadow qui imite la main humaine en terme de structure avec 5 doigts, et 19 degrés de liberté, et en terme de capacité tactile avec 5 capteurs BioTac sur les bouts des doigts. Chaque capteur BioTac peut mesurer des informations de pression, vibration, température et flux thermique. Également, le système est évalué en utilisant deux ensembles d'objets. Le premier est celui de la base de données haptiques PHAC-2 contenant 60 objets. Le deuxième consiste à un ensemble de 22 objets qu'on utilise dans notre quotidien. Cela permet de tester la solution proposée pour la main Shadow. Ce système de reconnaissance haptique est proposé dans [1] et décrit dans le chapitre 4.

ZSL visio-haptique

Les données haptiques décrivent les propriétés physiques d'un objet, liées à sa matière, texture et rigidité. En revanche, plusieurs travaux (e.g. [76]) ont montré l'efficacité de

leur fusion avec d'autres modalités afin d'améliorer les performances du système, en particulier la vision. La vision est la modalité la plus ancienne et largement étudiée pour la reconnaissance d'objets. L'intégration de la vision au système de reconnaissance haptique permet de percevoir plus d'informations sur la forme géométrique de l'objet ainsi que sur sa couleur.
Le chapitre 5 présente l'adaptation de la méthode basée sur les attributs, pour reconnaître les nouveaux objets qui n'ont pas été touchés ou vus par le robot pendant la phase d'entraînement [2]. Cette méthode utilise les réseaux de convolution afin d'apprendre la prédiction de la présence des attributs dans les objets. L'utilisation de l'apprentissage profond permet de mieux exploiter les données visuelles et d'améliorer les performances de classification des données tactiles. C

Reconnaissance avec une base de données d'entraînement déséquilibrée

Dans le cas d'un robot qui fonctionne dans un scénario réel, il peut rencontrer un objet qu'il a déjà rencontré pendant la phase d'entraînement, et donc il a suffisement de données d'entraînement sur cet objet, ou un objet nouveau. La solution proposée dans [3] et présentée au chapitre 6 traite les deux cas dans le même système. La méthode consiste à générer des données d'entraînement virtuelles pour les objets nouveaux et d'entraîner un système de reconnaissance basé sur les réseaux de convolution afin de reconnaître tous les objets.

Conclusion

Cette thèse propose plusieurs méthodes afin d'améliorer la reconnaissance d'objets dans les environnements incertains et dynamiques. En particulier, le but est d'optimiser la collecte des données haptiques, chose qui demande généralement beaucoup de temps et d'efforts. Les solutions proposées sont capable de reconnaître des objets de la vie quotidienne, qui n'ont aucune donnée d'entraînement. Cela se fait en se basant sur les données haptiques collectées à partir des objets d'entraînement ainsi que la relation entre les objets d'entraînement et les nouveaux objets.
Malgré les résultats prometteurs obtenus, les performances de reconnaissance peuvent être améliorées de plusieurs façons telles que l'utilisation des types plus avancés d'attributs comme les attributs relatifs et non-sémantiques. En plus, il est nécessaire de proposer une solution pour le problème de changement de domaine. Finalement, l'évaluation des méthodes proposée est basée sur un nombre restreint d'objet. Il est intéressant de tester les performances obtenues pour des bases de données haptiques contenant des centaines d'objets, chose qui n'existe pas actuellement.

Contents

Résumé iii

Abstract iv

Résumé Étendu vi

List of Figures xiii

List of Tables xv

Glossary xvi

Acronyms xvii

1 Introduction 2
- 1.1 Visuo-haptic recognition . . . 2
- 1.2 Learning from few data . . . 3
- 1.3 Contributions . . . 4
- 1.4 Manuscript structure . . . 5

2 State of the art of Object Recognition 6
- 2.1 Introduction . . . 6
- 2.2 Object Recognition . . . 6
- 2.3 Visual Object Recognition . . . 7
  - 2.3.1 Related work . . . 7
  - 2.3.2 Image datasets . . . 9
  - 2.3.3 Challenges . . . 10
- 2.4 Haptic Object Recognition . . . 10
  - 2.4.1 Human haptic sensing . . . 11
  - 2.4.2 Robotic haptic sensing . . . 11
  - 2.4.3 Robotic setups . . . 13
  - 2.4.4 Haptic exploration . . . 14
  - 2.4.5 Shape reconstruction . . . 16
  - 2.4.6 Machine learning for haptic recognition . . . 17

2.4.7 Multi-modal tactile recognition . . . 18
2.4.8 Active exploration . . . 18
2.4.9 Challenges . . . 19
2.5 Visuo-haptic Recognition . . . 20
2.6 Conclusion . . . 21

3 State of the art of Learning from Few Data 23
3.1 Introduction . . . 23
3.2 Zero-shot Learning . . . 24
3.2.1 Definition . . . 24
3.2.2 Attributes . . . 24
3.2.3 Textual description . . . 27
3.2.4 Mining relationships . . . 27
3.3 Visual Zero-Shot Learning . . . 28
3.3.1 Attribute-based approach . . . 28
3.3.2 Direct and hierarchical similarity . . . 29
3.3.3 Two-steps classification . . . 30
3.3.4 Predicting one-vs-all classifiers for novel classes . . . 30
3.3.5 Generating synthetic training data . . . 30
3.3.6 Domain Shift Problem . . . 31
3.4 Haptic Zero-Shot Learning . . . 31
3.5 Visuo-haptic Zero-Shot Learning . . . 31
3.6 One-Shot Learning . . . 32
3.6.1 Definition . . . 32
3.6.2 Visual One-Shot Learning . . . 32
3.6.3 Haptic One-Shot Learning . . . 33
3.7 Conclusion . . . 33

4 Haptic Recognition of Objects Never Touched Before 34
4.1 Introduction . . . 34
4.2 Haptic Zero-Shot Learning on PHAC-2 . . . 34
4.2.1 PHAC-2 database description . . . 35
4.2.2 Splitting the object set . . . 36
4.2.3 Feature Extraction . . . 36
4.2.4 Attributes classification . . . 37
4.2.5 BioTacs combination . . . 37
4.2.6 Attributes priors . . . 38
4.3 Evaluation and discussion . . . 38
4.3.1 Implementation . . . 38
4.3.2 Attributes classification . . . 38
4.3.3 Recognition of novel objects . . . 39
4.3.4 Robustness to the choice of objects . . . 40
4.3.5 Influence of attributes number . . . 42

4.3.6 Summary . . . 42
4.4 Zero-Shot Learning on daily life objects held by a robotic hand . . . 43
4.4.1 Motivation: Beyond PHAC-2 . . . 43
4.4.2 System overview . . . 43
4.4.3 Robot hand setup . . . 44
4.4.4 Daily life object set . . . 44
4.4.5 Attribute-based description . . . 45
4.4.6 Data collection . . . 46
4.4.7 Feature extraction and attributes classification . . . 47
4.4.8 DAP – handling robotic constraints . . . 48
4.4.8.1 Single grasp classification – Combining information from various BioTacs . . . 48
4.4.8.2 Single grasp classification – Handling local view . . . 49
4.4.8.3 Multi-grasp classification – Developing a global view . 49
4.5 Experimental evaluation and discussion . . . 50
4.5.1 Attributes classification . . . 50
4.5.2 Single grasp DAP . . . 51
4.5.3 Multi-grasp DAP . . . 52
4.5.4 Summary . . . 53
4.6 Conclusion . . . 54

5 Visuo-Tactile Recognition of Daily-Life Objects Never Seen or Touched Before 56
5.1 Introduction . . . 56
5.2 Visuo-Tactile Zero-Shot Learning . . . 57
5.2.1 Attributes Learning . . . 57
5.2.2 Visuo-Tactile DAP . . . 58
5.3 Experimental Setup . . . 59
5.3.1 Data Augmentation and Pre-processing . . . 59
5.3.2 Objects Splits . . . 60
5.4 Evaluation and Results . . . 60
5.4.1 Implementation Choices . . . 60
5.4.2 Attributes Learning . . . 60
5.4.3 Visuo-Tactile DAP . . . 62
5.4.4 Adding Visual Attributes . . . 64
5.5 Conclusion . . . 65

6 Deep Learning for Tactile Recognition of Known and Novel Objects 66
6.1 Introduction . . . 66
6.2 Tactile Object Recognition Framework . . . 67
6.2.1 Recognition Framework Overview . . . 67
6.2.2 Training FC_{FZ} . . . 68
6.3 Generating Synthetic Features for Novel Objects . . . 68
6.3.1 Solution Overview . . . 68

6.3.2 Classifying Training Objects 69
6.3.3 Training a Synthetic Features Generator 69
6.3.4 Generating Realistic Features 70
6.3.5 Generating Training Data for Novel Classes 71
6.3.6 Extension to One-Shot Learning 72
6.4 Experimental Setup 73
6.4.1 Dataset 73
6.4.2 Implementation Choices 73
6.5 Evaluation 74
6.5.1 Object Splits 74
6.5.2 Novelty Detection 75
6.5.3 Multi-class Classification of Known Objects 75
6.5.4 Evaluation of Synthetic Features Generation 76
6.5.4.1 Comparison with real features 77
6.5.4.2 Zero-Shot Learning 78
6.5.5 One-Shot Learning 79
6.6 Conclusion 81

Bibliography **82**

List of Figures

2.1 Number of layers and parameters and the top-5 error rate at the ILSVRC challenge for the winning deep architectures. 9
2.2 Example of robotic hands equipped with haptic sensing. 12
2.3 Examples of state of the art tactile sensors. 12
2.4 The PR2 robot. 13
2.5 The Shadow robotic hand. 14
2.6 Example of robotic setups equipped with BioTac sensors. 15
2.7 The BioTac tactile sensor. 15
2.8 An example of the BioTac raw uncalibrated readings given when contacting a rigid body. The BioTac in this case is mounted on the finger of a Shadow Hand grasping an object 16

3.1 Attribute-based ZSL (solution overview for N=5,M=4 and L=3): First, both Y and Z objects are described using A attributes. Then, a classifier f_m is learned for each attribute. Last, attributes classifiers are used by the DAP model to infer the object class. 29

4.1 Recognition accuracies and standard deviations of attributes binary classifiers. 39
4.2 Splitting the PHAC-2 objects set into five partitions. 41
4.3 Recognition accuracy vs. the number of attributes utilized for ZSL. 42
4.4 Our zero-shot recognition system overview. 44
4.5 Disjoint training (left) and test (right) object sets. 45
4.6 Examples of object grasps by the Shadow Hand. 47
4.7 Recognition accuracy of attributes binary classification. 50
4.8 Distance between attributes binary labels and their posteriors for test objects. 51
4.9 Representative grasps on the "measuring cup" show how different grasps may indicate distinct features. 52
4.10 Recognition accuracy vs. number of combined grasps T for DF-MDAP (a) and vs. number of similar classifications k for SC-MDAP (b). 53

4.11 Confusion matrices of LDAP , DF-MDAP ($T = 5$) and SC-MDAP ($k = 4$). SC-MDAP achieved perfect recognition of 8 out of 10 new objects. 54

5.1 Visuo-tactile CNN classifying an attribute as absent (0) or present (1) based on visual and tactile data. 58
5.2 Example of images taken for some of the PHAC-2 objects [19]. 59
5.3 Architecture of CNN for predicting attribute presence (1) or absence (0) based on tactile data. 61
5.4 Architecture of CNN for predicting attribute presence (1) or absence (0) based on visual data. 61
5.5 Attribute binary classification accuracies for all object splits averaged over all attributes (tactile SVM [1] in blue and tactile CNN in red). 61
5.6 Attributes classification accuracies for split 1: purple with tactile data alone, yellow with visual data alone, and green with both visual and tactile data. 62
5.7 Color attributes classification using visual images (bars are colored by the colors they represent). 64

6.1 Overview of our framework: recognition of known and novel objects. . . 67
6.2 Classification of training objects using CNN_{XY}: $CONV_{XF}$ is the convolutional part and FC_{FY} is the fully connected part. 69
6.3 Train G to generate features associated with objects in Y. 69
6.4 Adversarial training of G and D. 72
6.5 Train FC_{FZ} using data generated by G. 72
6.6 Test objects (framed in blue) with their attributes (right side) in Z for split 1 and examples of training objects with their attributes (right side). . 75
6.7 Tuning σ_{nov}: the accuracy of classifying X_{te} as known (blue) and X_{val} as novel (red) for split 1. 76
6.8 Confusion matrices and attribute-based similarity matrix of split 1. 80

List of Tables

2.1 List of popular datasets designed for object recognition and detection. . . 10

4.1 ZSL accuracy for independent attributes classifiers. 39
4.2 DAP accuracy for independent attributes classifiers. 40
4.3 DAP accuracy for challenging splits. 42
4.4 Class-attribute matrix $\boldsymbol{K}$ for training (upper) and test (lower) objects. . . 46
4.5 Recognition accuracy of DAP on test objects from a single grasp. 52

5.1 Comparison of tactile, visual and visuo-tactile ZSL recognition accuracies (%). 63
5.2 VT-SM-ZSL recognition accuracies (%). 63
5.3 Recognition accuracies (%) when adding color attributes to visual and visuo-tactile ZSL. 65

6.1 Neural Networks' hyper-parameters used in this work. 74
6.2 Accuracy of novelty detection ($\%$): distinction between known and novel objects. 76
6.3 Recognition accuracies ($\%$) for multi-class classification of Y with many training samples per object. 76
6.4 Recognition accuracies ($\%$) for Multi-Class classification (real training data) and ZSL (synthetic training data) when training FC_{FZ} using 0, 10, 50, 90 or 100 samples per class. 77
6.5 Recognition accuracies ($\%$) for ZSL using GEN-F and GEN-NN-F. 78
6.6 Recognition accuracies ($\%$) for ZSL with GAN and without GAN. 78
6.7 Recognition accuracies ($\%$) for ZSL with the method of [1] and GEN-NN-F. 79
6.8 Recognition accuracies ($\%$) for OSL. 79
6.9 ZSL and OSL recognition accuracies of 12 novel objects. 80

Glossary

Amazon Mechanical Turk a crowdsourcing platform, owned by Amazon, where multiple paid human participants perform complex tasks [148]. 25

Bag-of-words model a model used in Natural Language Processing for extracting features from textual data [164]. 17

Convolutional Neural Network a deep artificial neural network that extracts discriminative features from visual data and classifies them [145]. 56

Gaussian Mixture Model a probabilistic model that estimates the probability distribution of observations using multiple Gaussian distributions [124]. 67

Iterative Closest Point an algorithm used to align two point clouds by minimizing the difference between them [10]. 17

Principal Component Analysis a supervised dimensionality reduction algorithm, that extracts a set of uncorrelated variables from a larger set of correlated variables, while keeping most of the information [66]. 17

Recurrent Neural Network a type of artificial neural networks that takes into account the temporal information in the sequence of input data [47]. 17

Self-Organizing Map an unsupervised feature extraction algorithm based on artificial neural networks [81]. 17

Support Vector Machines a discriminative classification algorithm that learns a separating hyperplane for discriminating data samples from two classes [51]. 8

Tactile image the conversion of the readings of a tactile sensor to an image that represents the contact pattern. 17, 18

Acronyms

AMT Amazon Mechanical Turk. 26, 27

CCD Charge-Coupled Device. 20

CMYK Cyan, Magenta, Yellow, and blacK color model. 7

CNN Convolutional Neural Network. xiv, 56, 57, 58, 60, 61, 63, 64, 65, 66

DAP Direct attributes Prediction. xiii, 28, 29

DOF Degrees Of Freedom. 13, 14

EP Exploratory Procedures. 15, 18, 36, 60, 73

GMM Gaussian Mixture Model. 67, 74, 75

LED Light-Emitting Diode. 12

LS-SVM Least Squares Support Vector Machines. 32, 33

OSL One-Shot Learning. 24, 32, 33

PCA Principal Component Analysis. 36, 40, 47, 59

RGB Red, Green and Blue color model. 7

SOM Self-Organizing Map. 17

SVM Support Vector Machines. 26, 30, 37, 40, 47, 48, 57, 60, 61

ZSL Zero-Shot Learning. vii, xiii, 21, 22, 23, 24, 26, 28, 29, 30, 31, 32, 33, 34, 54

Chapter 1

Introduction

Over the past decades, industrial applications have shown the potential of robots in repetitive, hard and dangerous tasks. Industrial robots have had success in many factories, helping humans in terms of increasing productivity and reducing the danger to human labourers. This success has promoted the recent trend to bring robots into our domestic environments. However, the shift of robots from industry to everyday life is not trivial. While, industrial robots work in well-controlled environments and execute well-defined tasks, domestic robots need to operate in unstructured, dynamic and uncertain environments, surrounded by a wide range of objects. In fact, they need to interact with humans for household chores like cooking, cleaning and moving heavy objects, which are not well-defined tasks. Domestic robots therefore require a high level of autonomy to perceive their environment, decide appropriate actions and then perform them, without the assistance of humans.

A fundamental ability for the autonomy of robots is *object recognition*. This is a critical ability for a robot, as it enhances robot perception and its understanding of its environment. This thesis provides robots with the ability to utilize haptic as well as visual data collected from an encountered object to recognize it. Particularly, my work focuses on the recognition of objects that the robot has not experienced before, which is a frequent problem faced by robots performing in uncertain environments.

1.1 Visuo-haptic recognition

Vision is the most popular modality for performing object recognition. Since the early 1960s, many studies have investigated visual object recognition, leading to very efficient state of the art systems. However, visual recognition is still constrained by limitations, such as issues of lighting, scaling, viewpoint limitations and visual occlusion. Visual object recognition is also limited by its inability to quantify many physical properties such as the weight and compliance of objects. These limitations encouraged the design of robotic sensors for providing the robot with new sensing capabilities. Haptic sensing is probably the most promising of them, and is the focus of this thesis.

Modern robots are equipped with dexterous hands, which are usually covered by tactile skins, allowing to physically interact with surrounding objects. Through this physical interaction, the robot collects haptic data, which can be kinesthetic or cutaneous. Kinesthetic data is obtained from the kinematic readings (e.g. joint readings, hand position) of the robot hand and can describe the shape and the weight of the object. Second, cutaneous data are provided by tactile sensors (e.g. pressure, temperature, contact pattern) and can describe the object texture, compliance and material. Many studies showed the importance of these object properties to recognize objects (e.g. [69]), especially in the case of distinguishing visually similar objects.

1.2 Learning from few data

The state of the art is rich with many haptic object recognition systems designed for a wide range of robotic setups and applied for many object sets. Most of studies are based on multi-class classification, in which a classifier is trained to map haptic data into object classes using a set of *training objects*. The trained classifier can then classify any new haptic data sample as one of the training object classes. This approach has two main limitations: First, it neglects object classes that were absent during the training phase. For instance, if the robot is trained to recognize the following object classes: *bottle*, *book*, *cup* and *pen*, and the robot encounters a *ball*, then it cannot correctly recognize it since it tries to classify the ball as one of training objects. Second, this approach omits object classes having one or very few training samples as the classifier underfits them. Previous studies do not handle this problem since they collect the same amount of samples for all training objects. Therefore, this approach requires the robot to collect a fair amount of haptic data from every object class that the robot needs to recognize.

Multi-class classifiers, therefore, are not suitable for robots performing in domestic environments. In fact, domestic robots are surrounded by a wide variety of objects, and new objects are continuously added. Moreover, the collection of haptic data is time consuming since collecting each haptic data sample from an object requires the robot to physically interact with it. This takes time, as the robot hand needs to reach the desired position, and the tactile sensors need to collect good quality measurements. Furthermore, haptic data collection can also be laborious since many state of the art studies (e.g. [42, 134]) use human participants to hand the training objects to the robot. Thus, from a practical point of view, the robot can only be trained on a limited number of objects in a reasonable time. Therefore, the training set available to a robot is usually very scarce, as it includes a small amount of object classes and many objects do not have training data. Moreover, the training set can include some objects having only one or very few training samples.

There are many state of the art studies for recognizing objects having many training data. However, before this thesis, there was no solution for haptic recognition of novel objects without training data, or the so called *Zero-Shot Learning*, and only one study on the haptic recognition of objects having one training data, or the so called *One-Shot*

Learning. This thesis starts by designing the first zero-shot haptic recognition system. Then it extends to a system which can include vision. And finally, ends with adapting this system to objects with different amounts of training data and to non-experienced coming data.

The results of this thesis can enable a robot to:

- Generalize what it learned from a limited set of object to recognize novel ones;
- Handle efficiently imbalance in training sets;
- Exploit non-experienced data to continuously improve its recognition performance with experience.

1.3 Contributions

This thesis proposes a haptic Zero-Shot Learning system in [1], and improves it in [2] and [3].

- The first solution [1] focuses on how to exploit haptic data collected from training objects to recognize novel ones, without collecting any additional training data. This is performed by collecting semantic information about objects, which are easier to obtain than haptic data.
- The second solution [2] extends the first by considering the vision modality besides the haptic one. This fusion of modalities uses both haptic and visual properties of objects, which significantly improves the recognition performance.
- While the first two solutions focus only on the recognition of novel objects, the third [3] considers the recognition of many objects, having very different amount of training data, which can go from zero to tens of samples per object.

These contributions led to the following publications:

- ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. Haptic Zero-Shot Learning: Recognition of objects never touched before. Robotics and Autonomous Systems 105 (2018), 1125
- ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. Visuo-tactile recognition of daily-life objects never seen or touched before. IEEE Int. Conf. on Control Automation Robotics & Vision (ICARCV), 2018.
- ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. A deep learning framework for tactile recognition of known as well as novel objects. submitted to Transactions on Industrial Informatics (TII), 2018 (under-review).

1.4 Manuscript structure

Chapter 2 - State of the art of Object Recognition

I start by describing the problem of object recognition, where I also present the advances made in both vision and haptics domains in regard to object recognition. This chapter details the state of the art in object recognition algorithms, most of which assume the availability of a fair amount of training data for any object class.

Chapter 3 - State of the art of Learning from Few Data

The problem of absence or sparseness of training data for some object classes is the core of this chapter. I present the advances made in this regard (specifically Zero-shot and One-Shot learning) in the visual domain, and highlight the importance of these techniques to optimize data collection and labeling. Then, I focus on the necessity of performing Zero-Shot learning for haptic recognition.

Chapter 4 - Haptic Recognition of Objects Never Touched Before

The first solution proposed for performing haptic Zero-Shot Learning is presented in this chapter, in which I adapt a Zero-Shot Learning framework proposed for visual recognition to the constraints presented by haptic data. I test this framework on the state of the art PHAC-2 [19] haptic attribute dataset, and apply it to enable haptic (Zero-Shot) object recognition by an anthropomorphic robotic hand.

Chapter 5 - Visuo-Tactile Recognition of Daily-Life Objects Never Seen or Touched Before

This chapter exploits the encouraging results found using this first haptic Zero-Shot Learning system. It develops a zero-shot recognition system that can integrate available haptic and visual data in one deep object recognition system.

Chapter 6 - Deep Learning for Tactile Recognition of Known and Novel Objects

This chapter extends the work to develop a single integrated object recognition framework that can handle objects with different amounts of training data; enabling multi-class, One-Shot and Zero-Shot classification of encountered objects.

Chapter ?? - Conclusion

Finally, this chapter concludes the thesis, by summarizing limitations and detailing required future explorations.

Chapter 2

State of the art of Object Recognition

2.1 Introduction

This chapter reviews the state of the art on object recognition using robot's sensory data. Among the different data modalities that can be exploited to perform recognition, this chapter focuses on *vision*, which is the earliest and the most widely used modality for recognition, and on *haptics*, which is the core of this thesis. Most studies focus on only one modality, either vision or haptics. Thus, here, recognition based on each modality is presented separately. Then, the studies carried out on fusing both modalities to perform visuo-haptic recognition are reviewed.

2.2 Object Recognition

From a linguistic point of view, the verb "to recognize" means to "*perceive to be something or someone previously known*"[1] and to "*identify (someone or something) from having encountered them before; know again*"[2]. From a robotic point of view, the recognition task consists in classifying robot sensory data collected from an encountered object as one of previously known objects. This breaks down to solving a classification problem that maps sensory data to object classes.

Providing a robot with the ability of recognizing encountered objects starts with a preliminary phase, called the *training phase*. During this phase, the robot experiences a set of *training objects*:

$$Y = \{y_1, y_2, \ldots, y_N\}, \tag{2.1}$$

to collect sensory data from each one of them. This ensues a *training set*:

$$D_{train} = \{(y, x) \in Y \times X\}, \tag{2.2}$$

such that x is a data sample collected from object y, and all data samples are represented in the same data space X. Then, D_{train} is used to train recognition system on how to

[1] Merriam-Webster dictionary

[2] Oxford dictionary

predict the object class $y \in Y$ from which a data sample $x \in X$ is collected. In other words, D_{train} is used to learn a mapping:

$$f : X \longrightarrow Y. \tag{2.3}$$

Afterwards, during a second phase, called the *test phase*, the trained system should recognize new data samples $x \in X$ collected from *test objects*. This is performed by mapping the collected x into one of Y objects, which recognizes x as:

$$y = f(x). \tag{2.4}$$

The set Y can be defined according to two approaches. First, the "category-based" approach that considers all objects belonging to the same category as the same class (e.g. [54]). For instance, the recognition system proposed in [23] classifies different mugs as the same class "mug". Second, the "instance-based" approach that considers each encountered object as a specific instance, and classifies it as a separate class (e.g. [98]). For instance, in [134], the training object set includes two mugs, however, the system is trained on recognizing each one of them separately.

Furthermore, the input data space X is heavily dependent on the robot's sensing capabilities. Thus, the quantity, quality and nature of the available data must be taken into consideration while developing an efficient recognition algorithm. This thesis focuses on two input modalities: images, used to perform visual recognition, and haptic data, used to perform haptic recognition. Although the main focus of this thesis is the haptic modality, it is inevitable to consider the visual modality since it is the earliest and most used modality to perform object recognition. The rest of this chapter presents how each modality is used separately or together to perform object recognition.

2.3 Visual Object Recognition

Visual object recognition is an important problem that has been widely studied in computer vision during the last decades. While humans naturally and effortlessly use vision to perceive surrounding objects, many algorithms are developed to solve the problem of recognizing objects appearing in an image. Despite the massive advance made, visual recognition is still an open research area that draws researchers attention.

Vision-based recognition is the problem of mapping digitized images into object classes. A digitized image is a matrix of $n \times m$ pixels, where each pixel can have a binary value for black and white images ($X = \{0, 1\}^{n \times m}$), a real value for grey scale images ($X \subset \mathbb{R}^{n \times m}$) or a d-dimensional vector for colored images ($X \subset \mathbb{R}^{d \times n \times m}$), e.g. $d = 3$ for RGB and $d = 4$ for CMYK.

2.3.1 Related work

Vision is the earliest modality used for providing robots with the ability of recognizing surrounding objects. The human visual perception and ability of recognizing objects from

their images was studied in Cognitive psychology [151, 46]. Afterwards, the transfer of this ability to machines has gained considerable attention since the early 1960s. At the present time, a huge progress is made and impressive recognition accuracies are reached.

The high-dimensionality of images is the first faced challenge due to the "curse of dimensionality" problem [9, 73]. Early studies approach this problem by designing hand-crafted feature vectors that represent the high-dimensional raw images in a low-dimensional feature space. Then, state of the art classifiers such as Support Vector Machines, neural networks and Bayesian classifiers classify the extracted features. A detailed review about this approach is given in [44].

Manually designed feature extraction were latter replaced by deep neural networks that automatically learn discriminant features and classify them at the same time. Despite the early studies addressing deep learning (e.g. [86]), the success of AlexNet [83] in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC2012) showed the efficiency of deep learning in performing object recognition. This is the actual beginning of abandoning hand-crafted feature extractors in favour of deep convolutional neural networks. This is clear in the number of deep convolutional networks contributing in the next ILSVRC editions, including popular ones such as VGGNet [138], GoogleNet [145] and Microsoft ResNet [50].

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) compares classification algorithms, classifying 1.2 million images to 1000 classes. Designing deep architectures is not evident since naively adding more layers to the convolutional network decreases the recognition accuracy. In fact, increasing the number of layers expands the parameter space by adding unnecessary parameters which leads to over-fitting. In addition, expanding the parameter space makes the optimization harder, and thus leads to a higher training error, which is called the degradation problem. Moreover, the error back-propagated from the last layers vanishes when it reaches the first ones due to the big number of layers. This means that the first layers are neglected during the training since the change of their weights is minimal, which is called the vanishing gradient problem. Therefore, handling this problems is necessary for ensuring a good classification accuracy when stacking more layers. This is considered in the following winning deep networks at the ILSVRC challenge to improve the accuracy:

- **AlexNet** [83] (1^{st} place at ILSVRC2012): This architecture includes 5 convolutional layers followed by 3 fully connected ones. AlexNet solves the vanishing gradient problem by replacing the **tanh** and **sigmoid** activation functions by the **ReLU** activation [109]. In addition, over-fitting is reduced by using a Dropout layer after each fully connected one.

- **VGGNet** [138] (2^{nd} place at ILSVRC2014): This network improves AlexNet by replacing large kernel-sized filters with multiple stacked small-sized ones, allowing to stack 16 convolutional layers followed by 3 fully connected ones. This architecture suffers from a very high computational cost.

- **GoogleNet** [145] (1^{st} place at ILSVRC2014): This network improves the com-

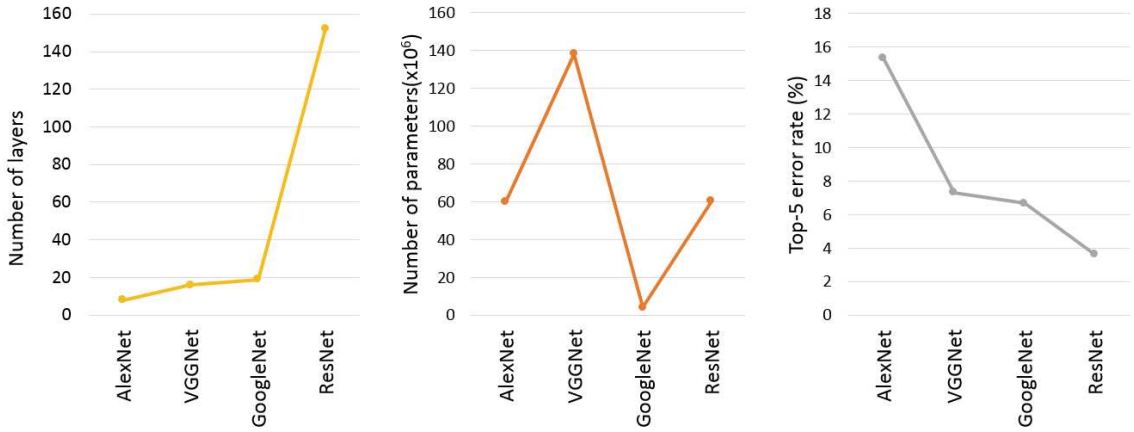


Figure 2.1: Number of layers and parameters and the top-5 error rate at the ILSVRC challenge for the winning deep architectures.

putational cost by removing unnecessary and redundant neurons. It uses the *inception module* to obtain a sparse CNN. In addition, the number of parameters is further optimized by replacing the fully connected layers with global average pooling. GoogleNet consists of 22 convolutional layers having 4 million parameters, compared to 60 million for AlexNet and 138 million for VGGNet.

- **ResNet** [50] (1^{st} place at ILSVRC2015): The number of layers is drastically increased to 152 layers, while keeping a low computational cost and increasing the recognition accuracy compared to the previous architectures. The key idea is to utilize skip connections that can skip some layers to avoid the vanishing gradient. The skip connections allow to start the training with few layers, then gradually consider new layers that learn more features until considering all layers.

Fig. 2.1 summarizes the characteristics of the architectures presented above. It illustrates how each architecture improves the previous one by stacking more layers to reduce the classification error. The number of parameters indicates the computational cost of each architecture.

2.3.2 Image datasets

One of the main reasons contributing to the progress of visual recognition is the construction of public datasets that make available labeled images for a great number of object classes. It is worth mentioning popular datasets such as Caltech 101 [33], Caltech-256 [45], LabelMe [130], CIFAR-10 and CIFAR-100 [82], Berkeley 3-D Object [58] and SUN [158]. As presented in table 2.1, the number of classes and images kept growing until the launch of ImageNet, the most popular dataset [25] which contains more than 14 million images labeled following the WordNet [106] hierarchical structure. This wide range of datasets allows to avoid the long process of image collection and labeling. Besides, it makes comparisons of different algorithms applied on the same dataset possible.

Table 2.1: List of popular datasets designed for object recognition and detection.

Dataset	Creation	Number of images	Object categories
Caltech 101	2003	9146	101
Caltech-256	2007	30.607	256
LabelMe	2005	187,240	32164
CIFAR-10	2009	60,000	10
CIFAR-100	2009	60,000	100
Berkeley 3D Object	2014	849	50
SUN	2014	131,067	4479
ImageNet	2009	14,197,122	21,841

2.3.3 Challenges

Despite the huge advance made in visual recognition, researchers are still facing some challenges related to vision:

- Vision can be unclear due to the changing lighting conditions, the darkness or the fog;
- The pose of the object and the viewpoint from which the image has been taken can give very different appearances;
- Since objects are not usually isolated in the image, recognition in cluttered scenes is more realistic. The image in this case includes unrelated objects that can even hide a part of the relevant object;
- Some objects may have the same visual appearance but have different material or compliance properties that make them different. Using vision alone is unable to distinguish between them, requiring to integrate data modalities other than vision. The focus of this thesis is on integrating the haptic modality.

2.4 Haptic Object Recognition

The great success of visual object recognition do not exempt from the need to integrate new data modalities to enhance the recognition performance. In fact, vision misses some important object properties (e.g. compliance) that can be discriminant for the recognition task. Inspired by the behavior of children who are usually not content with their vision of the object and want to know more about it by exploring it using their hands, haptic recognition has gained more interest as a way to cope with visual recognition deficiencies.

2.4.1 Human haptic sensing

The word "haptic" was derived from the greek word *haptikos* that means "able to touch or grasp" [3]. While tactile sensing is related to the human skin, specifically to the sense of touch, haptic sensing encompasses both the skin and the hands. Lederman and Klatzky [89] divide the information related to the haptic sensing into: cutaneous and kinesthetic information. When a hand explores an object, the skin's mechanoreceptors and thermoreceptors respond to external stimuli by providing cutaneous information, while kinesthetic information is provided by mechanoreceptors of interacting muscles, tendons and joints.

On one hand, cutaneous receptors provide information about physical properties that vision cannot provide. Different types of mechanoreceptors embedded in the skin such as Meissner's and Pacinian corpuscles detect vibrations, which are critical for perceiving roughness. In addition, thermoreceptors provide thermal information necessary for distinguishing materials according to their thermal properties. Furthermore, cutaneous information is critical for stable hand-object physical interaction. This is demonstrated in several studies showing that human subjects face difficulties in manipulation [63] and typing [41, 119] tasks when their fingertips are anesthetized. The effect of tactile impairment on tactile object recognition tasks is studied in [154]. Experiments show that subjects with deficient sensing are less effective in discriminating the roughness of different objects.

On the other hand, kinesthetic cues ensued from mechanoreceptors are used to perceive more properties such as weight, shape and size [78]. Some prejudices that are made when seeing objects can be found wrong when physically interacting with them. For instance, when someone sees two objects, he/she often thinks that the biggest object is the heaviest. The opposite can be confirmed when lifting both objects. Also, someone can be mislead by a far object that looks smaller than a close one, while this is not the case when exploring them using hands. Moreover, combined with cutaneous data, kinesthetic data allow to perceive compliance properties.

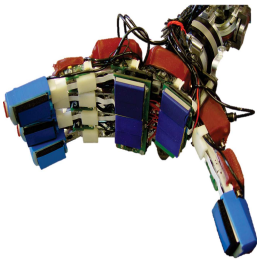
2.4.2 Robotic haptic sensing

Robotic hands Transferring human haptic sensing to robots has been addressed by many researchers. On one hand, numerous studies address the design of artificial hands and grippers, having as final goal to mimic human hand dexterity in manipulation, interaction and grasping. According to [21], studies are carried out in six main axes: kinematic architecture, actuation, transmission, sensing, materials and manufacturing. As a result of decades of research, a wide variety of robotic hands are designed. Among them different grippers (e.g. WRT102 Parallel Gripper [110]), multi-fingered hands (e.g. UtahIMIT dextrous hand [57], U.B. Hand Version 2 [102] and Schunk Dexterous Hand [75]) and anthropomorphic hands (e.g. LUCS Haptic Hand 3 [65] and ARMAR-IIIb Hand [6]). Other hands integrated in human robots are the hand of iCub [105] and TWENDY-ONE [56]. Some of these robotic hands are illustrated in Fig.2.2. An in-depth review on the state of art of artificial hands can be found in [21].

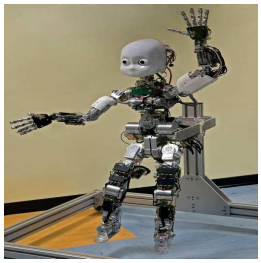
[3] Source: `https://en.oxforddictionaries.com/definition/haptic`

(a) WRT102 Parallel Gripper [110]

(b) Schunk Dexterous Hand [110]



(c) ARMAR-IIIb Hand [110]

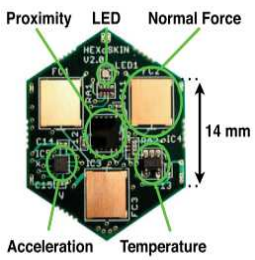


(d) iCub [105]

Figure 2.2: Example of robotic hands equipped with haptic sensing.

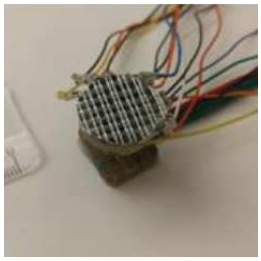
Tactile sensors Many tactile sensors are designed to provide the hands with tactile sensing. Nowadays tactile sensors are able to provide information about the object's texture, hardness and temperature, as well as contact force and pattern. For instance, in [22], a simplified and inexpensive tactile sensor was designed. The sensor consists of a silicone rubber hemisphere including 8 illuminating LEDs (see Fig 2.3a). When an object contacts the sensor, the contact pattern is estimated from the shading pattern resulting from the silicon deformation. CellulARSkin [108] is a tactile sensor capable of measuring vibrations, pressure and temperature (Fig 2.3b). A piezoelectric sensor was designed in [4] with a high sensitivity for applied forces (see Fig. 2.3c). Authors of [24] designed a bio-inspired tactile sensor, capable of measuring multiple modalities such as temperature, pressure, acceleration and strain (see Fig 2.3d). Furthermore, the haptic perception of a robotic hand dramatically improves by adding tactile perception, as is the case of the anthropomorphic hand developed by [137] when equipping it with tactile sensors in [40], and that of the Shadow Hand [4] when mounting a BioTac sensor on each fingertip. In-depth reviews on the state of the art on tactile sensors, their types, application, their integration in robotic hands and their use for multiple tasks can be found in [21, 71].

(a) Inexpensive tactile sensor [22]



(b) CellulARSkin [108]

(c) Soft PZT Based Tactile Sensor [4]



(d) Bio-inspired tactile sensor [24]

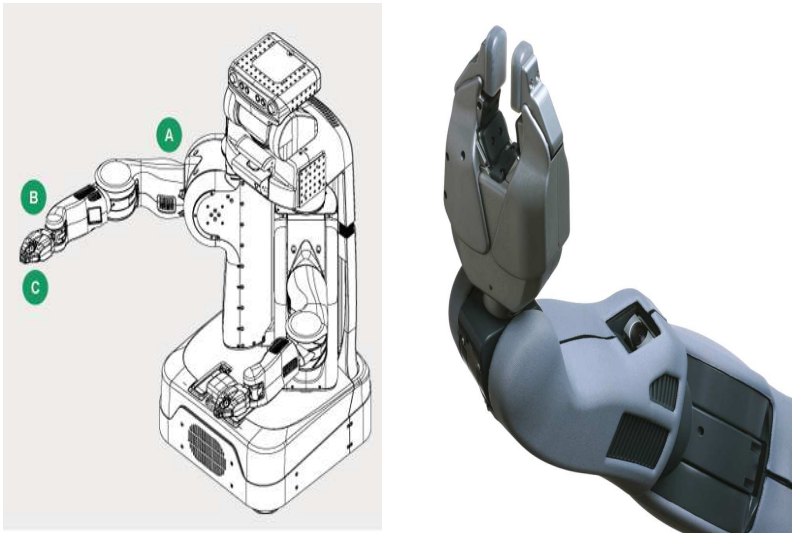
Figure 2.3: Examples of state of the art tactile sensors.

[4] http://www.shadowrobot.com/%20products/dexterous-hand/

2.4.3 Robotic setups

In this thesis, the solutions proposed for object recognition are applied on two robotic setups: the gripper of the PR2 robot equipped with two BioTac sensors (see Fig. 2.6c) and the Shadow Hand equipped with five BioTac sensors (see Fig. 2.6d).

PR2 robot gripper This robot is used in [19] to build the PHAC-2 dataset. The Willow Garage PR2[5] is a personal robot developed for research in robotics. It is designed to navigate in human environments and to manipulate the encountered objects. As illustrated in Fig. 2.4a, the PR2 has two arms, each with 7-DOF. Its end-effector on each arm is the 1-DOF gripper illustrated in Fig. 2.4b. The PR2 is equipped with different types of sensors such as a Microsoft Kinect camera on its head and a pressure sensor array on each of the gripper fingertips. In order to improve its haptic sensing, the robot is equipped in [19] with two BioTac sensors, one on each fingertip. This allows to perceive more tactile properties of the manipulated objects.



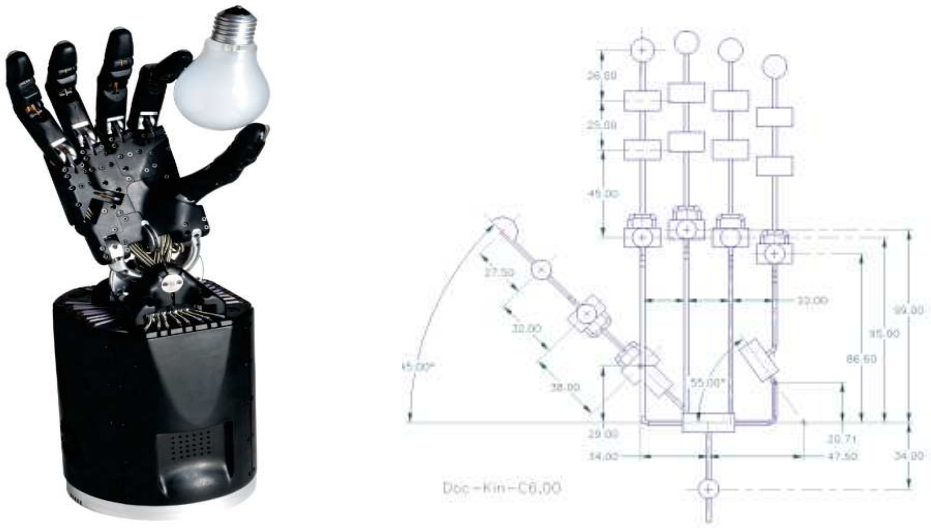
(a) PR2 design[5] (b) PR2 gripper[5]

Figure 2.4: The PR2 robot.

Shadow robotic hand The Shadow robotic hand is an anthropomorphic hand designed by the Shadow robot company[6]. As illustrated in Fig. 2.5, it has a total of 24-DOF, 2-DOF for the wrist, 4-DOF each for the index, middle and ring fingers, and 5-DOF for the thumb and little fingers. The hand is equipped with different types of sensors such as position sensors on each joint and force sensors for each pair of tendons. To improve

[5] http://www.willowgarage.com/pages/pr2/overview
[6] https://www.shadowrobot.com/

tactile sensing of the Shadow Hand, it can be equipped with BioTac sensors, one for each fingertip, reducing the DOF to 19 since this deactivates the distal phalanges of all fingers.



(a) Robot image (b) Kinematic model[4]

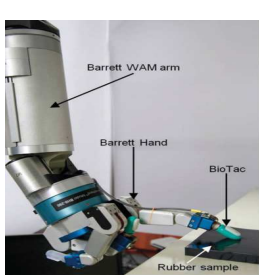
Figure 2.5: The Shadow robotic hand.

BioTac sensor BioTac is a tactile sensor designed by the SynTouch company[7]. It mimics the human fingertips in terms of compliance and sensing capabilities [34]. Fig.2.6 illustrates some examples of robotic setups equipped with BioTac sensors for improving tactile sensing. BioTac consists of a rigid core separated from a covering deformable skin with a fluid (see Fig. 2.7). The contact of the BioTac skin with an object makes the skin and fluid deform. This is sensed by the 19 electrodes distributed on the surface of the BioTac core, see Fig. 2.7c. In addition, the contact generates vibrations that are sensed by a hydro-acoustic pressure transducer. Besides, the heat flow between the BioTac and the object is sensed by a thermistor. Therefore, a BioTac contacting, or sliding across, the surface of an object can measure five types of raw data: absolute fluid pressure (Fig. 2.8a), dynamic fluid pressure (Fig. 2.8b), temperature (Fig. 2.8c), heat flow (Fig. 2.8d), and voltage change for each of the 19 electrodes (Fig. 2.8e).

2.4.4 Haptic exploration

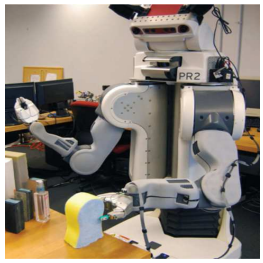
The human ability of perceiving and recognizing objects via haptic exploration is studied in cognitive psychology. Authors of [87, 88] studied how a human explores an object using his/her hand to perceive its physical properties. The result of these studies is the definition of eight elementary Exploratory Procedures (EPs), and the corresponding sensed

[7] https://www.syntouchinc.com/en/

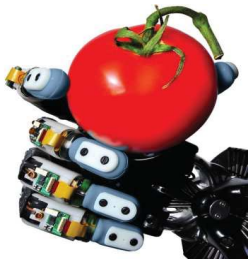


(a) Barrett with BioTac [144]

(b) PA10 robot equipped with a BioTac finger [53]



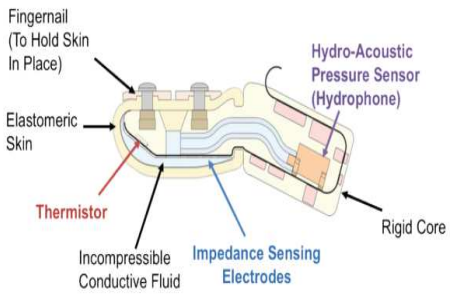
(c) PR2 robot gripper equipped with two BioTac fingers [19]



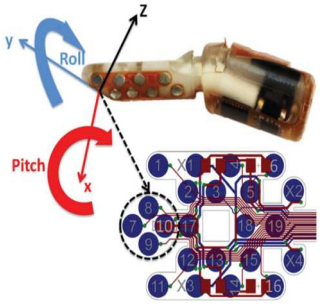
(d) Shadow hand equipped with BioTac fingertips[4]

Figure 2.6: Example of robotic setups equipped with BioTac sensors.

(a) BioTac image [34]



(b) BioTac design [34]

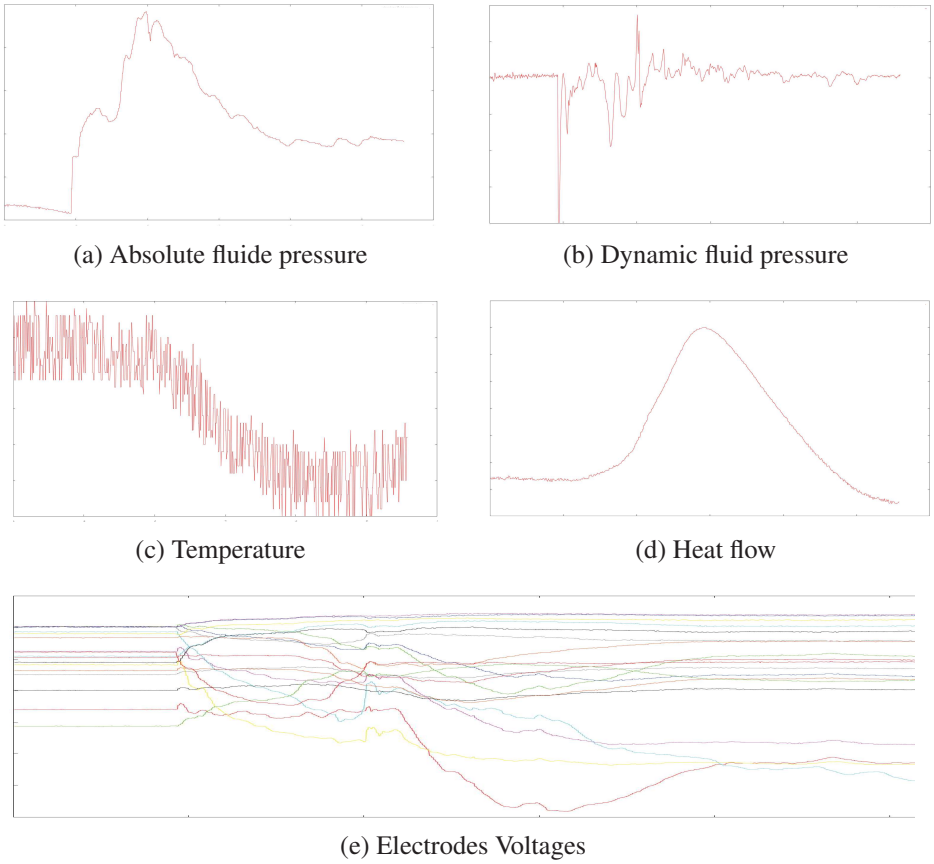


(c) BioTac electrodes distribution and orientation [144]

Figure 2.7: The BioTac tactile sensor.

properties: (1) *lateral motion* for texture properties, (2) *pressure* for hardness properties, (3) *static contact* for thermal properties, (4) *unsupported holding* for weighing, (5) *enclosure* for estimating volume and shape, (6) *contour following* for more obtaining details about volume and shape, (7) *part motion* for objects having moving parts and (8) *function testing* to test the functions that can be performed using the object. Authors of [80] studied how haptics alone, or combined with vision, is discriminative to perceive objects properties, particularly size, shape, texture and hardness.

Results found by cognitive psychology studies are the starting point for providing robots with haptic perception. Exploratory procedures defined by [87] are widely used for exploring objects using robotic hands and tactile sensors. According to the robot sensing capabilities and the desired object properties, one or multiple EPs can be combined to define a complete exploration strategy. The next section presents state of art studies making use of collected haptic data to recognize explored objects.



(a) Absolute fluide pressure

(b) Dynamic fluid pressure

(c) Temperature

(d) Heat flow

(e) Electrodes Voltages

Figure 2.8: An example of the BioTac raw uncalibrated readings given when contacting a rigid body. The BioTac in this case is mounted on the finger of a Shadow Hand grasping an object

2.4.5 Shape reconstruction

Early studies perform shape-based recognition by exploiting robot-object interaction data. Contact points and joints readings are used to estimate a geometric model that best fits the object shape. Then, recognition is performed by matching the estimated model with geometric models of training objects, which are stored in a database. In [5], kinesthetic data collected from grasping objects using a multi-fingered robotic hand are used to model objects as superquadrics. Recognition is performed by matching the recovered model with the superquadric parameters of five daily-life objects. In [15], the authors simulate a grasping exploration of objects, ensuing a point cloud and the estimated normal vectors. Then, they perform a volumetric approximation of the object using polyhedral models. Matching the recovered model with the training ones is performed using a multilayer neural network. These studies are limited to convex objects only.

In [101], objects are explored using a three-fingered robot hand equipped with tactile sensors. The shape of the object is estimated using a point cloud and the matching is

performed using Iterative Closest Point method. In [43], a 8-DoF hand equipped with tactile sensors is used to construct a point cloud of the explored object. The authors propose a features extraction method of the point cloud and use k-nearest neighbor in features space for recognition. They assume that the point cloud is complete in every region of the object, which is not always the case in haptic exploration with random shaped objects. The authors of [62] simulate the exploration of superquadric shapes. The resulting point cloud is replaced with a fixed length feature vector and a Gaussian process is used for classification. Two main limitations of the previous studies: First, the object pose must be fixed during the whole exploration time, and the kinematic model of the robot must be known. Second, these studies are based only on object's shape, which does not exploit cutaneous information.

2.4.6 Machine learning for haptic recognition

Instead of reconstructing the object's shape, a more recent approach recognizes the object directly based on haptic raw data such as joint readings, Tactile images, vibration signals. Machine learning techniques for feature extraction and classification are investigated in multiple studies to extract haptic features directly from raw exploration data and to classify them. In this approach, the pose of the object can change from an exploration step to the other, and there is no need for a reference coordinate system.

In [131], primitive objects are rolled on a planar force sensor, and the object is recognized based on the time evolution of collected Tactile images. The data type collected from the force sensor limits the generalization to randomly shaped objects. In [64], an anthropomorphic robot hand is used to repetitively grasp primitive objects and Self-Organizing Maps are used to map proprioceptive sensory data into object classes. In [135], Bag-of-words model is used to recognize objects using a robotic gripper equipped with tactile sensors. In [54], an anthropomorphic hand is proposed and used to repetitively grasp primitive objects, and Recurrent Neural Network is used to map Tactile images into object classes. In [42], each one of seven household objects is grasped using a five-fingered robotic hand, SOMs are used for feature extraction and Bayes classifier for recognition using the collected kinesthetic and tactile data.

Each of the previous studies evaluates the proposed algorithm on a specific robotic setup and object set which does not give an idea on their performance when applied to new setups and objects. To cope with this problem, the authors of [110] test their recognition system on three different robotic hands and three sets of real-life objects. Joints and tactile readings collected from each object by repetitively grasping it are used to recognize the object using SOMs and neural networks.

Next, the wide success of deep learning in vision-based recognition encouraged [134] to use deep neural networks to recognize twenty daily-life objects. High-dimensional kinesthetic and cutaneous data are collected by grasping each object at least 20 times using an anthropomorphic robotic hand. Results show how deep architectures outperform Principal Component Analysis combined with a shallow network.

While previous studies consider sensory data without taking into consideration the

time dimension, [98] propose a spatio-temporal hierarchical descriptor that captures temporal features from time series of tactile data, in addition to spatial features from each Tactile image. In [23], a simple F/T sensor mounted on a robotic arm automatically explores an object given its position in the workspace and a Naive Bayes classifier is used to perform instance-based and category-based recognition of ten household objects. In [97], the popular SIFT visual descriptor is adapted to a tactile-SIFT. The authors use training data to generate a dictionary of tactile-SIFT descriptors, and then a histogram of word occurrences for each training class. Recognition of a touched object is performed by comparing the resulting histogram to training classes histograms.

2.4.7 Multi-modal tactile recognition

All previous systems use mainly pressure information and shape information. A main difficulty faced by the majority of works is the difficulty in distinguishing between similar objects. This highlights the need for improving the robots sensing capabilities to perceive more physical properties such as material and texture. For instance, the authors of [53] perform a material-based recognition by exploring different flat objects using a BioTac sensor mounted on a robotic arm. The authors compare multiple classifiers and feature extraction techniques to distinguish eight different materials. Only cutaneous information is used and objects shapes are not taken into account. The authors of [69] perform texture-based recognition of twenty objects by sliding BioTac fingertips mounted on the Shadow robot hand.

2.4.8 Active exploration

In previous studies, the exploratory strategy is often random or predefined. For instance, in [42, 110] the object is repetitively grasped in multiple random positions and orientations. Exploration time and efficiency can be improved by developing an algorithm that generates the sequence of actions the robot should perform, which is known as *active exploration*. The key idea is to choose the next exploratory action that maximizes the information gain. This reduces the exploration time and improves the recognition accuracy by carefully choosing interesting regions or features for the recognition.

In [135], at each exploration step, the next generated arm height is the one that minimizes the entropy of object posterior. The authors do not take into account the optimization of the gripping process. Authors of [12] use dynamic potential fields to guide a simulated five-fingered robot hand mounted on an arm along the object surface. The goal is to recover the point cloud that best fits the object. In [42], the exploration space is decomposed into voxels by associating each voxel with an attention value. The used simulated hand, is guided at each step to the voxel having the highest attention value. In [118], a simulated robotic arm with a planar tactile array is guided to actively choose the contact direction and force to improve the quality of the Tactile images. The choice of the optimal EP among several ones is studied in [159] using the Shadow Hand. The authors test their algorithm on ten objects having different compliance, texture and thermal prop-

erties. Authors of [146] learn from training data an observation model that models the relationship between: observed data, explored object parameters and exploration action parameters. This is applied to a three-fingered Shadow Hand when exploring ten cylindrical objects. Finally, the authors of [99] reduce tactile readings uncertainty by guiding the robot towards discriminative locations. They apply their method on three-fingered robotic hand when exploring six objects having distinguishable primitive shapes.

Further challenges are targeted in recent studies such as replacing the long exploration strategies by a single unplanned grasp in [91, 143]. The authors use a 2-fingered robot hand equipped with barometric pressure sensors to grasp primitive objects having different size and stiffness properties. Authors of [141] study incremental learning instead of performing a whole grasping sequence to perform recognition. Their online recognition method allows an accurate classification after only 30% of the whole grasp sequence. They test their method by grasping nine different everyday objects using the iCub humanoid hand. Another improvement proposed by [160, 92] consists in exploiting the intrinsic relationships between the hand's fingers. When grasping an object, instead of considering all tactile data identically, a joint kernel sparse coding is used to group readings from the same finger. The authors test their algorithm on a set of empty and full water plastic bottles.

2.4.9 Challenges

Haptic recognition is a domain that has made the most of the visual recognition and pattern recognition state of the art. However, it still suffers from multiple obstacles that make it a challenging task:

- Sensory data are often noisy due to the robot-object interaction and sensors quality and accuracy;
- The data sparsity is a frequent problem faced when modeling the object shape using a point cloud constructed from robot-object contact points. Compared to dense point clouds that can be extracted from vision, point clouds gathered from haptic exploration are very sparse and can miss important parts of the object;
- State of the art studies on haptic recognition use a wide variety of object sets and robotic setups, which makes their comparison very difficult. Some studies use primitive objects whereas others use industrial or household objects. In addition, some studies use real robots and objects while others simulate an exploration strategy on virtual objects that can be superquadrics, points clouds or geometric models. To solve this last challenge, multiple haptic databases have been designed and made available to the research community [136, 7, 142, 26, 19]. However, the number of objects and the variety of robotic setups is still very modest compared to image databases such as ImageNet [25].
- The format of haptic data is heavily dependent on the sensory capabilities of the robot. The kinematic model of the hand and the modalities measured by the tactile

sensors differ a lot from a setup to another. This requires an adaptation of the method according to each experimental setup;

- The robotic hand or tactile sensor are often smaller than the object size. This gives a local and incomplete view of the object, requiring the combination of several exploratory steps from different parts of the object;

- The cost of collecting haptic data is very high. It can be labour intensive if the object is handed to the robot by a human operator. In addition, it is time consuming since each exploration step may require few seconds to allow stable tactile measurements. Moreover, it requires robot-object interaction which must avoid robot auto-collisions, collisions with the object and with the workspace and guarantee safety of the object and the robot;

2.5 Visuo-haptic Recognition

Sections 2.3 and 2.4 present the progress made in visual and haptic recognition as well as the challenges related to each modality. This section shows how fusing both modalities to perform a visuo-haptic recognition can help cope with problems faced when using modality separately.

The authors of [79] studied when humans decide to improve their visual perception of an object by haptically exploring it. According to the targeted physical properties, participants may make their judgment using vision only or they may use their hands to enhance their judgment. For instance, results show that geometric judgments are often made by merely seeing the object whereas for determining the material , participants quickly decide to touch the object. This psychological study highlights the importance of each modality to improve the perception of certain physical properties.

Visuo-tactile sensory fusion is studied in [76] to improve 3D object recognition in terms of recognition accuracy and number of learning iterations. The authors train a neural network on classifying the object images collected from a CCD camera, merged with haptic data collected by grasping the object with a four-fingered robot hand. Authors of [133] show how haptic feedback can improve visual curvature estimation. In addition, they show how this visuo-haptic curvature estimation can be beneficial for category-based object recognition. Another study showing the power of visuo-haptic fusion is [49]. The goal is to predict what is inside a container by squeezing it using a robot hand. The resulting visual and tactile deformation is used to predict if the container is empty or, if full, whether its content is solid or liquid. Results show that both modalities are complementary. Authors of [38] address robot tactile understanding based on both visual and tactile data. They use the PHAC-2 dataset [19] that provides a list of haptic adjectives that describe a set of objects. The authors provide the robot with the ability of describing an explored object using these adjectives from both visual and tactile exploration data. The authors of [165] perform material-based recognition by jointly using surface images and haptic signals gathered while sliding an acceleration sensor across the object surface.

Finally, visuo-haptic material recognition is studied in [95] to perform weakly paired recognition of six materials using the PHAC-2 dataset. Results show that combining visual and haptic data is not an evident task and that naive combination methods may lead to worse results than each modality separately.

2.6 Conclusion

This chapter presents the progress made during the last few decades in understanding the human behavior during recognition using vision, haptic sensing and both of them together, in addition to how to transfer this ability to robots. First, Sect. 2.3 presents visual recognition, the earliest and the most widely studied modality. This section shows that despite the great advance in visual recognition, vision alone is not enough to perceive all objects properties, which are important for recognition. This requires the integration of new sources of information, such as haptics.

After presenting in Sect. 2.4 the state of the art of haptic recognition, Sect. 2.4.9 summarizes multiple challenges faced while developing haptic recognition algorithms. These challenges were taken into account while developing our haptic and visuo-haptic recognition algorithms presented in the next chapters as follows:

- The dexterity of the Shadow hand used in chapter 4, as well as, the use of multiple BioTacs and the variety of data modalities measured by each one of them give kinesthetic and cutaneous information about the object properties. This rich amount of information copes with the problems of haptic data quality, accuracy and sparsity;
- To avoid the dependency of results to the chosen object set and robotic setup, the experimental evaluation in chapter 4 is performed on two experimental setups. The Shadow hand with BioTacs exploring 20 daily-life objects, which is specific to this thesis, and the PHAC-2 dataset, which is a public dataset allowing to compare our results with other algorithms proposed for the same task;
- Since the data format depends heavily on the experimental setup. The theoretical solutions proposed for performing recognition are adapted to each robotic setup. This allows to make the most of the sensing capabilities of each setup;
- The evaluation objects used in this thesis have a variety of geometric properties. Some objects are bigger than the robotic hand, which requires the combination of multiple exploration steps. Therefore, two solutions are proposed to perform multi-grasp recognition, which consists in recognizing an object based on data gathered by grasping multiple parts of the object;
- To cope with the high cost of collecting haptic data, we perform haptic Zero-Shot Learning. This avoids collecting haptic data to perceive each new object by generalizing the knowledge acquired from known objects. Two proposed solutions for performing ZSL based on haptic data only are presented in chapters 4 and 6;

- Finally, the haptic recognition performance is improved by proposing in chapter 5 the integration of vision.

Next chapter presents how ZSL can cope with the problem of data scarcity by recognizing objects having no training data. Then, the following chapters present the proposed solutions for adapting ZSL to haptic and visuo-haptic data.

Chapter 3

State of the art of Learning from Few Data

3.1 Introduction

The previous chapter reviews different state of the art solutions for performing object recognition based on robotic visual and haptic sensory data. The robot first explores a set of training objects and collects multiple data samples from each one of them. A multi-class classifier is then trained to map each data sample to the object class from which it is collected. Next, during a test phase, the robot should recognize an encountered object. The robot explores the object and uses the trained classifier to predict from which training object the test data sample is collected. Therefore, this approach omits any object the robot is not trained on, i.e. objects from which the robot has not collected any training data.

The integration of robots into household environments has however made the recognition task more challenging. A big challenge is the great number of objects the robot should recognize and the environment dynamics. This raises a problem when training a robot on recognizing daily-life objects using the above-mentioned approach. On one hand, it is intractable to collect all possible objects to train the robot on them during a preliminary training phase. First, because of the big number of objects (30.000 classes as estimated by [11]). Second, because tactile data collection requires robot-object interaction, which is time consuming, especially because some sensors need a stable contact with the object surface to obtain good-quality measures. On the other hand, new objects are continuously added with time, and exploring each new object for training data collection hinders robot operation. Therefore, recognition systems based on multi-class classifiers, that omit objects the robot is not trained on, are not adaptable for recognition in daily-life.

Zero-Shot Learning (ZSL) is a learning paradigm that solves the problem of the unavailability of training data for some objects the robot should recognize. It consists in generalizing the models learned from training objects to recognize completely novel ones the robot is encountering for the first time during the test phase. The key idea is to mine relationships existing between training and novel objects, which will guide the use of

training data for recognizing the novel objects. This chapter reviews the different solutions proposed for ZSL, using both visual and haptic data.

A complementary problem to ZSL is the so called One-Shot Learning (OSL). OSL refers to the recognition of objects from which the robot previously collected only one or a handful of training data. It is known that multi-class classifiers require a fair amount of training data for each object class. Thus, they underfit object classes having very few training data which can go to one training sample only. In addition to ZSL, this chapter reviews the different studies carried out to solve the OSL problem using both visual and haptic data.

3.2 Zero-shot Learning

3.2.1 Definition

Zero-Shot Learning (ZSL) is the problem of training and testing a recognition system on two disjoint sets. Let

$$Y = \{y_1, \ldots, y_N\}, \tag{3.1}$$

be the set of objects the robot has been trained on, ensuing a training set

$$D_{train} = \{(y_n, (x_1, \ldots, x_{In}))\}_{n=1}^{N}, \tag{3.2}$$

where collected sensory raw data x_i are represented in the data space X. During the test phase, the robot collects $x_{test} \in X$ by exploring an unknown object that should be classified as one of novel objects

$$Z = \{z_1, \ldots, z_L\}. \tag{3.3}$$

Given $Y \cap Z = \emptyset$, the robot has no training data for any of Z objects. Their recognition can be made possible by acquiring an auxiliary information that relates them with Y object. This allows to classify x_{test} as one of Z objects based on sensory training data collected from Y objects.

Different types of auxiliary/side information can be used to describe objects. Acquiring descriptions of novel objects can relate them to known ones, which allows the use of training data to recognize novel objects. We present the three most popular means to relate novel objects to known ones: attributes, textual description and mining objects relationships.

3.2.2 Attributes

Attributes are properties that represent/describe an object. These properties can be *semantic* or *discriminative*. On one hand, *semantic attributes* are high-level properties that describe the object and can be understood by humans. They can describe multiple aspects of the object such as its geometrical properties, visual properties and tactile properties.

Geometrical properties of objects can be described using shape (e.g. round, concave, thin), volume (e.g. big, small) and part (e.g. neck, handle, tail) attributes. Visual attributes can describe color properties (e.g. white, multi-colored, dark, glittery) and specific patterns on the surface (e.g. stripes, spots, patches). Tactile attributes can describe object texture (e.g. soft, rough, furry), thermal properties (e.g. cold, hot, warm), material properties (e.g. glass, wooden, plastic) and compliance properties (e.g. bumpy, squishy, hard). Semantic attributes can also describe more general properties such as mobility (e.g. mobile, fixed, fast), category (e.g. clothes, utensils, food) and use (e.g. containers, tools, sport equipment). On the other hand, *discriminative attributes* are automatically designed attributes that do not necessarily have a semantic meaning. They are not used to describe objects but to distinguish between them. One can see them as separators between two sets of classes, e.g. a discriminative attribute can be the separator between the two sets of classes $\{$mug, cup, casserole$\}$ and $\{$bottle, jar$\}$, which means that mugs, cups and casseroles have this attribute but bottles and jars do not, or the opposite. Semantic and discriminative attributes can be used complementarily as in [30], such that semantic attributes describe objects and discriminative attributes separate between them.

Semantic attributes The set of semantic attributes used to describe objects should be carefully defined to best describe them. Thus, attributes should be chosen depending on what they describe. For instance, to describe animals [85] used attributes such as *tail*, *horns* and *meat-teeth* which are related to animals. On the other hand, to describe scenes [117] used attributes such as *ocean*, *natural light* and *open area*, which are properties that we usually use to describe scenes. Defining robust attributes is not an easy task, according to [127], good-quality attributes should discriminate well between objects, cover all objects, and be related to their visual observable properties (in case of visual recognition). Multiple methods were proposed in order to obtain such robust attributes. First of all, since semantic attributes can be understood by humans, then human participants can define attributes used to describe a set of classes. For instance, [113] asked 10 students to provide properties (attributes) that describe 48 animals, which ensued a list of eighty-five attributes that were used to build the Animals with Attributes dataset in [85]. More recent studies used crowd-sourcing in order to involve more participants, which is more efficient for obtaining good-quality and robust attributes for large-scale settings. For instance, [117] defined attributes used to describe scenes from textural descriptions provided by Amazon Mechanical Turk participants. In order to minimize the human labelling effort, another approach for defining semantic attributes is to mine them automatically from other sources of knowledge. For instance, [127] used the explicit *part relation* between WordNet nodes in order to define attributes describing a set of classes. They considered all parts of the studied object classes as their attributes, which ensued 74 mined attributes. Also, they proposed to use *Objectness* as attributes, which means that each object class represents an attribute. Then, they used this object/attribute to describe other objects according to the similarity of this object/attributes with the described objects.

Describing objects using semantic attributes requires the association of a numerical

value with each attribute-object pair. This numerical value can be binary, as in [85], indicating if the attribute describes the object or not. This value can also be real, as in [116], which determines the strength of the association between the attribute and the object. On one hand, binary attributes are easier to obtain, especially when dealing with large-scale datasets. On the other hand, real valued attributes (called relative attributes) describe objects more precisely. However, most studies use binary attributes since they are easy to learn, but still efficient for performing ZSL. Attribute-based description of objects can be obtained using multiple methods. They can be defined by humans as in [129], three AMT participants were asked to give a binary value for each attribute-image pair. Then, a positive value is given for the attribute only if all the three participants give it a positive value. Besides, attributes can be mined automatically as in [127], the presence of a direct or recursive "part relation" between an object and an attribute in WordNet architecture is used to determine the object-attribute binary association. In [127], semantic relatedness between objects is used to determine the association between the object and attribute.

Discriminative attributes They do not have necessarily a semantic meaning. In this case, they cannot be understood or defined by humans. Thus, they are automatically designed from training classes. In [30], tens of thousands of discriminative attributes are defined by randomly creating splits of 2 to 10 classes, then learning these splits using linear SVM and keeping only splits that are well predicted using validation data. The authors of [161], automatically designed non-semantic data-driven discriminative attributes from data available for training classes. They define three properties that should be considered while designing discriminative attributes: these latter should (1) increase the separability between classes, (2) be shared between classes to enhance their learnability, and (3) be non-redundant to avoid unnecessary attributes. On the other hand, discriminative attributes can have a semantic meaning as in[117]. First, the authors use AMT to acquire textual descriptions of classes, from which they extract a set of attributes. Then they present splits of classes to AMT participants and ask them to select the attributes that distinguish between classes from the two sides.

Attribute datasets Many studies are carried out on designing and building datasets that describe classes using attributes. Here is a non-exhaustive list of image attribute datasets and the only available haptic attribute dataset:

- **aPascal and aYahoo [30]:** they are composed of two disjoint sets of classes, including animals, transport vehicles and household objects. Images for the aPascal 20 classes are collected from Pascal VOC 2008 dataset [28], and images for the aYahoo 12 classes are collected from the Yahoo image search engine. Both datasets are described using the same set of 64 semantic attributes and was annotated using AMT.

- **Animals with Attributes [85]:** For all 50 animal classes defined by [113, 72], Lampert et al. [85] collected images from image search engines: Google, Mi-

crosoft, Yahoo and Flickr by using animal names as queries. This dataset describes these animals using 85 semantic attributes[113].

- **SUN attribute dataset [117]:** describes 717 scenes classes using 102 attributes using AMT. Images are obtained from the SUN dataset [158].

- **The Caltech-UCSD Birds-200-2011 [149]:** describes 200 bird species with 312 attributes using AMT. Images for each class are collected by querying the image search engine Flickr and filtered using AMT.

- **PHAC-2 [19]:** at the current time, this is the only existing attribute dataset that provides haptic data for object classes. This dataset describes 60 objects having a variety of texture, compliance and material properties using a list of 25 haptic binary attributes. This dataset is exploited in multiple studies to improve robot haptic perception using the provided *haptic adjectives* describing the objects. Gao et al. [38] use deep learning for recognizing PHAC-2 adjectives from haptic and visual data. The authors of [94] improve haptic perception by making use of adjectives correlations and learned adjectives in a multi-label setting instead of separately as in [38].

3.2.3 Textual description

Another way to acquire auxiliary/side information about novel classes is their textual descriptions. On one hand, humans have the ability to express what they are seeing/touching using sentences. This is exploited in [122], where AMT is used to provide textual descriptions of the visual appearance of images in at least 10 words. On the other hand, many resources, such as Wikipedia, provide rich textual descriptions of a wide variety of classes. Authors of [27] automatically extracted Wikipedia articles for each of the Caltech-UCSD Birds 200 dataset [152] classes. They also gathered textual descriptions for each of Oxford Flower-102 dataset [111] classes from Wikipedia, Plant Database, Plant Encyclopedia, and BBC articles.

3.2.4 Mining relationships

Instead of describing novel objects using text, one can describe them using known objects. This requires to define a similarity/relatedness measure between each pair of objects. Thus, novel objects can be classified according to their similarities to known objects. Multiple semantic relatedness measures between objects are proposed and compared in [127]. They use multiple linguistic knowledge bases such as WordNet and Wikipedia. They also query Yahoo search engine and Yahoo image search engine using object class names, and measure the semantic relatedness according to the returned results.

3.3 Visual Zero-Shot Learning

Although many broad image datasets are available for learning object recognition (such as ImageNet [25]), image labeling for all possible classes is still intractable. This justifies the great attention gained by ZSL in visual recognition. This section presents the state of the art on visual ZSL.

3.3.1 Attribute-based approach

The solution proposed by [85], illustrated in Fig. 3.1 and largely used by later studies, consists of defining a set of attributes $A = \{a_1, \ldots, a_M\}$ describing objects. Then, each object $o \in Y \cup Z$ is described using A. This associates o with a deterministic vector

$$\boldsymbol{a}^o = [a_1^o, \ldots, a_M^o], \tag{3.4}$$

where for $m = 1, \ldots, M$: $a_m^o = 1$ if attribute a_m is a property present in object o (e.g. $a_m = stripes$ for $o = zebra$) and $a_m^o = 0$ otherwise. Authors of [85] propose two models for using the attributes layer to transfer training data from Y to recognize objects in Z. In [1], we chose the Direct Attributes Prediction (DAP) model and showed better performance [85].

The DAP model uses training data collected from Y to learn a classifier per attribute. During the training phase, for each attribute $a_m \in A$, a probabilistic binary classifier $f_m : X \longrightarrow [0, 1]$ is trained on:

$$D_{train}^m = \{(x_i, a_m^{y_n}),\ s.t.\ (x_i, y_n) \in D_{train}\}. \tag{3.5}$$

During the test phase, the test sample x_{test} is input to each trained f_m to return the posterior $f_m(x_{test}) = p(a_m \mid x_{test})$. For each novel object $z_l \in Z$, all attributes posteriors are used given its associated $\boldsymbol{a}^{z_l}$ to infer its posterior as follows:

$$p(z_l \mid x_{test}) = \frac{p(z_l)}{p(\boldsymbol{a}^{z_l})} \prod_{m=1}^{M} p(a_m^{z_l} \mid x_{test}), \tag{3.6}$$

By replacing object and attribute priors with a uniform distribution, the test sample x_{test} is classified as the object having the highest posterior:

$$z_{test} = \underset{z_l \in Z}{\operatorname{argmax}}\, p(z_l \mid x_{test}). \tag{3.7}$$

A series of works were carried out on developing and improving ZSL based on attributes. The authors of [127] compare multiple techniques for mining attributes and class-attribute associations using multiple linguistic knowledge bases. The authors of [129] study attribute learning in large-scale datasets. The authors of [116, 17] study the generalization to real-valued attributes. Kankuekul et al. [70] handle attributes inconsistency when learned incrementally from different persons. To minimize human attribute

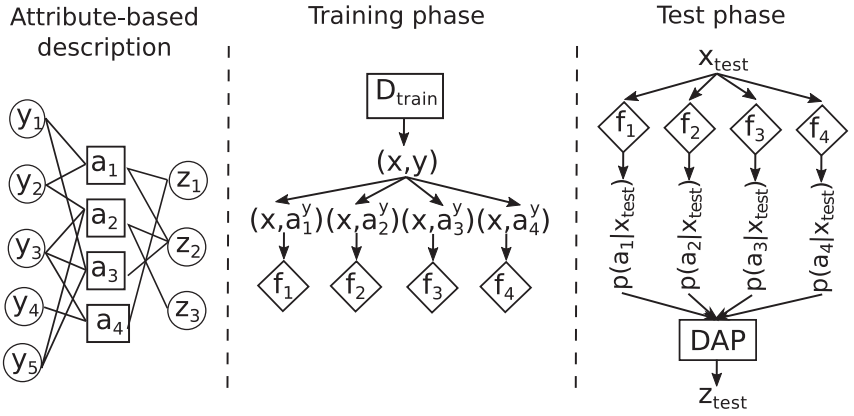


Figure 3.1: Attribute-based ZSL (solution overview for N=5,M=4 and L=3): First, both Y and Z objects are described using A attributes. Then, a classifier f_m is learned for each attribute. Last, attributes classifiers are used by the DAP model to infer the object class.

definition and labeling effort, the authors of [161] propose an automatic method to design non-semantic data-driven attributes. The authors of [59] consider attributes unreliability and propose a statistic solution to leverage errors in attributes. In [128], the two stages of learning attributes and inference are replaced by a simplistic implementation. The authors of [107] generate visual exemplars for unseen classes but using conditional variational autoencoders that maps attributes into visual examples.

The aforementioned studies evaluate their proposed algorithms on a wide variety of attributes datasets. Each database includes a set of classes sharing the same properties such as animals [85], birds [153], objects [30] or human faces [84]. According to the nature of classes, a set of semantic attributes are used to describe all classes of each dataset. Note that ZSL is usually performed on a set of classes that share the same properties which are in this case represented by attributes. Thus, generalizing models learned between very different sets of classes, for example training on birds and recognizing novel human faces, is still an open issue.

3.3.2 Direct and hierarchical similarity

The similarity between the novel and the known object classes can be used to directly classify a novel class according to the classification results of similar known classes. The authors of [127] express the posterior probability of a novel class given a test sample as the product of posterior probabilities of all similar known classes given the same test sample. The authors of [126] study ZSL in large-scale settings. They propose multiple methods for classifying novel classes according to their relationships with known ones: they propose a hierarchy-based approach that computes the classification score of a novel class using the classification scores of its neighbor classes in the WordNet structure. Also, they propose a direct similarity-based approach that computes the classification score of

a novel class as the average of the classification scores of K semantically related classes. They improve both methods using semantic relatedness measures.

3.3.3 Two-steps classification

This approach performs ZSL by mapping sensory data into an embedding space that captures semantic relatedness between classes, then classifying mapped data in the embedding space. The authors of [115] learn two mappings, one from raw sensory data into a semantic space, and the other from the semantic space to the class label space. First, they use multiple output linear regression to represent raw data by semantic codes. Then, they use 1-nearest neighbor in the semantic space to classify the mapped data. In [140], the authors learn a semantic word space from text corpora in an unsupervised manner, then a mapping from visual to this semantic word space. In addition to novel classes, they include training objects in the test set. Thus, to recognize a data sample, they first use a novelty detection metric that predicts if it is known or novel. If it is known, it is classified by a multi-class classifier to one of the training classes. Otherwise, the data sample is mapped to the semantic word space, and its likelihood of being each novel class is computed by assuming an isometric Gaussian distribution around the word vector representing the class in the semantic word space.

3.3.4 Predicting one-vs-all classifiers for novel classes

This approach learns a one-vs-all classifier for each novel class. Since there is no data for novel classes, the side information available for the novel class is used to predict the parameters of its associates one-vs-all classifier. An example is the work of [27] which transforms ZSL into a regression problem. First, the authors learn a one-vs-all binary linear classifier for each known class given the available training data. Then, they learn a regressor that estimates the parameters of the known class classifiers given their textual descriptions. Finally, a domain transfer function from textual to visual domains is learned. In [103], co-occurrence statistics of visual concepts between images are used to compute the similarity between each known and novel class. Then, a binary SVM is associated to each novel class, such that its weight vector is the linear combination of weight vectors of known classes' SVMs. Finally, the authors of [90] predict the weights of CNNs associated to novel classes directly from their textual descriptions.

3.3.5 Generating synthetic training data

This approach copes with the absence of real training data for novel objects by generating synthetic ones. Then, novel classes can be recognized using traditional multi-class classifiers trained on the synthetic training data. In [16], a prototypical visual sample is predicted for each novel class using learned kernel-based regressors from semantic space into visual space. Then, recognition is performed using nearest neighbor classification in the visual space. The authors of [150] propose a two-stages relational knowledge transfer

for ZSL. They generate for each unseen class a set of synthesized instances by estimating a Gaussian distribution for each class in the feature space.

3.3.6 Domain Shift Problem

In ZSL, sensory data collected from training classes are used to recognize novel classes given some sort of side information. However, data distribution between training and novel classes can be very different. Thus, the mapping usually learned from sensory data to the semantic space from training classes may not generalize well to novel classes, which is known as the *domain shift problem*. This problem is studied by multiple works. For instance, authors of [35] improve the generalization capability by using unlabeled data available for novel classes. In addition, the authors of [36] use an absorbing Markov chain process for estimating the similarity in the semantic space based on the semantic manifold structure.

3.4 Haptic Zero-Shot Learning

Tactile recognition systems suffer not only from the difficulty of labeling data, but also from the difficulty of data collection. Tactile data collection requires robot-object interaction, which is time consuming, especially because some sensors need a stable contact with the object surface to obtain good-quality measures, e.g. [74] maintained the contact with the object for 20 seconds. Nevertheless, visual ZSL has gained much more research attention than haptic ZSL, which motivated us to propose the first Zero-Shot haptic (tactile and shape) recognition system in [1], which is presented in chapter 4. Then, we improve this framework by adding training classes into the test set in [3], which is presented in chapter 6.

3.5 Visuo-haptic Zero-Shot Learning

As for haptic ZSL, combining haptic/tactile and visual data for performing ZSL has gained little attention. In a recent study, the authors of [93] proposed a visuo-tactile dictionary learning for ZSL of the eight material categories grouping PHAC-2 objects. Since visual data are much easier to obtain than tactile data, they perform tactile ZSL by assuming the availability of visual data for novel objects. Results show that incorporating both visual and tactile modalities is effective for performing ZSL. To cope with the absence of a ZSL system capable of recognizing novel objects with the absence of both tactile and visual data, we propose a visuo-tactile ZSL framework [2], which is presented in chapter. 5.

3.6 One-Shot Learning

ZSL solves the problem of absence of training data for some classes, that are usually omitted by the recognition system during the test phase. Thus, an extension of ZSL is the advent of one or very few training samples for novel classes, which is known as One-Shot Learning (OSL). The recognition system usually under-fits these classes because of the very few number of samples compared to other classes having a fair amount of training data. Few works extend their ZSL to OSL, an example of the importance of integrating both problems is [162], the authors apply topic modelling to attributes to recognize objects having zero or one training sample per each.

3.6.1 Definition

One-Shot Learning (OSL) solves the problem of recognizing under-represented classes in the training set. The number of training samples for each class during OSL can be as low as just one sample per class. These classes are under-fitted in such scenarios because the classifier cannot learn robust models using this sparse amount of training data. Rather than using the one available sample for each class, OSL exploits the knowledge acquired from previously known classes, for which a fair amount of training data is available. This process is inspired by the ability of generalization exhibited by humans. The next section presents the state of the art on OSL.

3.6.2 Visual One-Shot Learning

The authors of [31, 32] recognize objects using only one to five training samples per class. They adopt a Bayesian approach by representing each object class by a probabilistic model. The fair amount of training data available for some objects is used to estimate the prior of one-shot classes. Then, the tiny amount of data available for each one-shot class is used to update the prior and to obtain the posterior model of this class. In [155], learning from few training data is performed by augmenting the training set by adding corrupted copies of original data. A hierarchical Bayesian approach is adopted in [114] such that relationships between features are exploited to estimate Bayesian classifier parameters. The proposed algorithm is able to efficiently classify very high-dimensional data with only 2 training samples per class. In [33], OSL of 101 object classes is performed using a generative probabilistic model. OSL is made fast by adopting an incremental approach that incorporates prior information learned from unrelated objects. In [125], the relevance of features, learned from classification tasks for which plenty training data are available, is used as a prior for recognizing one-shot classes. In [147], one-shot classes are classified using LS-SVM. Another hierarchical Bayesian model is used in [132], the OSL is performed by estimating the mean and variance of each object class. The authors of [156] use distance metric to extract samples, from plenty training samples available for known classes, that are the closest to the few samples available for the one-shot class. Then, they use them to learn the model of the one-shot class. In [55]; the plenty training

data for known classes are used to improve the recognition of one-shot classes. In order to avoid transferring irrelevant/unrelated information, abundant training data are reconstructed by the one sample available for the one-shot class. In [139], a simple but efficient OSL is proposed based on prototypical networks. The authors learn a metric space where each object is represented by the mean of its training data, and a sample is classified to the class corresponding to the closest prototypical point. Furthermore, OSL is exploited to performed many other tasks such as gesture recognition [163] and action recognition [60].

3.6.3 Haptic One-Shot Learning

Few works tackle OSL problem from haptic/tactile data. Kaboli et al. [67, 68] perform texture-based OSL using LS-SVM. Results show the ability of the proposed solution to recognize twelve totally novel textures having one or few training samples, by training a Shadow Hand equipped with BioTac tactile sensors on a set of ten objects.

3.7 Conclusion

This chapter reviews the state of the art on object recognition with the lack of training data for novel object classes. First, ZSL is presented, which handles the recognition of novel objects having no training data. Followed by the OSL, which handles the recognition of objects having one or a handful of training data. Despite the promising results found by the different studies, there is much more scope to improve ZSL and OSL. First, the state of the art on visual Zero-Shot and One-Shot Learning is very rich, and very good accuracies have been reached. However, there is no previous study on tactile, and more generally haptic, ZSL. Coping with this lack is very important since haptic data collection is effort and time consuming. This motivated us to propose a haptic ZSL system in chapter 4, which will be presented in the next chapter. Second, motivated by the success of recognition systems combining both haptic and visual data, as presented in Sect. 2.5, we propose our visuo-tactile ZSL framework, which is presented in chapter 5. Finally, studies on haptic OSL are very few, and even for the available visual OSL systems, they are not often combined with ZSL although their complementarity for recognizing daily-life objects. To improve this point, we propose a second tactile recognition system, capable of performing at the same time ZSL, OSL and multi-class classification, according to the quantity of training data available for each object class. This will be presented in chapter 6.

Chapter 4

Haptic Recognition of Objects Never Touched Before

4.1 Introduction

Robots operating in household environments should be capable of using the prior knowledge acquired from previously experienced objects for learning novel ones. Recognition of such *novel objects* can be achieved with Zero-Shot Learning (ZSL). This chapter presents our proposed solution for performing ZSL based on haptic data. Among the different approaches allowing to perform ZSL, presented in Sect. 3.2, we choose the attribute-based approach, presented in Sect. 3.3.1.

Since haptic recognition algorithms depend on the used experimental setup, we study the application of attribute-based ZSL on two different experimental setups. First, we use the extensive PHAC-2 database [19] and our own robotic setup to adapt, analyze and optimize the ZSL for the challenges and constraints introduced by haptic recognition. Then, we apply the optimized ZSL for haptic recognition of daily-life objects using an anthropomorphic robot hand. Our algorithm enabled the robot to recognize eight of the ten novel objects handed to it.

This chapter is organized as follows: Sect. 4.2 presents the application of ZSL to haptic data using the PHAC-2 database. Then, Sect. 4.3 evaluates the influence of different criteria on the recognition performance. Next, Sect. 4.4 details our proposed solutions to adapt the theoretical framework to object recognition with a robot hand, and the experimental evaluation is presented in Sect. 4.5. Finally, Sect 4.6 provides conclusions.

4.2 Haptic Zero-Shot Learning on PHAC-2

The adaptation of the visual attribute-based ZSL presented in Sect. 3.3.1 to haptic data collected by a robot is not trivial. The specific nature of haptic data adds challenges that require various adaptations of the aforementioned theoretical framework:

- Haptic data collection is costly, requiring the optimization of training data;

- Both the choice of attributes and the nature of haptic data depend heavily on a robot's sensing capabilities, hindering comparisons and generic solutions;
- Data gathered from the robot can be multimodal (e.g. joints, temperature...) requiring processing and fusion technique;
- The spatial limitation of robot exploration leads to sparse and noisy, or missing data, which is challenging to the recognition algorithm.

To assess the use of ZSL on haptics in the presence of all these difficulties, we start by applying it to the state of art PHAC-2 dataset [19].

4.2.1 PHAC-2 database description

The PHAC-2 dataset is designed to study how a robot can learn to describe its haptic perception using words. The developers of PHAC-2 use *haptic adjectives* which are binary high-level physical properties of objects such as 'hard', 'absorbent' and 'bumpy'. By considering the PHAC-2 adjectives as attributes, we can apply an attribute-based approach to perform ZSL.

Several characteristics motivated us to use PHAC-2 for haptic ZSL. First, it is probably the single largest object database with objects labeled by their haptic characteristics [23, 48, 123]. Second, the objects in the database encompass a variety of physical properties in terms of texture, material and stiffness. Third, the adjective definition and the binary associations between objects and adjectives have been developed using an arguably unbiased procedure. And finally, PHAC-2 provides data from multiple explorations of the same object, which tests the robustness to inter-trial variabilities.

The PHAC-2 database contains 60 objects, labeled by a broad range of material, texture and stiffness related adjectives. These adjectives are defined by human participants, who blindly explored objects using their hands, and expressed their sensations using words. For our analysis, we utilize 19 adjectives (after removing adjectives present in less than 3 objects): $A = \{absorbent, bumpy, compressible, cool, fuzzy, hard, hairy, metallic, porous, rough, scratchy, slippery, smooth, soft, solid, springy, squishy, textured, thick\}$.

These attributes were defined by human participants, who blindly explored objects using their hands, and expressed their sensations using words. Defining good quality attributes is still an open research problem. Two main points must be considered: (1) Defining non-ambiguous attributes that describe well the objects and increase objects separability and (2) reducing the effort of attributes definition and class-attribute labeling. For instance, thanks to crowd-sourcing, humans can collaborate to describe voluminous object datasets using haptic attributes. Clearly, this takes much less effort than exploring all objects using the robot. Studies such as [161] focused on this problem. However, this is out of the scope of this thesis, we take attributes provided with PHAC-2 dataset.

In addition to the adjectives, the PHAC-2 database provides 48 raw haptic signals.

These have been obtained using the Willow Garage PR2 robot[1] gripper, equipped with two BioTacs sensors[2]. The gripper provides the following kinesthetic information: gripper aperture X_g and height Z_g in the robot torso coordinate frame. Additionally, each BioTac provides cutaneous information: core temperature T_{DC}, heat flow T_{AC}, static pressure P_{DC}, dynamic pressure P_{AC} and the voltage change for each of the 19 impedance sensing electrodes $[E_1 \dots E_{19}]$. These data items (X_g, Z_g and 23 signals from each BioTac) are stacked to define a 48-dimensional vector. Ten vectors are provided for each object, one corresponding to each trial. Each trial has four Exploration Procedures (EP), including a squeeze, hold, slow slide and fast slide on the object, making a total of 600 samples (60 objects $\times$ 10 trials).

4.2.2 Splitting the object set

We split the set of 60 objects contained in the PHAC-2 database, denoted O, into two disjoint sets Y (training set) and Z (test set). To ensure that the results are independent from the chosen splits, we generate 5 splits $\{(Y_s, Z_s),\ s = 1, \dots, 5\}$ where each pair respects two constraints: $Y_s \cup Z_s = O$ and $Y_s \cap Z_s = \emptyset$. To generate each pair (Y_s, Z_s), we randomly choose for the test set 10 objects out of the 60:

$$Z_s = \{o_k \in O\ ,\ k \in rand(10, 60)\}^3 \tag{4.1}$$

and the remaining 50 objects are taken for training $Y_s = O - Z_s$. However, some objects have the same attributes vector and since ZSL identifies each test object by its attributes vector, we verify that the attributes vectors of all objects in Z_s are mutually different. If not, the random selection (of the 10 objects) is repeated until this condition was satisfied.

4.2.3 Feature Extraction

Two types of features are considered during each EP: kinesthetic features and Biotac features. As in [19], kinesthetic features are composed of the following data relative to the gripper: the minimum aperture, the mean aperture and the displacement distance. All the kinesthetic features are then stacked into a vector having 12 components (3 features $\times$ 4 EP). In order to extract BioTac features, we exploit the results from [53, 159, 20]. First, we remove the baseline activity from the BioTac readings, by subtracting the mean of the first 100 readings. Next, the average of each signal is computed over the exploration time. The computed averages are then normalized to have zero mean and standard deviation of 1. Since the BioTacs are not calibrated in the dataset, we consider each BioTac separately. Thus, by concatenating features of the 4 EP, we obtain a vector of 92 features for each BioTac (23 BioTac features $\times$ 4 EP). Each vector is concatenated with the 12 kinesthetic features, resulting in a vector of 104 features for each BioTac. This high dimension is reduced using PCA to a 25-dimensional vector that justifies 95% of the variance. To sum

[1] `http://www.willowgarage.com/pages/pr2/overview`

[2] `https://www.syntouchinc.com/sensor-technology/`

[3] $rand(k, n)$ returns k random numbers in range $1, \dots, n$.

up, an exploration trial results in 2 vectors of cutaneous and kinesthetic extracted features $x_{b1}, x_{b2} \in \mathbb{R}^{25}$, one for each BioTac.

4.2.4 Attributes classification

Attributes classification aims at estimating the presence of attribute a_m for a given feature vector x, i.e. to derive $p\left(a_m|x\right)$. By assuming attributes independence, we train a SVM classifier for each attribute and each BioTac. When a test sample x is given to a SVM classifier, it returns a score $s\left(x\right)$ corresponding to the distance from x to the decision boundary. The attribute posterior $p\left(a_m|x\right)$ is estimated by transforming $s(x)$ into a probability using a sigmoid function.

We train a binary classifier for each attribute a_m and BioTac b_i using the training set

$$D^m_{train,b_i} = \{(x_{b_i}, a^y_m)\ s.t.\ (x, y) \in D_{train}\}. \tag{4.2}$$

First, by analyzing the matrix associating all object-attribute pairs with binary labels, provided by [19], we notice that for each attribute the number of objects in which it is present is significantly smaller than the number of objects from which it is absent. This leads to an imbalanced training set D^m_{train,b_i}: the ratio of class 1 samples to class 0 samples is $a : b$ with $a << b$. An imbalanced training set can lead to over-fitting of the over-represented class. To cope with this problem, we under-sample class 0 by randomly removing pairs $(x_{b_i}, a^{y_k}_m)$ having $a^{y_k}_m = 0$ from D^m_{train,b_i} until we obtain an equal number of training samples for classes 0 and 1. This under-sampling is repeated multiple times, and sets giving the best results for attributes learning are used. Using the balanced set, we train a non-linear SVM classifier with a Gaussian kernel. SVM parameters C and γ are tuned using leave-one-out cross-validation, by varying C between 1 and 10^2 and γ between 10^{-2} and 1.

4.2.5 BioTacs combination

Since we separately extract features from each one of the two BioTacs: $x = (x_{b1}, x_{b2})$, attributes posteriors $p(a^{z_l}_m|x_{b1})$ and $p(a^{z_l}_m|x_{b2})$ must be combined to infer $p(a^{z_l}_m|x)$. For this, we propose two methods. The first (that we name MAXDAP) considers only the highest posterior, by assuming that it is the most confident value between the two:

$$p(a^{z_l}_m|x) = \max(p(a^{z_l}_m|x_{b1}), p(a^{z_l}_m|x_{b2})). \tag{4.3}$$

The second method (AVGDAP) considers the average of both posteriors to eliminate the influence of individual BioTacs misclassification and uncertainty:

$$p(a^{z_l}_m|x) = \frac{p(a^{z_l}_m|x_{b1}) + p(a^{z_l}_m|x_{b2})}{2}. \tag{4.4}$$

4.2.6 Attributes priors

Inferring novel objects posteriors according to equation (3.6) requires to estimate the prior probability for each attribute. We compare three methods for measuring the attribute prior $p(a_m^{z_l})$:

1. based on its presence in training objects: $p(a_m = 1) = \frac{1}{N} \sum_{n=1}^{N} a_m^{y_n}$;
2. based on its presence in test objects: $p(a_m = 1) = \frac{1}{L} \sum_{l=1}^{L} a_m^{z_l}$;
3. based on a uniform distribution: $p(a_m = 1) = 0.5$.

We refer to these methods as *train-prior*, *test-prior* and *uni-prior* respectively. Note that test-prior is feasible because the attribute vectors of novel objects are available to the robot during the test phase.

Finally, we combine the attribute posteriors and priors to infer the posterior of each novel objects in Z according to equation (3.6). Test sample x is classified according to equation (3.7) using the maximum a posteriori estimator.

4.3 Evaluation and discussion

The performance of a recognition algorithm may be evaluated by its classification accuracy, computation speed and memory requirement. Here we concentrate on classification accuracy as we are using an offline recognition method and are not constrained by the training response time. Furthermore, we suppose that enough resources are available to run our algorithms.

4.3.1 Implementation

The presented method is implemented using Python[4], and our machine learning algorithms are implemented using the Python library *Scikit-learn*[5]. Our experiments were carried out successfully on a PC with an Intel(R) Core(TM) i7-3840QM processor having a speed of 2.8 GHz and a RAM of 8 GB.

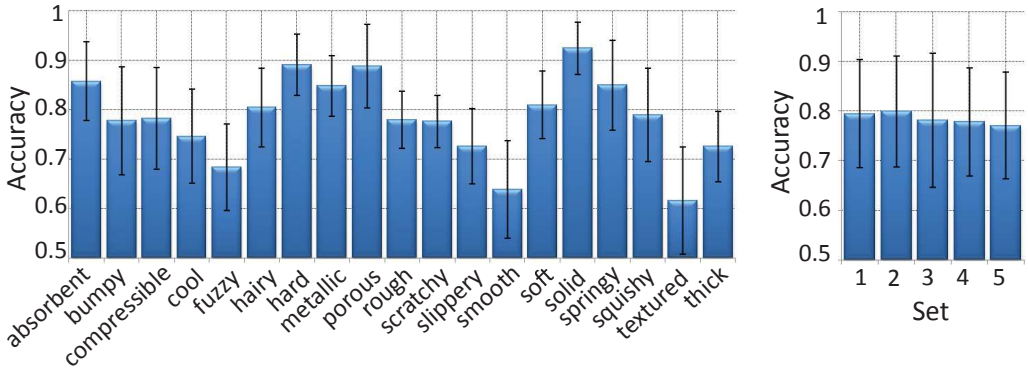
4.3.2 Attributes classification

Fig. 4.1a shows the average classification accuracy for each attribute. We average the accuracies obtained from the two BioTacs classifiers that are tested on the 5 different test object sets. The results show that attributes classifiers are able to learn from training data whether an attribute is present or absent in a novel object. However, performance varies from one attribute to the other: we obtain an accuracy of 92% for attribute *solid* whereas we obtain modest, though still better than random, performance for attributes

[4]based on `https://github.com/IanTheEngineer/Penn-haptics-bolt`

[5]`http://scikit-learn.org/stable/`

smooth (64%) and *textured* (64%). To evaluate the influence of the object splits, we plot in Fig. 4.1b the attributes classification accuracies for each of the five splits (Sect. 4.2.2), averaged on the 2 BioTacs and on all attributes. The figure shows that classification accuracies are not heavily influenced by the choice of object splits: the average accuracy is 78%, with a maximum difference of only 3% between splits 2 and 5.



(a) (b)

Figure 4.1: Recognition accuracies and standard deviations of attributes binary classifiers.

4.3.3 Recognition of novel objects

First, we compare in table 4.1 the use of MAXDAP and AVGDAP to combine BioTac posteriors by assuming uniform attributes posteriors. Since we use the MAP estimator in equation (3.7) to distinguish the ten test objects, the random classification accuracy for the ZSL algorithm is 0.1. The reported accuracies are therefore significantly higher than random chance. Performances of the two methods are similar, since the objects have homogeneous properties on their surfaces, giving close attributes posteriors for the two BioTacs.

Split	MAXDAP	AVGDAP
1	0.48	0.48
2	0.38	0.36
3	0.38	0.38
4	0.35	0.35
5	0.35	0.36
Average	0.39	0.39

Table 4.1: ZSL accuracy for independent attributes classifiers.

We report in table 4.2, comparison results between different methods proposed to measure attributes priors presented in Sect. 4.2.6. The reported results are obtained using

AVGDAP method. The table last line shows that the three methods give similar accuracies with a slight improvement with *uni-prior*. This is coherent with results reported in [85], which show that attribute prior estimation is not crucial.

Split	uni-prior	test-prior	train-prior
1	0.48	0.24	0.36
2	0.38	0.33	0.28
3	0.38	0.39	0.32
4	0.35	0.39	0.38
5	0.35	0.33	0.29
Average	0.39	0.34	0.33

Table 4.2: DAP accuracy for independent attributes classifiers.

While above we report results assuming attribute independence, we did also consider attribute classification without the assumption of independence. For this, we trained a multi-label SVM classifier for each BioTac b_i using $D_{train,b_i} = \{(x_{b_i}, \boldsymbol{a}^y)$ s.t. $(x, y) \in D_{train}\}$, with the Python *scikit-multilearn*[6] library implementing the classifier proposed in [121], and tuned the SVM parameters C and γ using a leave-one-out cross validation. This multi-label classifier predicts, given a test sample x, a vector of posteriors, one for each attribute. However, with this multi-label classifier, we obtained an average recognition accuracy of 0.31, which is lower than that (0.39, see table 4.1) obtained by assuming independent attributes. This decrease in performance is probably due to the fact that the multi-label classifier is sensitive to the imbalance in the data size available for each attribute, while we solved this problem for independent classification in Sect. 4.2.4.

4.3.4 Robustness to the choice of objects

Previously, we reported results for five random training/test splits. In order to make the recognition more challenging for our ZSL algorithm, we try to minimize the similarities between the training and test data. In addition, we maximize them between test data. For this aim, we consider our objects in a 2-dimensional representation of haptic similarities (Fig. 4.2). To obtain this representation, we use PCA to reduce the dimension of each trial feature vector from $\mathbb{R}^{25}$ to $\mathbb{R}^2$ (by considering the first two principal components to represent the objects). Following this, we average the 10 vectors corresponding to the 10 trials of each object so as to obtain one vector that represents the object in the 2-dimensional space.

In this representation space, note that near objects feel similar when touched, and are harder to distinguish by a haptic recognition system. Therefore, to create challenging splits, we first use lines $y = x$ and $y = -x$ (continuous) to divide the object space into 4 partitions. Since the partition on the left of the graph is significantly denser than

[6] `http://scikit.ml/`

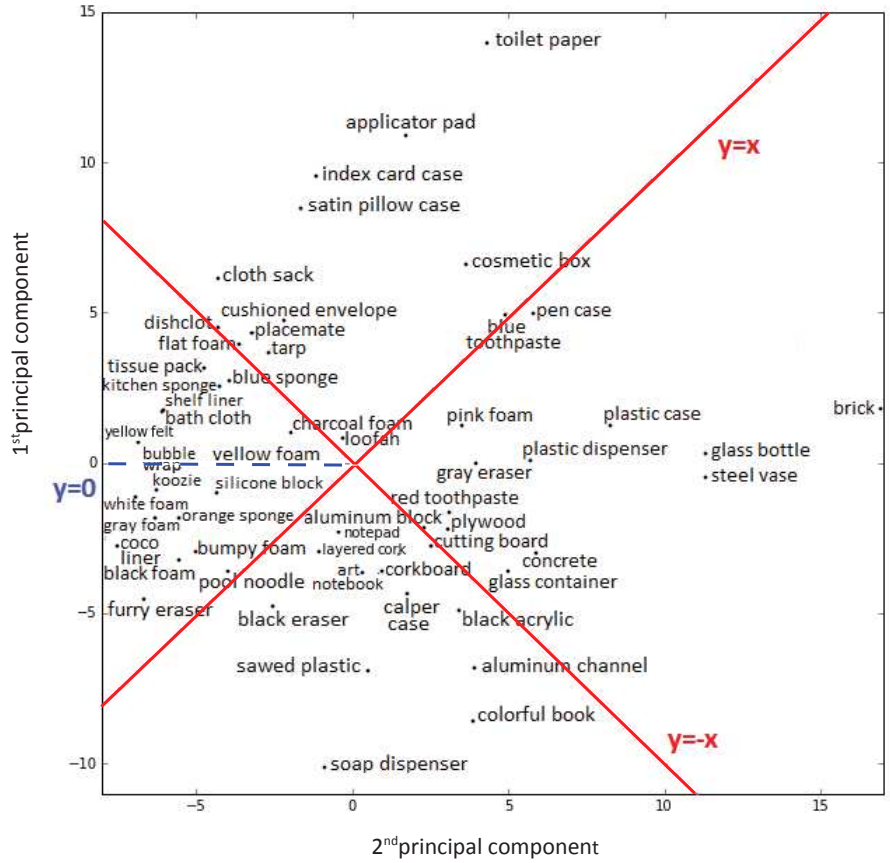


Figure 4.2: Splitting the PHAC-2 objects set into five partitions.

the others, we divide it again in two with line $y = 0$ (dashed), to yield 5 partitions in total. We then consider 5 splits, by using each time one partition as test set and the 4 others for training. Since attribute-based ZSL cannot distinguish between objects having identical attribute vectors, in each test set we keep only objects having different attributes vectors. Next, to increase the attributes classifiers' ability to generalize to new objects, we ensure that each attribute classifier was trained using at least 3 different objects. Overall, the above procedure creates challenging object splits that ensure that, a) test objects are *different* from training objects, and b), test objects are *similar*, making them harder to distinguish.

In table 4.3, we report the classification accuracies with the challenging splits. Overall, we note a decrease in accuracies, as compared to the results obtained using random splits. This was expected since we intentionally made the ZSL task more difficult. Although the obtained accuracies are significantly lower than what could be obtained if haptic data were available, the recognition accuracy is above chance, highlighting the ability of the algorithm to recognize objects very different from those it has trained on, without any additional data.

Split	number of objects	random accuracy	AVGDAP
1	11	0.09	0.18
2	7	0.14	0.21
3	7	0.14	0.3
4	10	0.1	0.22
5	11	0.09	0.27
Average			0.24

Table 4.3: DAP accuracy for challenging splits.

4.3.5 Influence of attributes number

Finally, we investigate the influence of the number of attributes on the ZSL accuracy. As mentioned before, we use 19 attributes to perform ZSL. To estimate the accuracy when using j attributes, we randomly pick up to 1000 combinations of j attributes out of the 19, and then average the accuracies of ZSL obtained with each of the combinations. We do so with j varying from 7 to 19. We start from 7 since it is the size of the minimum subset of attributes that allows to distinguish between objects. Results, reported in Fig. 4.3, show that, as expected, increasing the number of attributes generally improves the performance. In fact, increasing the number of attributes improves the separation between objects, which are now represented in a higher dimensional attribute space. Thus, objects have more distinct attribute vectors which alleviates the chances of misclassifying them.

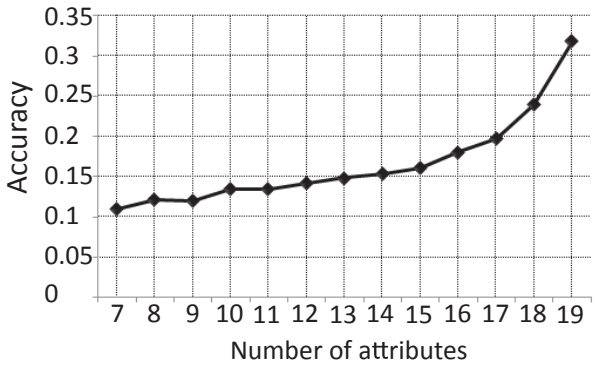


Figure 4.3: Recognition accuracy vs. the number of attributes utilized for ZSL.

4.3.6 Summary

To summarize, we developed a variant of ZSL for haptic recognition of novel objects. Using the PHAC-2 dataset, we analyzed the influence of several factors on the performance of the haptic ZSL, including the way of combining the attribute posterior, the choice of attribute prior, the object set split and the number of attributes. Furthermore, we showed the robustness of our algorithm by minimizing the similarity between the training and test

set. We showed that even in the worst case, when the training and test objects are very different, the algorithm can still give above random recognition accuracies. In the next section, we apply the developed haptic ZSL on an anthropomorphic hand system, to test its performance in a real experimental setting.

4.4 Zero-Shot Learning on daily life objects held by a robotic hand

4.4.1 Motivation: Beyond PHAC-2

PHAC-2 offers an important amount of objects, attributes and haptic data allowing us to test the application of ZSL on haptics. However, the database was built in a controlled setting and with objects of regular shape, though this may not always be the case in real robotic applications. Therefore, in our experimental setup we use a less controlled exploration, more realistic objects with heterogeneous surface properties, and different modalities of haptic data. The goal is to show how ZSL can be applied to real life robotic applications, and to handle new constraints the recognition system could face in such applications. Specifically:

1. We do not have a planned object exploration strategy (like in PHAC-2), and our object exploration is achieved via open-loop random grasps. Our robot grasps the object in an unknown position and orientation, and uses whatever it sensed as exploration data. This make the exploration short and coarse, but crucially more realistic. Human-inspired exploration approaches (e.g., active perception and motor babbling) while being more realistic than ours, are out of scope here.

2. We use daily-life objects with semantic meanings (e.g. bottle, mug and box). The objects are not of any particular shape. In addition, we allow objects to have heterogeneous physical properties on their surfaces, meaning that the object can feel different depending on the touched part. This requires the exploration to be incremental and to include different sources of information.

3. We use a dexterous anthropomorphic robotic hand that offered not only cutaneous information, but also rich kinesthetic data, typical of whole hand object grasps.

4. We make use of available (online dictionary based) textual descriptions of the object in order to avoid the time-consuming human exploration process and minimize the human effort needed in this procedure. This is more suitable for real scenarios, since we aim at minimizing the cost of adding a new object to set O.

4.4.2 System overview

In Fig. 4.4, we summarize the different steps of our robotic experimentation. First, we collect two disjoint sets of daily life objects, one for training and the other for testing.

Then, we define a list of attributes $\{a_1, \ldots, a_M\}$ allowing us to describe both sets of objects, resulting in the object-attribute matrix. Next, we train binary attribute classifiers: using features extracted from BioTac readings x_{mat} for material attributes and from robot joint readings x_{sh} for shape attributes. Finally, haptic readings collected by exploring a new test object are introduced to the attributes classifiers and the resulting posteriors $p(a_m \mid x_{sh})$ and $p(a_m \mid x_{mat})$ are used by the DAP model to infer the test object's class.

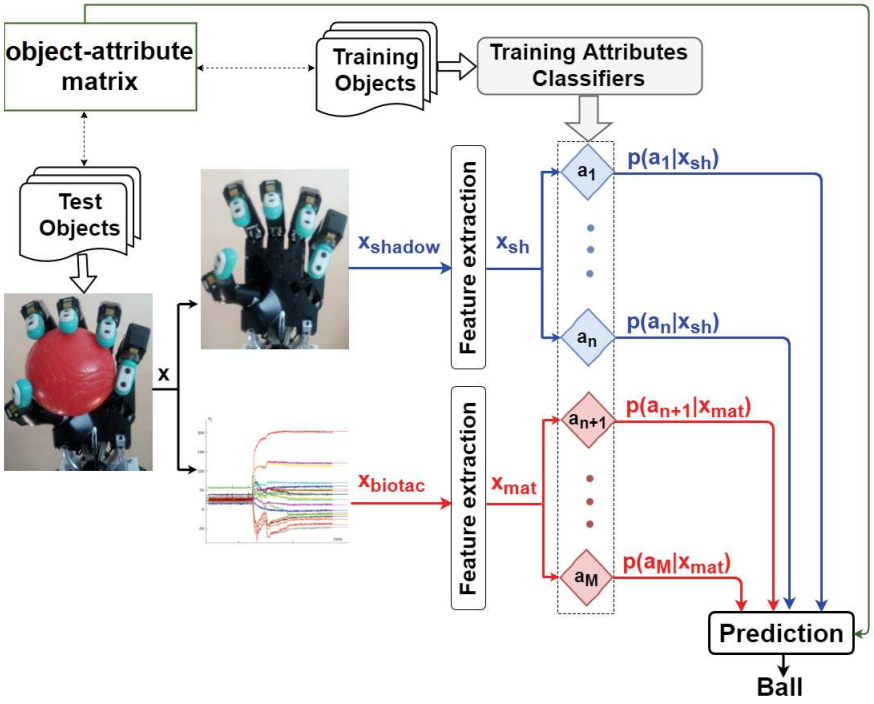


Figure 4.4: Our zero-shot recognition system overview.

4.4.3 Robot hand setup

Our robot setup consists of a cable-driven Shadow Dexterous Hand[7] with a BioTac sensor mounted on each fingertip. The encoders on the hand's 19 joints provide kinesthetic information $x_{shadow} = \{q_1, \ldots, q_{19}\}$, with q_i the angular position of joint i, and each BioTac provides cutaneous information $x_{biotac} = \{T_{AC}, T_{DC}, P_{DC}, P_{AC}, E_1, \ldots, E_{19}\}$.

4.4.4 Daily life object set

We collect a set of different daily life objects to form our object set O. Our choice is based on multiple state of art studies that established lists of real objects that are interesting for robotic manipulation. The authors of [18] propose a list of objects ordered according to their relevance for automatic retrieval after surveying people with amyotrophic lateral

[7] shadowrobot.com

sclerosis (ALS). A larger list of daily life objects categorized according to their use is given in [100]. The YCB database [14] also regroups a set of physical objects to be used for object manipulation benchmarking. We choose objects mentioned in these works, that are big enough to be grasped by the Shadow Hand, and which are not hot, nor with sharp edges that could damage the BioTacs. Finally, we select the 20 objects illustrated in Fig. 4.5 to form our object set.

Figure 4.5: Disjoint training (left) and test (right) object sets.

4.4.5 Attribute-based description

For attribute-based ZSL, the design of the dataset requires defining: (1) training objects Y, (2) test objects Z, (3) the set A of attributes that can be derived from the data collected with our setup and (4) matrix $\boldsymbol{K}$. First, we randomly split the object set O into two equal disjoint sets: $Y = \{$*cardboard box*, *glass bottle*, *plastic bottle*, *round container*, *mug*, *thermal mug*$\}$ and $Z = \{$*ball*, *rectangular container*, *tube*, *blender leg*, *bowl*, *glass*, *plastic cup*, *measuring cup*, *jar*, *salter*$\}$ as illustrated in Fig. 4.5. Next, we define a set of attributes A appropriate for describing the haptic sensation of Y and Z. Multiple works studied the definition of attributes by relying on human expressing capabilities [19, 113] or linguistic knowledge databases [129, 127]. In PHAC-2, authors rely on human experiments to define the adjectives list, which can depend on the participants' choice of words. Here, we choose a more objective approach by making use of objects names to extract their textual descriptions from online dictionaries[8]. From these descriptions, we choose attributes that can be sensed by our robot. For instance, from the definition of a bottle: *a glass or plastic container that has a narrow neck and usually has no handle*, we extract the underlined statements as attributes. Overall, by analyzing all objects descriptions, we extract a list of 11 shared attributes : $A = \{$*porcelain*, *plastic*, *glass*, *cardboard*, *steel*, *cylindrical*, *round*, *rectangular*, *concave*, *has a handle*, *has a narrow part*$\}$.

Next, we set $a_m^o = 1$ if the attribute a_m is used to describe the object o as a required property (e.g. *has a narrow part* for a bottle), and $a_m^o = 0$ if the attribute is an undesired or unnecessary property (e.g. *has a handle* for a bottle). Using this procedure, we obtain

[8] We used merriam-webster.com and en.oxforddictionaries.com.

the object-attribute matrix $\boldsymbol{K}$ illustrated in table 4.4. Training objects y_n having identical $\boldsymbol{a}^{y_n}$ are indicated only once in the table (this is why there are 6 instead of 10 training objects). Note that each test object z_l has a specific $\boldsymbol{a}^{z_l}$ that will be used to distinguish it from other objects during recognition.

	porcelain	plastic	cardboard	glass	steel	cylindrical	round	rectangular	concave	has handle	has narrow part
cardboard box	0	0	1	0	0	0	0	1	0	0	0
glass bottle	0	0	0	1	0	1	0	0	0	0	1
plastic bottle	0	1	0	0	0	1	0	1	0	0	1
round container	0	1	0	0	0	0	1	0	0	0	0
mug	1	0	0	0	0	1	0	0	1	1	0
thermal mug	0	1	0	0	1	1	0	0	1	1	0
ball	0	1	0	0	0	0	1	0	0	0	0
rectangular container	0	1	0	0	0	0	0	1	0	0	0
tube	0	1	0	0	0	1	0	0	0	0	0
blender	0	1	0	0	0	1	0	0	0	0	1
bowl	1	0	0	0	0	0	1	0	1	0	0
glass	0	0	0	1	0	1	0	0	1	0	0
plastic cup	0	1	0	0	0	1	0	0	1	0	0
measuring cup	0	1	0	0	0	1	0	0	1	1	0
jar	0	1	0	1	0	0	0	1	0	0	0
salter	0	0	0	1	1	1	0	0	0	0	1

Table 4.4: Class-attribute matrix $\boldsymbol{K}$ for training (upper) and test (lower) objects.

4.4.6 Data collection

The next step is data collection from the training objects, to learn attributes. We note that our attributes can be decomposed into 2 categories: material and shape attributes. We refer to the state of art on haptic exploration to choose the best procedure a hand must perform in order to recognize shapes and materials. According to [87], materials can be measured by performing a static contact between sensors and object surface. Shape can be inferred from the hand grasping/enclosing the object. Thus, we choose to explore each object by grasping/enclosing, which combines the exploration procedures required to perceive both material and shape.

During the training phase, each training object is handed to the Shadow hand by an experimenter 10 times in random positions and orientations. The experimenter ensures

that the hand touched all the relevant parts of each object in the maximum number of possible orientations with respect to the object form and hand kinematic limits. In addition, the experimenter chooses specific positions for which some BioTacs could not touch the object. To grasp the object, the robot fingers are spread out to their joint limits. Then, they are closed by setting a desired constant current to each joint actuator. The currents (and corresponding cable tensions) are kept low enough to avoid damage during contact with object or self collisions, while ensuring a "good" contact between the BioTac and object. Once all joints stopped (either because the finger contacted the object, or because it reached its joint limit), the contact is maintained for 20 seconds to obtain the thermal equilibrium between sensors and object. Three examples are illustrated in Fig. 4.6.

Figure 4.6: Examples of object grasps by the Shadow Hand.

Since our attributes have different nature, we assume attribute independence and learn each one separately. From each grasp, we gather BioTac readings x_{biotac} for material attributes classification, and encoders readings x_{shadow} for shape attributes classification. After exploring all objects from Y, the collected data are used to build a training set $D^{a_m}_{train}$ for each shape attribute and $D^{a_m}_{train,b}$ for each material attribute-BioTac pair.

Each test object is grasped up to 15 times, again using the same grasping procedure as during training, to build the test set D_{test}. Grasps in which none of the fingers touched the object are dropped as they do not include material information. A video of the experiments is provided on the IDH YouTube channel[9].

4.4.7 Feature extraction and attributes classification

For material attributes, we choose the same feature extraction technique as in the PHAC-2 experiments (see Sect. 4.2.3). We use a time average of the features and linear SVM as suggested in [53] with C equals to 1. But in contrast to that work, since we perform a static contact, we do not consider the vibrations signal P_{AC} from the BioTac. We obtain a feature vector of 22 normalized means which is reduced using PCA to a 8-dimensional vector x_{mat} that explains more than 98% of the variance. The resulting x_{mat} is used for the classification of the material attributes.

For each shape attribute, a binary classifier is trained using joints position measurements from the Shadow hand x_{shadow}. Joints that do not contribute to the closing proce-

[9]https://youtu.be/Ekd28b0BiQs

dure (e.g. the wrist) are excluded, resulting in a feature vector x_{sh} of 10 angular positions input to the shape attributes classifiers. The classification is performed using nonlinear SVM with a Gaussian kernel having C and γ equal to 1. We use the same sigmoid function as in Sect. 4.2.4 to convert the SVM classification score to an attribute posterior.

4.4.8 DAP – handling robotic constraints

To classify a test sample x, we introduce it in each attribute binary classifier, to obtain a set of posteriors $\{p(a_m \mid x), m = 1, \ldots, 11\}$. The attributes posteriors and object-attribute matrix are used by DAP (see Sect. 3.3.1) to return the object class.

When the hand grasps an object, it provides a data sample $x = (x_{sh}, x_{mat})$ where $x_{mat} = (x_{1,mat}, \ldots, x_{B,mat})$ with $1 \leq B \leq 5$ depending on the number of BioTacs in contact, which varies from a grasp to another. Thus, the attributes classifiers return $p(a_m \mid x_{sh})$ for shape attributes and $\{p(a_m \mid x_{b,mat}), b = 1, \ldots, B\}$ for material attributes. These posteriors must be combined to infer the final attribute posterior required in equation (3.6).

Furthermore, the attribute posteriors have to take into account several constraints posed by our realistic experimental setup, which were absent when we worked with the PHAC-2 database:

1. Test objects can be made of multiple materials. Thus, BioTacs on different fingers can be in touch with different materials (as in the salter grasp in Fig. 4.6), requiring the integration of information from different fingers to estimate the material attributes.

2. Objects can be heterogeneous and a grasp may provide only a local view. For example, in Fig. 4.6 (right) the hand grasps the lower part of the bottle and misses the presence of the narrow neck. Therefore, we need to deal with information missing from touched parts and to combine grasps to obtain a global view.

3. The number of touching fingers B can vary from one grasp to another, giving a different size of the test sample each time, which must be taken into account in multi-grasp classification.

In the next sections, we propose and test different solutions to take into account these constraints.

4.4.8.1 Single grasp classification – Combining information from various BioTacs

Since the test object is grasped in an unknown pose, the number of BioTacs B making contact may vary for each grasp. The contact of a BioTac is detected when the difference in static pressure exceeds a given threshold. For each material attribute a_m, we obtain a set of posteriors from the contacting BioTac classifiers $\{p(a_m \mid x_{1,mat}), \ldots, p(a_m \mid x_{B,mat})\}$. To obtain the final attribute posterior for the material attributes, we test both MAXDAP and AVGDAP (see Sect. 4.2.5) to combine classifications from the contacting BioTacs.

MAXDAP considers only the BioTac that is most confident about the presence of the a_m, and is implemented as:

$$p(a_m = 1 \mid x) = \max_{b=1,\dots,B} p(a_m = 1 \mid x_{b,mat}). \tag{4.5}$$

AVGDAP on the other hand, combines all contacting BioTacs by averaging their posteriors:

$$p(a_m = 1 \mid x) = \frac{1}{B} \sum_{b=1}^{B} p(a_m = 1 \mid x_{b,mat}). \tag{4.6}$$

Finally, while the material attribute posteriors are obtained using either MAXDAP or AVGDAP, the shape attributes are assessed as: $p(a_m = 1 \mid x) = p(a_m = 1 \mid x_{sh})$.

4.4.8.2 Single grasp classification – Handling local view

In regard to both material and shape attributes, a grasp may miss some attributes that are not present in the touched part of the object. This implies that if $p\left(a_m = 1 \mid x\right) < 0.5$, the attribute can be absent from the whole object or only from that particular grasp. To alleviate the effect of a possible misclassification, we replace the attribute posterior with a uniform distribution if the attribute is absent from x. For shape attributes:

$$p(a_m = 1 \mid x) = max(0.5, p(a_m = 1 \mid x_{sh})), \tag{4.7}$$

and for material attributes:

$$p(a_m = 1 \mid x) = \frac{1}{B} \sum_{b=1}^{B} max(0.5, p(a_m = 1 \mid x_{b,mat})). \tag{4.8}$$

We refer to this method as 'Local DAP' or LDAP.

4.4.8.3 Multi-grasp classification – Developing a global view

While single grasp classification can recognize objects by making some assumptions about absent attributes, combining several grasps is obviously advantageous as it gives a wider view of the object, and thus is expected to improve recognition performance [134]. Grasping an object T times in different positions results in a set $x = \{x^{(1)}, \dots, x^{(T)}\}$ of test samples. We compare two approaches to exploit information from multiple grasps: data fusion and decision fusion.

Data fusion merges data from the T grasps to form one "super grasp", that can be used to classify an object just like a single grasp. Formally, for shape attributes, we have:

$$p(a_m = 1 \mid x) = \frac{1}{T} \sum_{t=1}^{T} max(0.5, p(a_m = 1 \mid x_{sh}^{(t)})), \tag{4.9}$$

and for material attributes:

$$p(a_m = 1 \mid x) = \frac{1}{T} \sum_{t=1}^{T} \frac{1}{B_t} \sum_{b=1}^{B_t} max(0.5, p(a_m = 1 \mid x_{b,mat}^{(t)})). \tag{4.10}$$

We refer to this method as 'Data Fusion for Multi-grasp DAP ' or DF-MDAP.

Decision fusion scans the set of grasps from $x^{(1)}$ to $x^{(T)}$ and classifies each grasp separately using single-grasp method. The final classification is made when we obtain k grasps that are classified as the same object class, which is taken as the final decision on the object giving this sequence of grasps. We refer to this method as'Similar classifications for Multi-grasp DAP' or SC-MDAP.

4.5 Experimental evaluation and discussion

We make a series of experiments to evaluate the multiple adaptations of DAP to our experimental setup. For better analysis, we decompose the test set Z into 3 subsets: objects having homogeneous material and shape properties Z_{hom} ={*ball*, *rectangular container*, *tube*}, objects whose shape properties can differ according to the touched part $Z_{het,sh}$ ={*blender leg*, *bowl*, *glass cup*, *plastic cup*, *measuring cup*} and objects made of multiple materials: $Z_{het,mat}$ ={*jar*, *salter*}.

4.5.1 Attributes classification

First, we evaluate the performance of the binary classification of attributes. In Fig. 4.7, we present the accuracies of attribute classifiers on the test set. All accuracies are averaged across the test trials. Material attributes are additionally averaged over the five BioTacs. Overall, attribute classification achieves a satisfying average accuracy of 78%. However, attribute *plastic* has an accuracy of 45% which is considerably lower than the rest. This is probably due to the variety of plastic types used in the training and test sets. For example, the round container is made of a softer plastic than the blender leg.

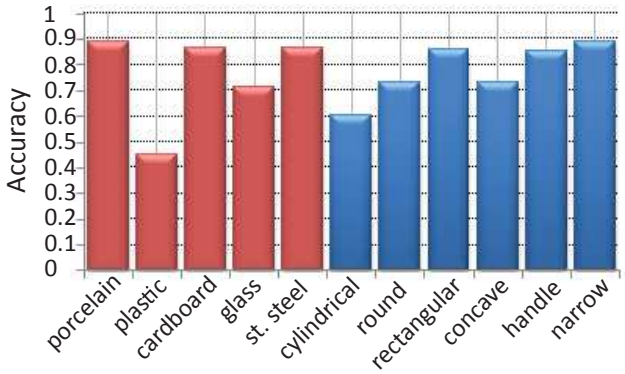


Figure 4.7: Recognition accuracy of attributes binary classification.

To analyze the performance of attributes classifiers on each object, we average $p(a_m = 1 \mid x)$ for each object-attribute pair across the test trials of each object. Ideally, if x is collected from object z_l then $p(a_m = 1 \mid x)$ should be close to $a_m^{z_l}$ since $p(a_m = 1 \mid x) \approx 1$ if $a_m^{z_l} = 1$ and $p(a_m = 1 \mid x) \approx 0$ otherwise. Matrix $\boldsymbol{S}$ presented in Fig. 4.8 measures L_1

distance between attributes binary labels and their posteriors:

$$\boldsymbol{S}_{l,m} = || \; a_m^{z_l} - \frac{1}{\mid D_{test}^{z_l} \mid} \sum_{x \in D_{test}^{z_l}} p(a_m = 1 \mid x) \; ||_1, \tag{4.11}$$

where $D_{test}^{z_l}$ represents the set of test samples gathered from object z_l. We note that the majority of distances are lower than 0.5 (white or yellow cells in the table), indicating that objects are well classified by the corresponding attribute classifier. However, some classifiers perform badly on some objects, probably because these are *too* different from the ones they have trained on. For instance, the *plastic* attribute column presents high distances for the ball, rectangular container and jar because of the difference between the plastic material constituting these objects, and that constituting the training objects.

	porcelain	plastic	cardboard	glass	st. Steel	cylind.	round	rectang.	concave	handle	narrow
ball	0.38	0.61	0.09	0.25	0.16	0.33	0.58	0.20	0.02	0.29	0.22
rec. cont.	0.21	0.73	0.33	0.14	0.21	0.43	0.15	0.35	0.05	0.11	0.11
tube	0.11	0.36	0.14	0.48	0.19	0.08	0.29	0.10	0.06	0.02	0.25
blender	0.19	0.54	0.12	0.27	0.25	0.60	0.01	0.33	0.37	0.06	0.33
bowl	0.17	0.38	0.01	0.31	0.12	0.13	0.60	0.19	0.51	0.25	0.17
glass	0.20	0.35	0.07	0.38	0.09	0.50	0.04	0.11	0.59	0.06	0.32
plas. cup	0.05	0.34	0.06	0.46	0.20	0.50	0.02	0.09	0.50	0.01	0.56
msr. cup	0.14	0.47	0.21	0.37	0.05	0.46	0.08	0.15	0.67	0.76	0.23
jar	0.12	0.70	0.24	0.67	0.21	0.59	0.15	0.45	0.06	0.03	0.16
salter	0.14	0.29	0.08	0.62	0.62	0.38	0.06	0.31	0.11	0.01	0.21

$S_{l,m} \leq 0.25$
$0.25 < S_{l,m} \leq 0.5$
$0.5 < S_{l,m} \leq 0.75$
$0.075 < S_{l,m}$

Figure 4.8: Distance between attributes binary labels and their posteriors for test objects.

4.5.2 Single grasp DAP

Recognition accuracies of test objects using a single grasp are presented in table 4.5. First, it is clear that the accuracy obtained with homogeneous objects is better than that obtained with heterogeneous ones. This is because all object properties can be felt from a single grasp. Furthermore, all BioTacs touch the same material over a homogeneous object, "collaborating" to give a more confident classification. However, this collaboration becomes more delicate for objects belonging to $Z_{het,mat}$, which explains the accuracy deterioration. Moreover, we note that AVGDAP performs better than MAXDAP for heterogeneous objects. This is probably because AVGDAP averages the decision from the BioTacs, and is hence less sensitive to errors from individual sensors. However, LDAP outperforms all methods because we found that more often than not, random grasps on daily life objects lead to missing some parts of the object, and LDAP can deal efficiently with absent attributes.

Method	Z_{hom}	$Z_{het,sh}$	$Z_{het,mat}$	Z
MAXDAP	**0.78**	0.45	0.20	0.52
AVGDAP	**0.78**	0.50	0.33	0.57
LDAP	0.75	**0.55**	**0.40**	**0.60**

Table 4.5: Recognition accuracy of DAP on test objects from a single grasp.

Table 4.5 should however be viewed considering that single grasps, by nature, are constrained by local object properties. Consider the three example grasps on the measuring cup during the single grasp classification (Fig. 4.9). We find that for 83% of grasps having the fingers touching the cylindrical part, the measuring cup is classified as a *tube*. All the grasps touching the upper side lead to the measuring cup being classified as the *pink cup*, and 66% of grasps touching the handle classify the object as a measuring cup. These cases should not all be considered as misclassifications because this is the best decision that can be made from the given local grasp. For instance, a human touching the cylindrical part of the measuring cup can also not be sure if this is the tube, the pink cup or the measuring cup, since the three objects share the same local shape. Hence, table 4.5 highlights the limitation of single grasps for object recognition.

Figure 4.9: Representative grasps on the "measuring cup" show how different grasps may indicate distinct features.

4.5.3 Multi-grasp DAP

Using a single grasp, LDAP gave the best recognition accuracy of 0.6 over all objects, which is modest. This encourages multi-grasp recognition. Hence, we check the recognition accuracy with multiple grasps, using both DF-MDAP and SC-MDAP. In Fig. 4.10, we analyze the DF-MDAP and SC-MDAP performances by changing their parameters T and k, respectively. We note that DF-MDAP accuracy increases constantly with the number of merged grasps T. This is understandable as an increase of T improves the information contained in the “super grasp". Since LDAP gave the best accuracy, we use it to perform each single grasp classification for SC-MDAP. For SC-MDAP, the best accuracies are obtained between $k = 2$ and $k = 4$. The presence of an *optimal* k can be explained by the fact that, while increasing the number of required similar classifications

helps removing noisy classifications, it becomes harder to find k similar classifications when k becomes large. Overall, both methods improve the performance compared to single grasp recognition ($k = 1, T = 1$).

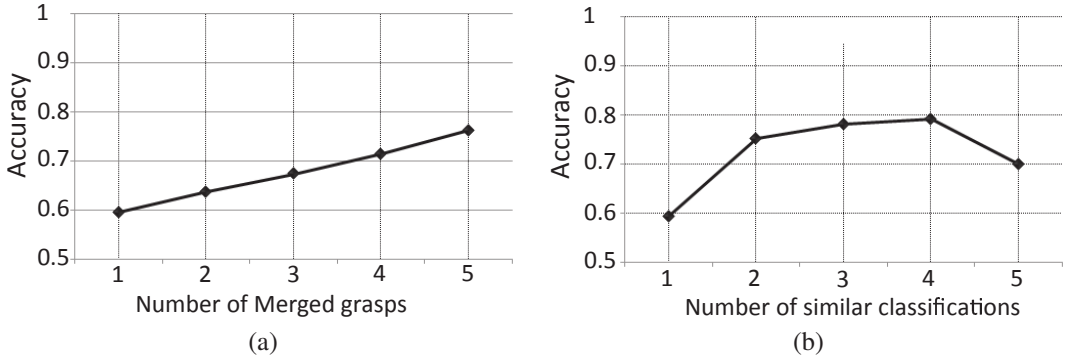


Figure 4.10: Recognition accuracy vs. number of combined grasps T for DF-MDAP (a) and vs. number of similar classifications k for SC-MDAP (b).

Figure 4.11 shows the confusion matrices with different DAP choices. The best classifications are achieved with SC-MDAP. We note that multi-grasp recognition is very beneficial to correct Z_{hom} misclassifications, to the point that SC-MDAP reached an accuracy of 100% for all objects. For $Z_{het,sh}$, we note that the blender, bowl and glass are correctly classified by both multi-grasp methods. The plastic cup is still frequently confused with a tube by DF-MDAP but these misclassifications are overcome with SC-MDAP. However, all methods perform weakly on the measuring cup and on the jar, which are often confused with the tube. This is because the attributes classifications are poor on these objects (see distance matrix in Fig. 4.8), subsequently influencing their classification.

4.5.4 Summary

Through a series of experiments, we analyzed and developed a haptic ZSL algorithm for an anthropomorphic robot hand. This algorithm enables good recognition of daily life objects that our robot encountered for the first time (see Fig. 4.11c). In Sect. 4.5.1, we evaluated the ability of our setup to recognize each attribute and found that the performance differs significantly from an attribute to another. Some attributes (e.g. *made of porcelain*) were efficiently classified, while for others the system failed to generalize models learned from the training objects to the novel test objects (e.g. *made of plastic*). This motivated us to investigate how we can combine the attributes classifications to explicitly handle the uncertainty of each sensor (Sect. 4.5.2). Results from table 4.5 show that AVGDAP performed better than MAXDAP because it takes into consideration possible miss-classifications due to the noise/uncertainty of sensors. Moreover, results were improved by using LDAP, which also accounts for the uncertainty regarding the untouched parts of the object. Finally, in Sect. 4.5.3 we showed how to implicitly build a

		ball	rect. cont.	tube	blender	bowl	glass	plas. cup	msr. cup	jar	salter
Z_{hom}	ball	0.6	0.1	0.2	0	0.1	0	0	0	0	0
	rec. cont.	0.06	0.75	0.19	0	0	0	0	0	0	0
	tube	0	0	0.9	0	0	0	0	0	0	0.1
$Z_{het,sh}$	blender	0	0	0	0.53	0	0	0.2	0	0.13	0.13
	bowl	0	0	0	0	1	0	0	0	0	0
	glass	0	0	0.3	0	0	0.7	0	0	0	0
	plas. cup	0	0	0.3	0	0	0.1	0.5	0	0	0.1
	msr. cup	0	0	0.46	0	0	0.08	0.31	0.15	0	0
$Z_{het,mat}$	jar	0.1	0.3	0.4	0	0	0	0	0	0.2	0
	salter	0	0	0	0.2	0	0	0	0	0	0.8

(a) LDAP

ball	rect. cont.	tube	blender	bowl	glass	plas. cup	msr. cup	jar	salter
0.85	0	0.15	0	0	0	0	0	0	0
0	0.86	0.14	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0.99	0	0	0.01	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0.67	0.28	0	0.04	0	0	0	0
0	0	0.9	0	0	0.03	0.04	0.03	0	0
0	0.22	0.61	0	0	0	0	0	0.17	0
0	0	0	0	0	0	0	0	0	1

(b) DF-MDAP

ball	rect. cont.	tube	blender	bowl	glass	plas. cup	msr. cup	jar	salter
1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0.8	0	0	0	0.2	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1

(c) SC-MDAP

Figure 4.11: Confusion matrices of LDAP , DF-MDAP ($T = 5$) and SC-MDAP ($k = 4$). SC-MDAP achieved perfect recognition of 8 out of 10 new objects.

global view of the object by combining multiple grasps. Object recognition is improved with a global view as this can account for attributes confined to certain parts of the object.

4.6 Conclusion

In this chapter, we proposed a haptic Zero-Shot Learning algorithm that enables a robot to recognize novel objects, which it has never been trained on before. First, we applied ZSL to probably the best haptic object database for attribute-based ZSL (PHAC-2), which includes 60 objects with a wide variety of texture, material and stiffness properties. This allowed us to analyze the possibilities and constraints associated with the application of ZSL to haptic recognition. We used this analysis to understand if and how the choice of attribute classifier (Sect. 4.2.4), the way of combining attribute posteriors (Sect. 4.2.5), the choice of training set (Sect. 4.3.4) and the number of attributes (Sect. 4.3.5) affect recognition performance. The algorithm developed from this analysis was then applied

on an anthropomorphic robot hand, to make it recognize unexplored objects (Sect. 4.4). In this realistic scenario, we further optimized our algorithm and enabled it to account for heterogeneous objects (Sect. 4.4.8) and to integrate information from multiple grasps (Sect. 4.4.8.3). Our final algorithm enabled the robot to correctly recognize eight out of ten objects, that it grasps for the first time (third panel of Fig.4.11).

Note that the recognition rate in our real robot experiment (Fig.4.11) was much better than in the PHAC-2 database recognition (Tables 4.1 and 4.2). Comparison is difficult, as different objects were used in the two cases. However, it is still interesting to note that the real robot implementation worked better, considering that it used coarse grasps to explore heterogeneous objects of different shapes, compared to recognition with the PHAC-2 database in which objects were homogeneous, regularly shaped, and explored using a regular and well controlled procedure. However, we believe that the results of our robot experiment are in fact a better indicator of the capabilities of our algorithm for haptic object recognition. This is because of several reasons. Primarily, the PHAC-2 database focuses on the tactile properties of objects, and not on shape. On the other hand, shape is definitely a fundamental feature for haptic recognition during grasps, and our robot is able to integrate both tactile and shape information (through its joint angles) efficiently to improve object recognition. Moreover, while homogeneous objects (as in PHAC-2) may intuitively seem easier to recognize, this may not always be true. Most real life objects are heterogeneous and in fact the heterogeneous nature of an object, if explored well (we propose one way in Sect. 4.4.6), can act as a signature for the object, making it easier to recognize. This was probably the case in our robot experiment as well. Finally, the recognition of the two discussed features, i.e. material heterogeneity and shape, are further improved by our algorithm by using multiple grasps in our robot experiment. In PHAC-2 on the other hand, the homogeneity of objects and regularity of explorations make the data similar across exploration trials, making simulated multiple grasps irrelevant.

The results obtained with our robotic setup show the promising capabilities of haptic ZSL for object recognition. This is very encouraging, given that this is still a prototype system that can be improved in several aspects. First, we studied haptic object recognition by assuming that vision is unavailable. Our proposition for extending the current system with vision is presented in the next chapter. Second, in this chapter we focused on the ZSL problem by assuming that all evaluated objects have never been experienced before. However, robots encounter previously explored as well as unexplored objects, and the integration of ZSL with other suggested multi-class object recognition systems can enable them to recognize known objects and progressively integrate novel ones. Therefore, we propose in chapter 6, a system capable of handling both known and novel objects.

Chapter 5

Visuo-Tactile Recognition of Daily-Life Objects Never Seen or Touched Before

5.1 Introduction

Information from multiple senses can be used for object recognition; the most prominent ones for this are vision and touch. However, object recognition systems use either vision [83] or touch [134], but less frequently both modalities together, despite the efficiency of their combination for improving the recognition performance. An example is the cross-modal visuo-tactile object recognition system in [29]. While tactile data are relevant to perceive the object's material, texture and compliance properties, the integration of vision can improve the performance by perceiving properties such as shape and color.

Therefore, this chapter extends our haptic attribute-based ZSL system, presented in chapter 4, with vision. We provide in [2] probably the first visuo-tactile recognition system that can handle objects that have neither been seen nor touched during the training phase. To the best of our knowledge, several studies have been carried out on visual ZSL [157], only one on haptic (tactile and shape) ZSL [1], but there are no studies on visuo-tactile ZSL. A recent study [93] suggests a hybrid ZSL framework that combines visual and tactile data, but the authors use visual data available for novel classes to perform tactile ZSL. Whereas in this work neither tactile nor visual data are available for any of the novel objects.

We improve our haptic ZSL framework proposed in [1] and presented in chapter 4 as follows:

- We improve attributes learning by replacing hand-crafted feature extractors with deep Convolutional Neural Networks. CNNs are used to classify attributes based on both tactile and visual data;
- We adapt the Direct Attributes Prediction (DAP) model used in [1] to take into account both visual and tactile modalities;
- For recognizing a novel object, three scenarios are investigated and compared:

1. using vision only, assuming the object cannot be touched, e.g. it is far from the robot;
2. using touch only, e.g. assuming the robot operates in the dark;
3. using both vision and touch.

- We applied our visuo-tactile recognition on the PHAC-2 dataset, leading to further adaptations. Since PHAC-2 objects have simple similar shapes and have a variety of texture, material and compliance properties, we exclude kinesthetic data and we use only tactile data. Thus, we do not take into consideration shape-related properties and focus on tactile properties only;
- In addition to the haptic attributes provided with the PHAC-2 dataset, we add more attributes describing the visual properties of the objects that cannot be described using haptic attributes.

This chapter is organized as follows. Sect. 5.2 presents our solution for integrating visual and tactile modalities. Then, our experimental setup and experimental evaluation are presented in sections 5.3 and 5.4 respectively. Finally, in Sect. 5.5, conclusions are provided.

5.2 Visuo-Tactile Zero-Shot Learning

In this section, we apply the theoretical framework presented in Sect. 3.3.1 to perform tactile, visual or visuo-tactile ZSL. When the robot sees an object, its visual sensors provide visual images in X^V. In addition, by physically interacting with the object, the robot tactile sensors provide tactile data samples in X^T. X^V and X^T are feature spaces in which respectively visual and tactile data are represented. Thus, our aim is to adapt the framework presented in Sect. 3.3.1 to data sample $x = [x^V, x^T]$ in $X = X^V \times X^T$.

5.2.1 Attributes Learning

The first step is to learn from training data how to predict the presence of each attribute in an object, i.e. to compute $\{p(a_1 \mid x), \ldots, p(a_M \mid x)\}$ given $x = [x^V, x^T]$. We propose three solutions. The first solution learns a binary classifier per attribute that predicts its presence in tactile data x^T. The second one learns a binary classifier per attribute predicting its presence in visual data x^V. The advantage of separating tactile and visual modalities for attributes prediction is that the system is operational even when only one sensor is available. The third solution learns one classifier per attribute having as input $x = [x^V, x^T]$. Its advantage is to use (when both are available) visual and tactile data together to learn the attribute.

First, to learn a tactile classifier per attribute, we replace the hand-crafted features extractor and SVM classifier used in [1] by a CNN, which requires representing tactile signals in the form of a tactile image. By deriving a tactile image from x^T, the CNN

automatically extracts discriminative features and predicts the presence of the attribute in the object. Likewise, we train a binary CNN per attribute, predicting its presence in a visual image x^V .

The third solution classifies both tactile and visual data at the same time using one CNN. As illustrated in Fig. 5.1, we extract features from each sensor separately using two independent convolutional networks. Then, we concatenate the tactile and visual CNN features to form one visuo-tactile feature vector that we classify using a fully connected neural network.

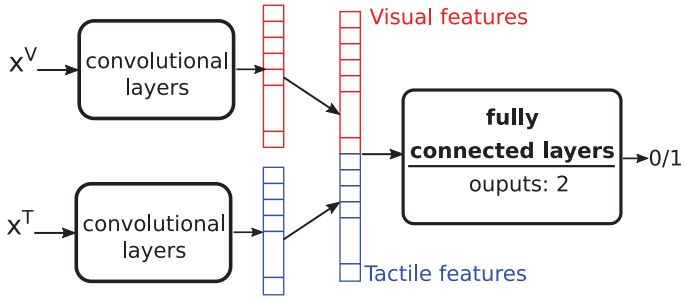


Figure 5.1: Visuo-tactile CNN classifying an attribute as absent (0) or present (1) based on visual and tactile data.

5.2.2 Visuo-Tactile DAP

The second step is to use the output of the attribute classifiers, described in 5.2.1, to compute posteriors of Z objects according to:

$$p(z_l \mid x) = \frac{p(z_l)}{p(\boldsymbol{a}^{z_l})} \prod_{m=1}^{M} p(a_m^{z_l} \mid x), \tag{5.1}$$

This requires the prediction of all attributes posteriors $p(a_m \mid x)$. However, choosing CNNs for attributes classification provides us with a classification score $s_m(x) \in \mathbb{R}$. To transform this score into a posterior probability, we use the following sigmoid function:

$$p(a_m \mid x) = 1/(1 + e^{-s_m(x)}). \tag{5.2}$$

Thus, for each attribute, given the test sample $x = [x^V, x^T]$ we have three posteriors: $p(a_m \mid x^T)$ returned by the tactile CNN, $p(a_m \mid x^V)$ returned by the visual CNN and $p(a_m \mid x^V, x^T)$ returned by the visuo-tactile CNN. Thus, inferring $p(a_m \mid x)$ used in (3.6) to compute $p(z_l \mid x)$ of each $z_l \in Z$ can be performed based on:

1. **Tactile data only:** by replacing $p(a_m \mid x)$ in (3.6) with $p(a_m \mid x^T)$. We refer to this method as Tactile-ZSL (denoted T-ZSL);

2. **Visual data only:** by replacing $p(a_m \mid x)$ in (3.6) with $p(a_m \mid x^V)$. We refer to this method as Visual-ZSL (denoted V-ZSL);

3. **Both tactile and visual data:** we proceed in two ways:

   (a) by replacing $p(a_m \mid x)$ in (3.6) with $p(a_m \mid x^V, x^T)$. We refer to this method as "Visuo-Tactile Features Concatenation ZSL" (denoted VT-FC-ZSL);

   (b) by combining the two independent visual and tactile attributes posteriors to compute a visuo-tactile attribute posterior:

$$p(a_m \mid x) = tact(a_m)\, p(a_m \mid x^T) + vis(a_m)\, p(a_m \mid x^V), \tag{5.3}$$

   Here $tact(a_m) \in [0, 1]$ and $vis(a_m) \in [0, 1]$ are user-tuned scores given to the importance of tactile and visual modalities for classifying attribute a_m respectively, s.t. $tact(a_m) + vis(a_m) = 1$. We refer to this method as "Visuo-Tactile Scores Merging ZSL" (denoted VT-SM-ZSL).

5.3 Experimental Setup

We use the PHAC-2 dataset [19] to perform our visuo-tactile ZSL. In addition to the haptic data collected by the gripper of a PR2 robot, presented in Sect. 4.2.1, PHAC-2 includes visual data for 53 objects. Eight RGB images are given for each of the 53 objects from different viewpoints, see Fig. 5.2. We keep here the same attributes set as in Sect. 4.2.1.

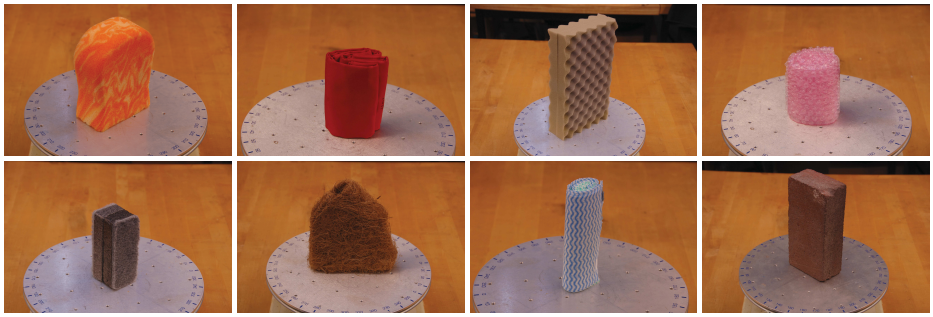


Figure 5.2: Example of images taken for some of the PHAC-2 objects [19].

5.3.1 Data Augmentation and Pre-processing

In PHAC-2, the available data for each object are very few; only 10 tactile samples and 8 visual samples per object. This requires both tactile and visual data augmentation. As in [38], we augment tactile data by combining data from BioTacs and also sub-sampling the signals measured by each BioTac using five different starting points, resulting in 100 samples per object instead of 10 (10 trials $\times$ 2 BioTacs $\times$ 5 starting points). Each sample includes 23 BioTac channels: static pressure, vibrations, temperature, heat flow and 19 electrode voltages. PCA is applied to electrode voltages and the first four principal components are kept, giving 8 signals per BioTac. Then, each of the 8 signals are sub-sampled

to 30 time samples. Next, by separating between the four EP (including a squeeze, hold, slow slide and fast slide), the 8 signals of each exploration step are concatenated to obtain a 32-dimensional signal (4 exploration steps $\times$ 8 signals). Therefore, the space in which tactile data are represented is $X^T = \mathbb{R}^{32\times30}$. On the other hand, 8 RGB images of resolution (224×224) yield a visual features space $X^V = \mathbb{R}^{3\times224\times224}$. This space is augmented by rotating each image multiple times and zooming in objects surface, resulting in 80 images per object instead of 8.

5.3.2 Objects Splits

By definition, performing ZSL requires splitting of object set into two disjoint sets: Y and Z. Since we aim at developing a visuo-tactile recognition system and visual data are not available for all 60 objects, we keep only the 53 objects for which visual data are available. We randomly select 6 objects ($\approx 11.3\%$) having different attributes vectors as Z objects, and the remaining objects as Y objects. We repeat the process 7 times in order to generate 7 random (Z, Y) splits to ensure the independence of the results from the choice of objects.

5.4 Evaluation and Results

5.4.1 Implementation Choices

We implemented our framework using Python based on [1] and other public projects[1][2]. CNNs are implemented using caffe [61]. The architecture of tactile the CNN is the same as in [38] and is illustrated in Fig. 5.3. Figure 5.4 illustrates the visual CNN architecture. We use a pre-trained model of GoogleNet [145] as a feature extractor for the visual data. We compared the BVLC[3] and MINC [8] GoogleNet pre-trained models and we found relatively similar results. Thus we choose the MINC model to have results comparable with [38]. Then, the extracted visual features are averaged and classified using a fully connected neural network that predicts the presence of the attribute. Finally, the convolutional parts of each of Fig. 5.3 and Fig. 5.4 are used to extract tactile and visual features for the visuo-tactile CNN illustrated in Fig. 5.1.

5.4.2 Attributes Learning

First of all, we focus on attributes learning using tactile data only. We compare our previous hand-crafted features extractor and SVM method [1] with the current deep classification method. Fig. 5.5 illustrates for each of the seven object splits defined in Sect. 5.3.2 the attribute binary classification accuracies averaged over all attributes. We note that

[1] https://github.com/IanTheEngineer/Penn-haptics-bolt
[2] https://people.eecs.berkeley.edu/~yg/icra2016/
[3] https://github.com/BVLC/caffe

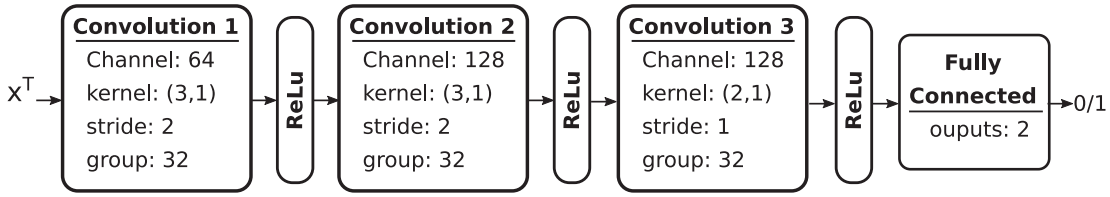


Figure 5.3: Architecture of CNN for predicting attribute presence (1) or absence (0) based on tactile data.

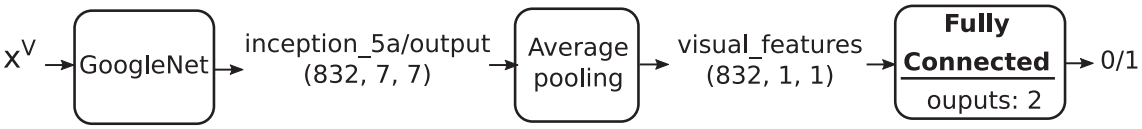


Figure 5.4: Architecture of CNN for predicting attribute presence (1) or absence (0) based on visual data.

for all the splits, CNN classification outperforms SVM with an average improvement of 4.37%. This shows the efficiency of deep learning in automatically extracting features and classifying them at the same time compared to hand-crafted features and separated classifiers.

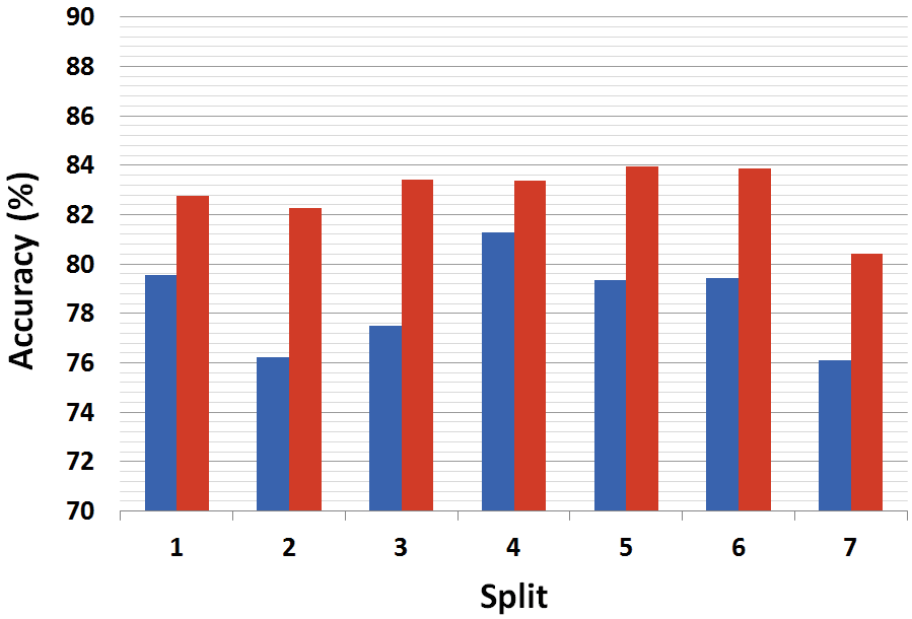


Figure 5.5: Attribute binary classification accuracies for all object splits averaged over all attributes (tactile SVM [1] in blue and tactile CNN in red).

In Sect 5.2.1, we propose three methods for learning attributes: based on tactile data only, based on visual data only and based on the concatenation of visual and tactile CNN features. Fig. 5.6 presents the comparison of the three classifiers of each attribute trained

and tested based on split 1. We note that the performance changes from an attribute to another. Some attributes such as *bumpy, metallic* and *squishy* are better classified using tactile data. Some attributes such as *rough, springy* and *textured* are better classified from visual data. Others such as *absorbent, compressible* and *hard* are better classified using both tactile and visual data. These results are coherent with [38] where the authors find that some PHAC-2 haptic attributes are better classified using visual data than using tactile data. Overall, 8 attributes are better classified using tactile data, 3 using visual data and 8 using visuo-tactile data. The fact that less attributes are better classified using vision only is obvious, since there are more haptic than visual attributes in our list. Besides, for 8 attributes, merging visual and tactile data improves learning compared to learning from each modality separately, which is promising for combining both modalities using VT-SM-ZSL.

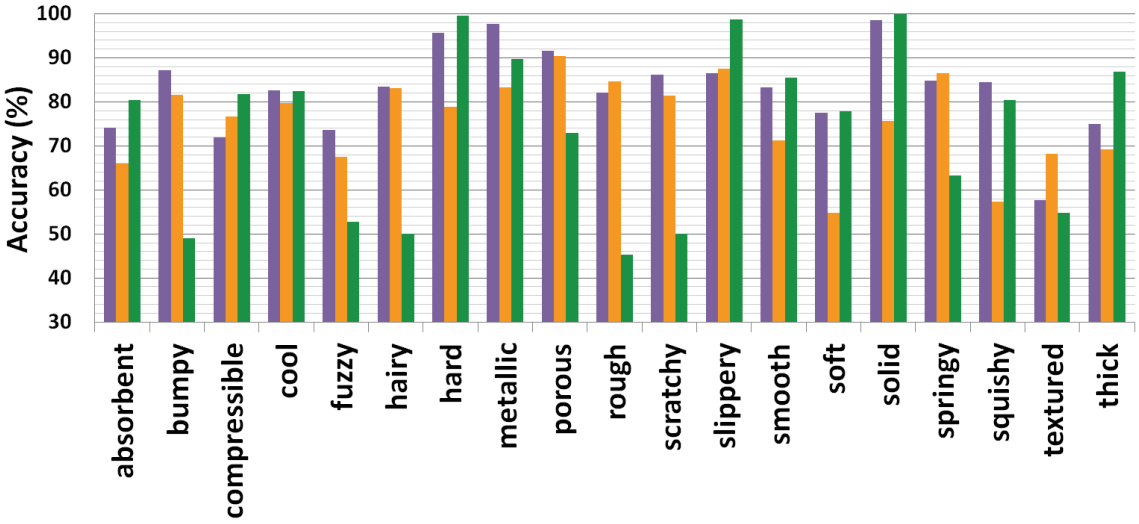


Figure 5.6: Attributes classification accuracies for split 1: purple with tactile data alone, yellow with visual data alone, and green with both visual and tactile data.

5.4.3 Visuo-Tactile DAP

Here we present results of classifying a test sample $x = [x^V, x^T]$ as one of the 6 objects in Z. Knowing that we have zero training data for each of the 6 objects, classifying them with traditional classifiers gives an average classification accuracy of 16.67% which is the random accuracy of classifying 6 objects.

Table 5.1 compares DAP classification accuracies for classifying Z objects based on x^T only (T-ZSL), x^V only (V-ZSL) and $[x^V, x^T]$ (VT-FC-ZSL). Results show that most splits are better classified with visuo-tactile data, some of them with tactile data only, and none with visual data only. This was expected from results of attributes learning. However, we note that even though visual data alone are not very efficient for classifying objects, they efficiently improve tactile recognition in 5 out of 7 cases.

Table 5.1: Comparison of tactile, visual and visuo-tactile ZSL recognition accuracies (%).

Split	T-ZSL	V-ZSL	VT-FC-ZSL
1	60.5	31.46	**71**
2	40.5	46.25	**53.33**
3	43.67	54.17	**61.95**
4	**62.83**	37.71	56.28
5	41.83	28.13	**42.97**
6	**64.33**	33.96	57.73
7	33.33	35.21	**54.88**
Average	49,57	38,13	**57,31**

Motivated by the good results obtained with the concatenation of visual and tactile features using VT-FC-ZSL, we continue investigating visuo-tactile DAP by using VT-SM-ZSL. With this method, defined by (5.3), the importance of each modality for classifying an attribute should be tuned, via $tact(a_m)$ and $vis(a_m)$. In table 5.2 compares three methods for computing $tact(a_m)$ and $vis(a_m)$. The first *binary* method gives a binary importance $tact(a_m) = 1$ and $vis(a_m) = 0$ if the attribute classification accuracy using tactile data is better than using visual data, and $tact(a_m) = 0,\ vis(a_m) = 1$ otherwise. The second *weighted* method gives a real valued importance: $tact(a_m) = acc^t_m/(acc^t_m + acc^v_m)$ and $vis(a_m) = acc^v_m/(acc^t_m + acc^v_m)$ where acc^t_m, acc^v_m are respectively the classification accuracies of tactile CNN and of visual CNN trained on classifying attribute a_m and tested on validation data. The third *uniform* method gives the same importance to vision and tactile data, i.e $tact(a_m) = vis(a_m) = 0.5$. Results show that the average accuracy of the binary method is greater than that the two other methods. Removing the least performing modality for each attribute helped to perform visuo-tactile DAP by taking the best of each modality which explains this improvement.

Table 5.2: VT-SM-ZSL recognition accuracies (%).

Split	binary	weighted	uniform
1	**55.27**	52.87	53.65
2	41.10	51.39	**51.79**
3	48.25	59.74	**61.13**
4	**57.08**	49.63	48.67
5	33.86	42.23	**42.34**
6	**49.98**	44.47	43.08
7	**51.53**	34.84	33.84
Average	**48.15**	47.88	47,79

5.4.4 Adding Visual Attributes

The experiments above use the haptic attributes provided by the PHAC-2 dataset as attributes to perform the ZSL. Results show that adding visual features to tactile ones gives an improvement of $7,74\%$ (see tables 5.1). This motivated us to try adding more visual attributes to further improve the visuo-tactile ZSL.

While haptic attributes describe the texture, compliance and material properties, visual attributes can better describe shape and color properties. Given that PHAC-2 objects have simple shapes with flat parallel sides, we assume that adding visual attributes describing object shapes will not be very effective. We therefore extend the attributes set with a set of color attributes. By observing objects' images, we define a set of 7 colors shared between all objects, which are $A_c = \{white, blue, yellow, red, beige, silver, black\}$. Also, we use human labeling to associate a binary value to each object-color pair.

First, for each color attribute, we train a visual CNN having the same architecture as for the haptic attributes (see Fig. 5.4). Figure 5.7 illustrates the classification accuracies of color attributes CNNs, obtained for each split. We note that the classification performance varies from an attribute to the other and from one set to the other. Overall, all colors are classified with more than 60% accuracy.

Next, we improve the DAP classification results by extending the haptic attributes with color attributes. We first improve V-ZSL by classifying all the 26 attributes (haptic and color) using visual data only. Results (reported in the first column of table 5.3) show the significant improvement of recognition accuracy for almost all object splits, compared to V-ZSL in table 5.1. This highlights the effectiveness of adding visual attributes to the haptic ones. Only split 3 shows a degradation in terms of accuracy, but this is coherent with the fact that it has the lowest average attributes classification accuracy of 83.15% and the lowest accuracy for classifying attribute *yellow* (60.83%). Furthermore, in table 5.3, we add color attributes for VT-FC-ZSL and VT-SM-ZSL by giving $vis(a_c) = 1$ and $tact(a_c) = 0$ for all color attributes. Compared to tables 5.1 and 5.2, this addition improves object classification for almost all splits, with an accuracy of 86% for split 6.

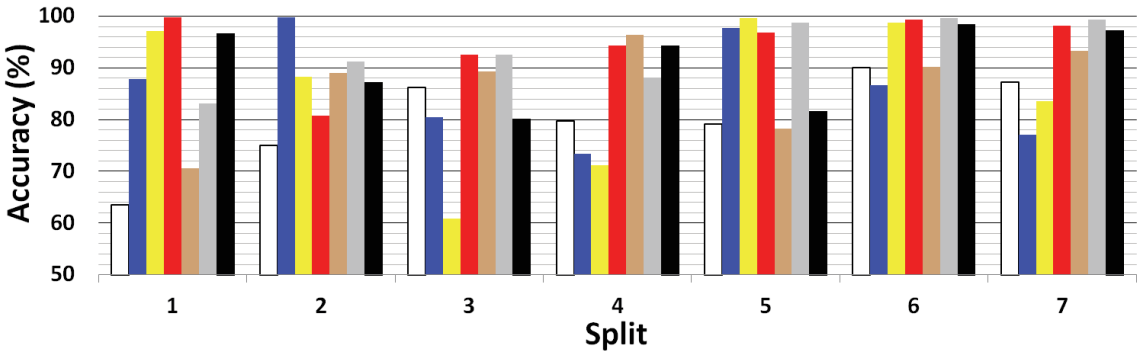


Figure 5.7: Color attributes classification using visual images (bars are colored by the colors they represent).

Table 5.3: Recognition accuracies (%) when adding color attributes to visual and visuo-tactile ZSL.

Split	V-ZSL+C	VT-FC-ZSL+C	VT-SM-ZSL+C
1	47.29	**77.48**	62.88
2	54.58	**66.67**	54.29
3	48.13	**59.58**	57.21
4	66.25	**75.1**	73.19
5	49.38	77.82	**80.38**
6	62.5	**77.82**	69.52
7	46.46	68.4	**86.27**
Average	53,51	**71.74**	66.53

5.5 Conclusion

This chapter extends our haptic recognition system, presented in chapter 4, with vision. It shows how replacing hand-crafted feature extraction with CNNs improved attributes learning (see Fig. 5.5) and how adding visual data to tactile data (see tables 5.1 and 5.2) significantly improved the Zero-Shot recognition accuracy. Finally, extending haptic attributes with visual ones improved the recognition performance (see table 5.3). The obtained improvement consolidates previous studies that highlighted the importance of visuo-tactile collaboration for improving robotic tasks. The next chapter presents another improvement of the haptic recognition system presented in chapter 4, which is the consideration of novel, as well as, known objects in the test set.

Chapter 6

Deep Learning for Tactile Recognition of Known and Novel Objects

6.1 Introduction

In the previous chapters, ZSL problem was handled by focusing on recognizing novel objects only. During the test phase, we assumed that test objects should be classified as one of a set of novel objects. However, in real life, the robot can encounter a novel object, as well as one of the training objects. Thus, this chapter proposes, a single recognition framework that, from tactile data, can recognize training objects as well as novel ones, without any prior knowledge if the test object is known or novel.

First, our proposed framework recognizes an object as a known (previously touched) or a novel object. Then, it uses previously available multi-class classifiers in case the object is known, and attribute-based Zero-Shot Learning in case the object is novel. Furthermore, our framework allows an efficient integration of new tactile data for novel objects, enabling a system that can handle objects starting from one training data sample, which is called One-Shot Learning (OSL). This can continuously improve recognition performance with experience.

Our proposed framework is based on deep CNNs. This choice was motivated by the good performance reported recently in multi-class tactile recognition by CNNs [37, 112]. However, note that the use of CNNs for Zero-Shot Learning is not straightforward: if one simply trains a CNN to map tactile data into object classes, the CNN will miss output classes having no training data. This decreases the classification accuracy due to the imbalance in the training set [52, 13]. To cope with this problem, synthetic tactile training data are generated for each novel class, given its attribute-based description.

Our main contributions are as follows:

1. We propose a comprehensive tactile recognition system, that can recognize classes having many training data as well as classes without or with very few training data, which is very common in daily-life tactile recognition.

2. Recognition of novel objects using their attribute-based description and tactile data

collected from training objects. This is a very important advantage in tactile recognition as it reduces tactile data collection and replaces it with semantic information which is much easier to obtain.

This chapter is structured as follows: Sect. 6.2 gives an overview of the proposed recognition framework. The details about the training data generation, and about the ZSL and OSL are presented in Sect. 6.3. The experimental setup is presented in Sect. 6.4. Finally, the evaluation results are presented and discussed in Sect. 6.5, and conclusions are given in Sect. 6.6.

6.2 Tactile Object Recognition Framework

6.2.1 Recognition Framework Overview

Figure 6.1 illustrates our proposed tactile recognition framework capable of recognizing both training and novel objects. To recognize a tactile test sample $x \in X$, it is processed by a convolutional network $CONV_{XF}$ that extracts a feature vector $f_{ext} \in F$. This latter is classified by one of two fully connected neural networks, FC_{FY} or FC_{FZ}, according to a novelty detection metric $ND(x)$. This metric predicts if x is novel or not; if x is novel, then f_{ext} is classified using FC_{FZ}, having only Z objects as outputs. If x has been collected from a training object, then f_{ext} is classified using FC_{FY}, classifying Y objects only. This architecture requires defining a novelty detection metric and using D_{train} to train convolutional and fully connected networks.

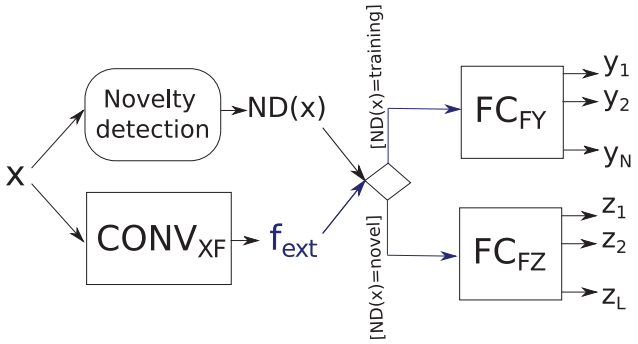


Figure 6.1: Overview of our framework: recognition of known and novel objects.

On one hand, we propose to use a Gaussian Mixture Model (GMM) as a novelty detection metric. We use GMM to estimate the density distribution of the tactile training data from D_{train}. A data sample x is classified as *novel* if it belongs to a region in the input space with low density, and as *training* otherwise. Formally, we compute a weighted log-likelihood of the fitted GMM given x: if it is lower than a threshold σ_{nov}, then x is *novel*.

On the other hand, D_{train} is the only training set we have to train the convolutional and fully connected networks in the framework. If we consider the CNN consisting of

$CONV_{XF}$ and FC_{FY}, it can directly be trained using D_{train} since it maps X into Y. However, a problem arises with training FC_{FZ} since D_{train} does not include any tactile data collected from Z. The next section presents how to proceed to train FC_{FZ} without collecting additional tactile data, other than D_{train}.

6.2.2 Training FC_{FZ}

Our solution to train FC_{FZ} without collecting any tactile data other than D_{train} is to generate synthetic training data for each $z \in Z$. This requires acquiring semantic information about objects. By learning the relationship between semantic and tactile spaces, synthetic tactile data can be generated for each object based on its semantic description.

Following the success of attribute-based ZSL, we choose attributes as a popular, efficient and intuitive semantic representation of objects. Let us consider the set of attributes $A = \{a_1, \ldots, a_M\}$. We describe each object $o \in O$ with a deterministic attribute vector $\mathbf{a^o} = (a^o_1, \ldots, a^o_M)$, where for each $m = 1, \ldots, M$: $a^o_m = 1$, if a_m is present for object o and $a^o_m = 0$ otherwise. Let us take the example of $O = \{$pencil, bottle, mug$\}$, and $A = \{$wooden, glass, porcelain, cylindrical, thin, concave$\}$. We can describe objects in O using the following attribute vectors: $\mathbf{a^{pencil}} = [1, 0, 0, 1, 1, 0]$, $\mathbf{a^{bottle}} = [0, 1, 0, 1, 0, 0]$ and $\mathbf{a^{mug}} = [0, 0, 1, 1, 0, 1]$.

Then, our solution for generating synthetic data for training FC_{FZ} is to learn a generator $G : A \longrightarrow F$ that generates a feature vector in F given an attribute vector in A. Once G is learned, G generates for each $z \in Z$ a set of feature vectors $f_{gen} = G(\mathbf{a^z})$ using $\mathbf{a^z}$. Finally, FC_{FZ} is trained on classifying f_{gen} as z.

6.3 Generating Synthetic Features for Novel Objects

6.3.1 Solution Overview

The following steps summarize our solution for training the generator G:

1. We train the CNN consisting of $CONV_{XF}$ and FC_{FY} on classifying objects in Y using D_{train};

2. We train a Deconvolutional Neural Network G on generating synthetic features $f_{gen} \in F$ using attribute vectors $\{\mathbf{a^{y_1}}, \ldots, \mathbf{a^{y_N}}\}$ describing training objects;

3. We improve the quality of G to generate features f_{gen} similar to those extracted from real tactile data f_{ext}.

4. We use the trained G to generate for each $z \in Z$ a set of I_z synthetic features $F_z = \{f_{gen,1}, \ldots f_{gen,I_z}\}$ using $\mathbf{a^z}$.

The next section details our proposed solution to perform each step.

6.3.2 Classifying Training Objects

The first step is to use D_{train} to train $CONV_{XF}$ and FC_{FY} to map tactile samples from X to Y classes. We remove from our framework illustrated in Fig. 6.1, the novelty detection and FC_{FZ} since we are working with Y only. This is equivalent to train the CNN illustrated in Fig. 6.2 using D_{train}.

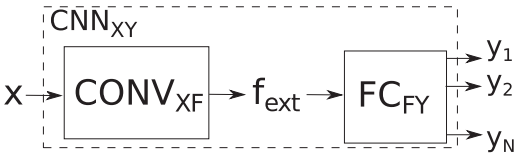


Figure 6.2: Classification of training objects using CNN_{XY}: $CONV_{XF}$ is the convolutional part and FC_{FY} is the fully connected part.

6.3.3 Training a Synthetic Features Generator

The second step is to train a Deconvolutional Neural Network G to generate synthetic features in F from attributes in A. As illustrated in Fig. 6.3, we use the pre-trained FC_{FY} to train G on how to generate features $f_{gen} \in F$ corresponding to attributes $\mathbf{a^{y_n}}$ from $D^a_{train} = \{(y_n, \mathbf{a^{y_n}}), n = 1, \ldots, N\}$. Here, FC_{FY} is not fine-tuned and thus its parameters are not updated. Training G is described in **Algorithm 1**: for each pair $(y_n, \mathbf{a^{y_n}})$ (line 3), $\mathbf{a^{y_n}}$ is input to G after adding random noise, to generate f_{gen} (lines 4, 5). The random noise serves to generate multiple feature vectors for the same attribute vector at different training epochs. The generated f_{gen} is input to FC_{FY} that classifies it as $y_{pred} \in Y$ (line 6). Actually, the goal of G is to generate from $\mathbf{a^{y_n}}$ features f_{gen} that are classified by FC_{FY} as y_n. This comes down to minimizing the loss L_{FC} between predicted y_{pred} and desired y_n, computed at line 7. The G parameters θ_G are updated using the Adam optimization algorithm [77], where $\partial L_{FC}/\partial \theta_G$ is the gradient of L_{FC} with respect to G parameters (line 8).

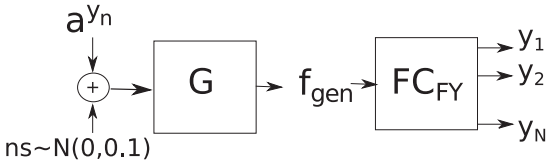


Figure 6.3: Train G to generate features associated with objects in Y.

Algorithm 1: Training G

Input: D^a_{train}, number of training epochs $epoch_{max}$, number of training objects N, pre-trained FC_{FY}

Output: Trained G

1: **for** $epoch_i = 1$ to $epoch_{max}$ **do**

2: **for** $n = 1$ to N **do**
3: $(\mathbf{a^{y_n}}, y_n) \leftarrow Next(D^a_{train})$ {Load the next pair}
4: $ns \sim \mathcal{N}(0,\ 0.1)$
5: $f_{gen} \leftarrow G(\mathbf{a^{y_n}} + ns)$
6: $y_{pred} \leftarrow FC_{FY}(f_{gen})$
7: $L_{FC} \leftarrow loss(y_n, y_{pred}))$
8: $\theta_G \leftarrow Update(\theta_G, \partial L_{FC}/\partial\theta_G)$
9: **end for**
10: **end for**
11: **return** G

6.3.4 Generating Realistic Features

Third, since our goal is to train FC_{FZ} using generated features and test it using real ones, we must improve the quality of the generated features to make them "as similar as possible" to the real ones. To this end, we continue training G by adding another convolutional network, called D, that discriminates between synthetic and real features. G and D are trained via an adversarial process illustrated in Fig. 6.4 and detailed in **Algorithm 2**: at each training iteration, we input noised $\mathbf{a^{y_n}}$ to G to obtain feature vector f_{gen} (lines 5-7) and a real tactile sample x to $CONV_{XF}$ to extract f_{ext} (lines 8, 9). Then, we train D and G alternately, such that we train D (resp. G) in Fig. 6.4a (resp. Fig. 6.4b) using G (resp. D) parameters obtained from the previous step and without fine-tuning G (resp. D). We start with training D (see Fig. 6.4a) on returning *'synthetic'* when inputting f_{gen} and *'real'* when inputting f_{ext} (lines 11-16). We keep fine-tuning D until its loss becomes lower than a certain threshold σ_D (lines 17-19). Then, we switch to fine-tuning G using the newly trained D and the pre-trained FC_{FY} simultaneously (see Fig. 6.4b). We update the G parameters, on one hand to minimize the loss between the desired y_n and predicted y_{pred} by inputting f_{gen} into FC_{FY} (lines 22-24). On the other hand, the updated G should generate f_{gen} that D erroneously classifies as *'real'* (lines 25-27). We keep training G until the losses of FC_{FY} and D become lower than a certain threshold σ_G (lines 28-30). Then, we go back to training D with the new updated G. We continue alternating between training D and G (Figures 6.4a and 6.4b, respectively). This adversarial training converges when D becomes unable to distinguish anymore between real and generated synthetic features. This means that G is generating synthetic features that are indistinguishable from real ones.

Algorithm 2: Adversarial training of G

Input: G trained using algorithm 1, D^a_{train}, D_{train}, pre-trained FC_{FY}, number of training epochs $epoch_{max}$, number of training objects N,σ_D: threshold of D training loss, σ_G: threshold of G training loss

Output: Trained G to output realistic features

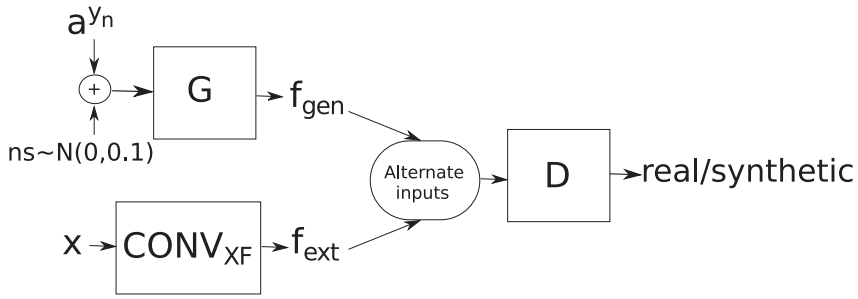
1: $train_G \leftarrow False$
2: $train_D \leftarrow True$ {Start with training D}

for $epoch_i = 1$ to $epoch_{max}$ **do**
  for $n = 1$ to N **do**
    $ns \sim \mathcal{N}(0,\, 0.1)$
    $(\mathbf{a^{y_n}}, y_n) \leftarrow Next(D^a_{train})$
    $f_{gen} \leftarrow G(\mathbf{a^{y_n}} + ns)$
    $x_i \leftarrow Next(D_{train})$
    $f_{ext} \leftarrow CONV_{XF}(x_i)$
    if $train_D$ **then**
      $d_{pred} \leftarrow D(f_{gen})$ {See Fig. 6.4a}
      $L_s \leftarrow loss(d_{pred}, 'synthetic')$
      $d_{pred} \leftarrow D(real\_feat)$
      $L_r \leftarrow loss(d_{pred}, 'real')$
      $L_D \leftarrow L_s + L_r$
      $\theta_D \leftarrow Update(\theta_D, \partial L_D / \partial \theta_D)$
      if $L_D/2 < \sigma_D$ **then**
        $train_G \leftarrow True$
        $train_D \leftarrow False$
      end if
    else if $train_G$ **then**
      $y_{pred} \leftarrow FC_{FY}(f_{gen})$ {See Fig. 6.4b}
      $L_{FC} \leftarrow loss(y_n, y_{pred})$
      $\theta_G \leftarrow Update(\theta_G, \partial L_{FC} / \partial \theta_G)$
      $d_{pred} \leftarrow D(f_{gen})$
      $L_D \leftarrow loss(d_{pred}, 'real')$
      $\theta_G \leftarrow Update(\theta_G, \partial L_D / \partial \theta_G)$
      if $L_{FC} < \sigma_G$ and $L_D < \sigma_G$ **then**
        $train_G \leftarrow False$
        $train_D \leftarrow True$
      end if
    end if
  end for
end for
return G

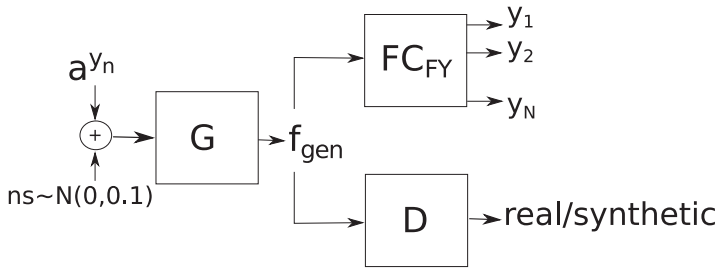
6.3.5 Generating Training Data for Novel Classes

As illustrated in Fig. 6.5, once G is trained, we use it to generate a set of I_z synthetic features for each $z \in Z$. We input its associated $\mathbf{a^z}$ to the trained G, I_z times with different noise values. This generates a set $F_z = \{f_1, \ldots, f_{I_z}\}$ that will be considered as the synthetic training set of z.

The last step is to use $D^Z_{train} = \{(z_l, F_{z_l}), l = 1, \ldots, L\}$ to train FC_{FZ}. We refer to this method as GEN-F. A variant of this method replaces the synthetic features by real ones collected from Y. Given F_{z_l} for each $z_l \in Z$, we generate another training set for z_l



(a) Train D to distinguish between real and generated features.



(b) Train G to generate synthetic features similar to real ones.

Figure 6.4: Adversarial training of G and D.

by replacing each f_i by its nearest neighbor (using L1 distance) in F space from features extracted from real training data. Thus, each z_l is trained using data collected from Y objects that are the most similar to F_{z_l} in F. We refer to this variant as GEN-NN-F.

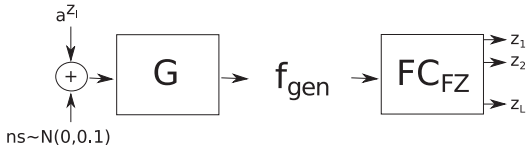


Figure 6.5: Train FC_{FZ} using data generated by G.

6.3.6 Extension to One-Shot Learning

Our framework, trained on real data for Y and on generated data only for Z, can integrate new real data for Z objects, which can become available with time. Here, we focus on the extreme case of OSL where one training sample arrives for each $z_l \in Z$. We obtain $D^Z_{train} = \{(z_l, x_l), l = 1, \ldots, L\}$. Directly integrating the only data sample available for each class is not expected to significantly improve the recognition performance, due to the tiny number of new samples. Instead, we use $CONV_{XF}$ to extract features of each x_l, yielding $f_l = CONV_{XF}(x_l)$. Then, we use the k nearest neighbors of each f_l in F to resume training FC_{FZ} for class z_l. In this case, each new sample for an object $z_l \in Z$ can improve training with k samples instead of only one sample.

6.4 Experimental Setup

6.4.1 Dataset

We evaluate our framework on the public PHAC-2 dataset [19] used by many state of art studies for tactile understanding [38, 96, 94]. This dataset contains 60 objects having a wide variety of texture, material and stiffness properties. Objects are described using a list of 24 binary *haptic adjectives*. After removing adjectives that are present in less than three objects, we obtained 19 adjectives that we use as attributes in this work: $A = \{absorbent, bumpy, compressible, cool, fuzzy, hard, hairy, metallic, porous, rough, scratchy, slippery, smooth, soft, solid, springy, squishy, textured, thick\}$.

Authors of [19] explored each one of the 60 objects 10 times (trials) using the gripper of the Willow Garage PR2 Robot equipped with 2 BioTac sensors. In this work, we use data collected from the pair of BioTacs and we do not consider the gripper kinesthetic data. This is because of the simple and similar shapes of PHAC-2 objects. BioTacs readings are pre-processed and data are augmented following the method of [38] that used BioTacs readings for binary classification of all attributes in A. This consists in first transforming BioTac signals measured from each exploration trial into a tactile image of 32 channels $\times$ 30 time samples. The 32 channels correspond to the 4 pressure and temperature BioTac readings along with the first 4 principal components (obtained by PCA) of the 19 BioTacs electrode signals, all measured during 4 EPs leading to $((4 + 4) \times 4 = 32$ channels). This defines the tactile data space $X = \mathbb{R}^{32\times30}$. By considering the 2 BioTacs as identical and after augmenting data by sub-sampling the signals using 5 different starting points, each exploration trial ensues 10 samples (2 BioTacs $\times$ 5 signals). Thus, we obtain a raw tactile dataset composed of 6000 samples (60 objects $\times$ 10 trials $\times$ 10 samples).

ZSL requires to split the 60 objects into 2 disjoint sets Z and Y. We randomly select 6 objects (10%) to be the test objects and the remaining 54 objects for Y (90%). In order to ensure the framework's robustness to the choice of Y and Z, we repeat this splitting process 7 times to generate different splits Z-Y. This avoids reporting results that are dependent on the choice of objects rather than on the design of the solution. Finally, we made sure that spaces F and A were correlated for each split.

6.4.2 Implementation Choices

In Table 6.1, we present the architecture of networks used in this work. Hyper-parameters were tuned to find a compromise between complexity and number of samples available to train each network. In our case, we have 100 samples for each object in Y, a number that does not allow us to train very complex models. FC_{FY} and FC_{FZ} are both one-layer fully connected networks. Convolutional Layers are followed by ReLU activation function for non-linearity. The weights of both the convolutional and fully connected layers are initialized using the Xavier method [39] and all deconvolutional layers are initialized using a Gaussian initializer. We used softmax function followed by multinomial logistic

loss to train the fully connected layers and cross-entropy loss to train D. We trained CNN_{XY} for 400 epochs and the adversarial training of G and D for $epoch_{max} = 600$. According to the architecture of $CONV_{XF}$, we obtained the features space $F = \mathbb{R}^{256\times 6}$, where 256 is the number of channels and (6×1) is the size of the output. Algorithms 1 and 2 are implemented by using batches of 50 samples.

Table 6.1: Neural Networks' hyper-parameters used in this work.

Neural Network	Type	layers	Convolutional layers parameters			
			channels	stride	kernel	group
$CONV_{XF}$	conv.	2	96-256	2-2	(3,1)-(3,1)	32
G	deconv.	2	96-256	2-1	(4,1)-(3,1)	No
D	conv.	2	96-1	2-2	(3,1)-(3,1)	No

All algorithms are implemented in Python and executed on a PC with an Intel(R) Core(TM) i7-3840QM 2.8 GHz processor and a 8 GB RAM. We exploited the available code[1] developed in [38] to process and extract features from the PHAC-2 database raw data. Python scikit-learn[2] was used to estimate the parameters of the GMM. We used Caffe [61] to implement all networks listed in Table 6.1. Finally, we used[3] [120] to implement Generative adversarial Networks (GAN) for the adversarial training of G and D. Finally, each convolutional network is trained for 600 epochs, which takes from 25 to 30 minutes, using the CPU only. Since we perform an offline recognition, we are not constrained by the training time, however, it can be optimized by using GPU.

6.5 Evaluation

6.5.1 Object Splits

First, we analyze the characteristic of objects used in our experiments. Fig. 6.6 illustrates some examples of PHAC-2 objects, their attributes, and the test objects of split 1. We note that, although test objects (framed in blue) are semantically different from training ones, both sets share the same attributes. For instance, the soap dispenser shares the attribute *smooth* with the koozie, all its attributes with the notepad and attribute *smooth* with the pool noodle. On the other hand, although Z objects share some attributes, each one of them has a discriminative attribute vector that distinguishes it from the other. For instance, the silicone block and the blue sponge are both *compressible* and *squishy*, yet, the first is *springy* while the second is *absorbent* and *soft*. The shared attributes between Z and Y

[1] people.eecs.berkeley.edu
[2] sikit-learn.org
[3] github.com/samson-wang/dcgan.caffe

objects and the uniqueness of the attribute vector of each Z object are verified for each split, which allows to perform ZSL using our framework.

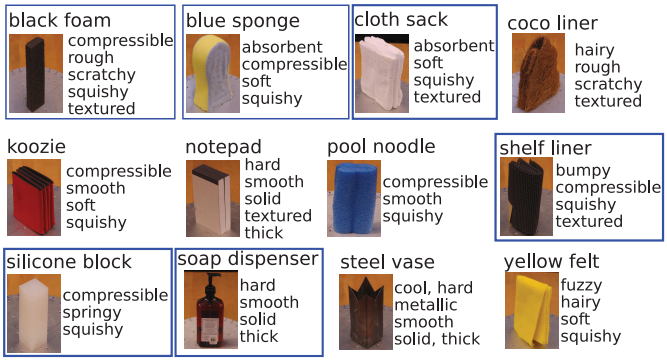


Figure 6.6: Test objects (framed in blue) with their attributes (right side) in Z for split 1 and examples of training objects with their attributes (right side).

6.5.2 Novelty Detection

Given a test sample $x \in X$, the first step is to estimate whether it is collected from a training or a novel object using a GMM. To tune σ_{nov}, we split Y into two disjoint sets Y_{tr} and Y_{val}: Y_{tr} contains 90% of the training objects (48 objects), while Y_{val} contains the remaining 6 objects (10%). Then, we split the data collected from Y_{tr} into X_{tr} and X_{te}: X_{tr} contains 90 samples (90%) per object and X_{te} contains the remaining 10 samples (10%) per class. First, X_{tr} is used to fit the GMM, then we tune σ_{nov} to maximize the accuracy of classifying X_{te} as collected from training objects and X_{val} (collected from Y_{val}) as novel. Fig. 6.7 shows that very low threshold values classify the majority of samples as known and very high values classify all the samples as novel. Thus, we choose for each split the σ_{nov} that maximizes the average accuracy of classifying X_{te} samples as known and X_{val} samples as novel. Once σ_{nov} is tuned for each split, we report in Table 6.2 the accuracies in classifying X_{te} as known, and both X_{val} and X_Z (collected from Z) as novel. By averaging accuracies over all splits, we found that 90.3% of x collected from training objects and not used to fit the GMM, have been classified as known, and that 89.5% of data collected from novel objects have been classified as novel.

6.5.3 Multi-class Classification of Known Objects

First, we focus on the part of the framework recognizing x in the case where it belongs to Y. We randomly select 10 samples from each $y \in Y$ and consider them as the test data, while the remaining 90 samples are used to train CNN_{XY}. We report in Table 6.3 the recognition accuracy that the framework can achieve when training data are available. We can see that the recognition accuracy is very high. This result is important because it has an impact on the training of $CONV_{XF}$ and thus also on recognizing novel objects. In

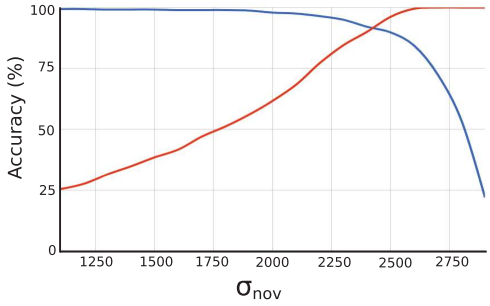


Figure 6.7: Tuning σ_{nov}: the accuracy of classifying X_{te} as known (blue) and X_{val} as novel (red) for split 1.

Table 6.2: Accuracy of novelty detection (%): distinction between known and novel objects.

Split	$x \in X_{te}$	$x \in X_{val}$	$x \in X_Z$
1	89.8	96.2	94.7
2	91.0	95.5	89.7
3	78.1	86.7	93.3
4	92.7	96.8	81.3
5	89.2	95.2	94.7
6	96.7	98.5	83
7	94.8	98	67.5
average	90.3	95.3	89.5

addition, it reveals the efficiency of our framework in classifying objects when BioTacs data are available.

Table 6.3: Recognition accuracies (%) for multi-class classification of Y with many training samples per object.

Split	1	2	3	4	5	6	7	avg.
90 samples	96.2	95.1	90.8	96.2	95.9	96.5	95.4	95.2

6.5.4 Evaluation of Synthetic Features Generation

Synthetic features are generated in order to train the recognition system if real training data are missing. Thus, what evaluates the quality of features generated using the algorithm presented in Sect. 6.3.3 is the accuracy of recognizing novel objects when training

the framework using synthetic features only, and testing it on real features.

6.5.4.1 Comparison with real features

In Table 6.4, we compare the recognition performance of test objects when real training features are available, and when they are replaced by synthetic training features. This can be achieved by training FC_{FZ} once using real features extracted from BioTac data, and once with synthetic features generated according to GEN-F using the attribute vectors.

Table 6.4: Recognition accuracies (%) for Multi-Class classification (real training data) and ZSL (synthetic training data) when training FC_{FZ} using 0, 10, 50, 90 or 100 samples per class.

Split	Training using real features				Training using synthetic features		
	0	10	50	90	10	50	100
1	17	88	97	98	36	34	35
2	17	95	100	100	24	22	23
3	17	95	95	100	20	10	10
4	17	97	98	100	36	38	37
5	17	68	85	100	32	34	33
6	17	70	88	100	33	34	33
7	17	67	77	83	35	31	33
average	17	83	91	97	31	29	29

There are several points to note from Table 6.4:

- One can directly notice that the recognition accuracies when training with real features, are significantly higher than when training with synthetic features. This is important to highlight that ZSL does not compete multi-class classification, but replaces it when training data are not available. In fact, BioTac readings are more efficient than attributes for the recognition. BioTacs give an average accuracy of 97% compared to 31% for attributes.

- The usefulness of ZSL can be shown by observing the performance of FC_{FZ} when no real data are available for any of Z objects (see the second column of the table). The classifier is not able to distinguish between objects and gives the accuracy of randomly classifying 6 objects. Therefore, only here comes the role of ZSL to improve this random accuracy by generating synthetic features.

- For all object splits, ZSL could improve the random classification accuracy. Generating synthetic features improved the recognition accuracy to 36% for splits 1 and 4, with an average accuracy of 31% for all splits.

- We notice, contrary to real features, increasing the number of generated training samples does not necessarily improve the recognition. We think that since all the synthetic features of each class are generated from the same attribute vector, by adding only a small noise value, then generating many features for each object may lead to over-fitting because of the similarity between the training samples.

6.5.4.2 Zero-Shot Learning

Here, we analyze the recognition performance of novel objects, for which there is no real training data. For each x collected from a novel object, its feature vector is classified using FC_{FZ}, which was trained using generated features. We compare in Table 6.5 the classification performance of the two methods GEN-F and GEN-NN-F using 10 generated training samples per class.

Table 6.5: Recognition accuracies ($\%$) for ZSL using GEN-F and GEN-NN-F.

Split	1	2	3	4	5	6	7	avg.
GEN-F	36	24	20	**36**	32	33	**35**	31
GEN-NN-F	**37**	**33**	**37**	34	**41**	**34**	**35**	**36**

Results show that GEN-NN-F outperforms GEN-F for almost all splits, with an improvement of 5% of the average accuracy of all splits. GEN-NN-F reaches an accuracy of 41% for recognizing Z objects of split 5, which is a considerable improvement compared to 17% obtained without generating the synthetic features.

Furthermore, to show the efficiency of our method, we compare it to another ones performing ZSL. First, we analyze the necessity of using the GAN-based setting, by skipping **Algorithm 2** and training the generator using **Algorithm 1** only. This means that the generator will be trained on generating synthetic features, not necessarily similar to real ones. Results reported in Table 6.6 show the significant drop in performance when removing the adversarial training from the learning algorithm. In fact, since FC_{FZ} is trained on generated features and tested on real ones, removing the GAN made the generated features very different from the real ones.

Table 6.6: Recognition accuracies ($\%$) for ZSL with GAN and without GAN.

Split	1	2	3	4	5	6	7	avg.
No-GAN	31	15	26	29	17	32	11	23
GAN	**37**	**33**	**37**	**34**	**41**	**34**	**35**	**36**

Second, we compare our ZSL framework to the only previous study on haptic ZSL [1]. Table 6.7 presents the comparison of recognition accuracies of all splits between

the framework of [1] and GEN-NN-F. The latter performs better than [1] for 4 of the 7 splits. However, both methods have the same average accuracy. Therefore, for ZSL, the two methods perform quite similarly. Yet, the improvement we make w.r.t [1] is the possibility of recognizing training and novel objects in the same framework, in addition to integrating new training data for a smooth transition to multi-class classification, which were not possible in [1].

Table 6.7: Recognition accuracies (%) for ZSL with the method of [1] and GEN-NN-F.

Split	1	2	3	4	5	6	7	avg.
[1]	23	20	32	**48**	**43**	**52**	32	**36**
GEN-NN-F	**37**	**33**	**37**	34	41	34	**35**	**36**

6.5.5 One-Shot Learning

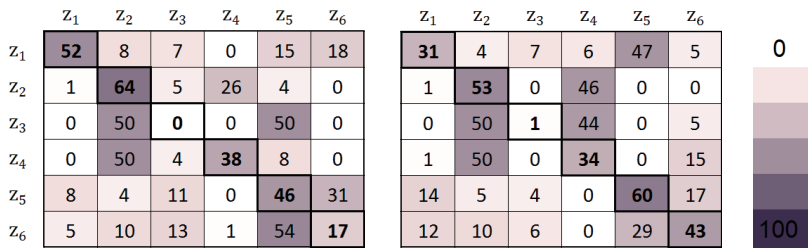
We use a single training sample for each $z \in Z$ to complete the training of FC_{FZ}, which was initially trained using GEN-NN-F (see Table 6.5). We report in Table 6.8 how performance is improved by integrating only one training sample per class (we have this sample since PHAC-2 dataset provides haptic data for all objects). We note that for most splits, performance is improved. For instance, adding one sample improved the accuracy of split 7 of 16%, and that of split 5 up to 55%. Overall, we obtain an average accuracy of 44% for all objects from all splits.

Table 6.8: Recognition accuracies (%) for OSL.

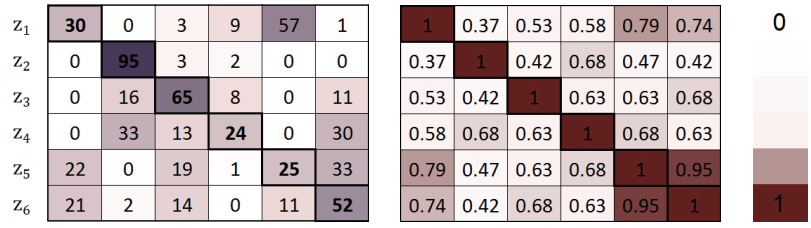
Split	1	2	3	4	5	6	7	avg.
GEN-NN-F	37	33	**37**	**34**	41	34	35	36
OSL	**49**	**41**	**37**	28	**55**	**49**	**51**	**44**

We illustrate in Fig. 6.8a, 6.8b and 6.8c the confusion matrices of ZSL and OSL classification results on split 1. In addition, we illustrate in Fig. 6.8d the similarity matrix of Z objects by computing the Jaccard distance between the attribute vectors of each pair of objects. Most misclassified objects are confused with objects that are close to them in attributes space. This typically happens to object z_2, that is mostly misclassified as z_4, which is its most similar object according to the similarity matrix. This is unexceptional since the training features have been generated from attributes, leading to close features generated from close attributes vectors.

Finally, we test the robustness of our ZSL and OSL recognition methods when reducing the number of training objects and increasing the number of novel ones. To this



(a) ZSL: GEN-F. (b) ZSL: GEN-NN-F.



(c) OSL. (d) similarity in attributes space

Figure 6.8: Confusion matrices and attribute-based similarity matrix of split 1.

end, we randomly redefine new object splits such that Y contains 48 objects (80%) and Z contains the remaining 12 objects (20%). Table 6.9 presents the recognition accuracies. As expected, the accuracies drop w.r.t. Tables 6.5 and 6.8, since we made the recognition task more challenging. However, the recognition accuracies are still significantly above the random accuracy (8% for classifying 12 novel objects) which would be obtained by any traditional recognition algorithm. From these results, we conclude that our method requires an important amount of training objects, in order to improve its ability to generalize the trained models for recognizing novel objects. Thus, the method cannot handle more challenging training/test splits such 70/30 or 60/40 since the amount of training data will make the learned model underfits the data. However, we cannot make a conclusion on the generalization capability of the method for a larger-scale setting, when hundreds of training objects are available. Unfortunately, PHAC-2 is probably the largest available dataset providing tactile data for objects, and building our own large-scale dataset is time consuming and is out of the scope of this work.

Table 6.9: ZSL and OSL recognition accuracies of 12 novel objects.

Split	1	2	3	4	5	6	7
GEN-F	17	19	10	17	14	14	13
GEN-NN-F	24	19	18	13	14	13	15
OSL	31	30	24	27	31	27	34

6.6 Conclusion

This chapter develops a recognition framework that is able to handle recognition of both known as well as novel objects. Results show the capacity of our framework to recognize objects having many training data (90 samples per class) with an average accuracy of 95% (Table 6.3), in addition to recognizing 6 objects having no training data with an average accuracy of 36% (Table 6.5), which was not possible using traditional training (Table 6.4). Furthermore, the framework efficiently integrates incoming data and reaches a high accuracy of multi-class classification when enough data become available with time (Table 6.8 for one sample, and Table 6.4 for many samples).

However, our framework still presents some limitations that can be a starting point for further improvements. First, recognition of novel objects is limited by the domain shift problem [35], and the correlation between attributes space and features space. Moreover, the set of novel classes that our framework can recognize must be known, and adding novel classes requires the modification of the output layer of FC_{FZ}. Similarly, the addition of new attributes requires the modification of the input layer of G. This can however be solved by utilizing classifiers that add new classes with a low cost, as in [104].

Bibliography

[1] ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. Haptic Zero-Shot Learning: Recognition of objects never touched before. *Robotics and Autonomous Systems 105* (2018), 11–25.

[2] ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. Visuo-tactile recognition of daily-life objects never seen or touched before. IEEE Int. Conf. on Control Automation Robotics & Vision (ICARCV), 2018.

[3] ABDERRAHMANE, Z., GANESH, G., CROSNIER, A., AND CHERUBINI, A. A deep learning framework for tactile recognition of known as well as novel objects. Transactions on Industrial Informatics (TII), 2019.

[4] ACER, M., YILDIZ, A. F., AND BAZZAZ, F. H. Development of a soft pzt based tactile sensor array for force localization. In *Information, Communication and Automation Technologies (ICAT), 2017 XXVI International Conference on* (2017), IEEE, pp. 1–6.

[5] ALLEN, P. K., AND ROBERTS, K. S. Haptic object recognition using a multi-fingered dextrous hand. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (1989), IEEE, pp. 342–347.

[6] ASFOUR, T., REGENSTEIN, K., AZAD, P., SCHRODER, J., BIERBAUM, A., VAHRENKAMP, N., AND DILLMANN, R. Armar-iii: An integrated humanoid platform for sensory-motor control. In *Humanoid Robots, 2006 6th IEEE-RAS International Conference on* (2006), IEEE, pp. 169–175.

[7] BEKIROGLU, Y., KRAGIC, D., AND KYRKI, V. Learning grasp stability based on tactile data and hmms. In *19th International Symposium in Robot and Human Interactive Communication* (2010), IEEE, pp. 132–137.

[8] BELL, S., UPCHURCH, P., SNAVELY, N., AND BALA, K. Material recognition in the wild with the materials in context database (supplemental material). In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2015).

[9] BELLMAN, R. Dynamic programming (dp).

[10] BESL, P. J., AND MCKAY, N. D. Method for registration of 3-d shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures* (1992), vol. 1611, International Society for Optics and Photonics, pp. 586–607.

[11] BIEDERMAN, I. Recognition-by-components: a theory of human image understanding. *Psychological Review 94*, 2 (1987), 115.

[12] BIERBAUM, A., ASFOUR, T., AND DILLMANN, R. Dynamic potential fields for dexterous tactile exploration. In *Human Centered Robot Systems*. Springer, 2009, pp. 23–31.

[13] BUDA, M., MAKI, A., AND MAZUROWSKI, M. A. A systematic study of the class imbalance problem in convolutional neural networks. *CoRR, abs/1710.05381* (2017).

[14] CALLI, B., WALSMAN, A., SINGH, A., SRINIVASA, S., ABBEEL, P., AND DOLLAR, A. M. Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols. *arXiv preprint arXiv:1502.03143* (2015).

[15] CASELLI, S., MAGNANINI, C., AND ZANICHELLI, F. Haptic object recognition with a dextrous hand based on volumetric shape representations. In *IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems* (1994), IEEE, pp. 280–287.

[16] CHANGPINYO, S., CHAO, W.-L., AND SHA, F. Predicting visual exemplars of unseen classes for zero-shot learning. In *IEEE Int. Conf. on Computer Vision (ICCV)* (2017), pp. 3496–3505.

[17] CHENG, Y., QIAO, X., WANG, X., AND YU, Q. Random forest classifier for zero-shot learning based on relative attribute. *IEEE Transactions on Neural Networks and Learning Systems* (2017).

[18] CHOI, Y. S., DEYLE, T., CHEN, T., GLASS, J. D., AND KEMP, C. C. A list of household objects for robotic retrieval prioritized by people with als. In *IEEE Int. Conf. on Rehabilitation Robotics* (2009), IEEE, pp. 510–517.

[19] CHU, V., MCMAHON, I., RIANO, L., MCDONALD, C. G., HE, Q., PEREZ-TEJADA, J. M., ARRIGO, M., DARRELL, T., AND KUCHENBECKER, K. J. Robotic learning of haptic adjectives through physical interaction. *Robotics and Autonomous Systems 63* (2015), 279–292.

[20] CHU, V., MCMAHON, I., RIANO, L., MCDONALD, C. G., HE, Q., PEREZ-TEJADA, J. M., ARRIGO, M., FITTER, N., NAPPO, J. C., DARRELL, T., ET AL. Using robotic exploratory procedures to learn the meaning of haptic adjectives. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2013), IEEE, pp. 3048–3055.

[21] CONTROZZI, M., CIPRIANI, C., AND CARROZZA, M. C. Design of artificial hands: A review. In *The Human Hand as an Inspiration for Robot Hand Development*. Springer, 2014, pp. 219–246.

[22] CORRADI, T., HALL, P., AND IRAVANI, P. Tactile features: recognising touch sensations with a novel and inexpensive tactile sensor. In *Conference Towards Autonomous Robotic Systems* (2014), Springer, pp. 163–172.

[23] CORRADI, T., HALL, P., AND IRAVANI, P. Bayesian tactile object recognition: learning and recognising objects using a new inexpensive tactile sensor. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2015), IEEE, pp. 3909–3914.

[24] DE OLIVERIA, T., CRETU, A.-M., AND PETRIU, E. M. Multimodal bio-inspired tactile sensing module. *IEEE Sens. J 17*, 11 (2017), 1–1.

[25] DENG, J., DONG, W., SOCHER, R., LI, L.-J., LI, K., AND FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2009), IEEE, pp. 248–255.

[26] DRIMUS, A., KOOTSTRA, G., BILBERG, A., AND KRAGIC, D. Design of a flexible tactile sensor for classification of rigid and deformable objects. *Robotics and Autonomous Systems 62*, 1 (2014), 3–15.

[27] ELHOSEINY, M., SALEH, B., AND ELGAMMAL, A. Write a classifier: Zero-shot learning using purely textual descriptions. In *IEEE Int. Conf. on Computer Vision (ICCV)* (2013), pp. 2584–2591.

[28] EVERINGHAM, M., VAN GOOL, L., WILLIAMS, C. K., WINN, J., AND ZISSERMAN, A. The pascal visual object classes (voc) challenge. *International journal of computer vision 88*, 2 (2010), 303–338.

[29] FALCO, P., LU, S., CIRILLO, A., NATALE, C., PIROZZI, S., AND LEE, D. Cross-modal visuo-tactile object recognition using robotic active exploration. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2017), pp. 5273–5280.

[30] FARHADI, A., ENDRES, I., HOIEM, D., AND FORSYTH, D. Describing objects by their attributes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2009), pp. 1778–1785.

[31] FE-FEI, L., ET AL. A bayesian approach to unsupervised one-shot learning of object categories. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* (2003), IEEE, pp. 1134–1141.

[32] FEI-FEI, L., FERGUS, R., AND PERONA, P. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence 28*, 4 (2006), 594–611.

[33] FEI-FEI, L., FERGUS, R., AND PERONA, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding 106*, 1 (2007), 59–70.

[34] FISHEL, J., LIN, G., AND LOEB, G. Biotac® product manual. *SynTouch LLC, February* (2013).

[35] FU, Y., HOSPEDALES, T. M., XIANG, T., AND GONG, S. Transductive multi-view zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence 37*, 11 (2015), 2332–2345.

[36] FU, Z., XIANG, T., KODIROV, E., AND GONG, S. Zero-shot object recognition by semantic manifold distance. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 2635–2644.

[37] GANDARIAS, J. M., GÓMEZ-DE GABRIEL, J. M., AND GARCÍA-CEREZO, A. Human and object recognition with a high-resolution tactile sensor. In *SENSORS* (2017), IEEE, pp. 1–3.

[38] GAO, Y., HENDRICKS, L. A., KUCHENBECKER, K. J., AND DARRELL, T. Deep learning for tactile understanding from visual and haptic data. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2016), IEEE, pp. 536–543.

[39] GLOROT, X., AND BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In *Int. Conf. on Artificial Intelligence and Statistics* (2010), pp. 249–256.

[40] GOGER, D., GORGES, N., AND WORN, H. Tactile sensing for an anthropomorphic robotic hand: Hardware and signal processing. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2009), IEEE, pp. 895–901.

[41] GORDON, A. M., AND SOECHTING, J. F. Use of tactile afferent information in sequential finger movements. *Experimental brain research 107*, 2 (1995), 281–292.

[42] GORGES, N., NAVARRO, S. E., GÖGER, D., AND WÖRN, H. Haptic object recognition using passive joints and haptic key features. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2010), IEEE, pp. 2349–2355.

[43] GORGES, N., NAVARRO, S. E., AND WÖRN, H. Haptic object recognition using statistical point cloud features. In *IEEE International Conference on Advanced Robotics (ICAR)* (2011), IEEE, pp. 15–20.

[44] GRAUMAN, K., AND LEIBE, B. Visual object recognition. *Synthesis lectures on artificial intelligence and machine learning 5*, 2 (2011), 1–181.

[45] GRIFFIN, G., HOLUB, A., AND PERONA, P. Caltech-256 object category dataset.

[46] GRILL-SPECTOR, K., AND KANWISHER, N. Visual recognition: As soon as you know it is there, you know what it is. *Psychological Science 16*, 2 (2005), 152–160.

[47] GROSSBERG, S. Recurrent neural networks. *Scholarpedia 8*, 2 (2013), 1888.

[48] GU, H., FAN, S., ZONG, H., JIN, M., AND LIU, H. Haptic perception of unknown object by robot hand: Exploration strategy and recognition approach. *International Journal of Humanoid Robotics 13*, 03 (2016), 1650008.

[49] GÜLER, P., BEKIROGLU, Y., GRATAL, X., PAUWELS, K., AND KRAGIC, D. What's in the container? classifying object contents from vision and touch. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on* (2014), IEEE, pp. 3961–3968.

[50] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 770–778.

[51] HEARST, M. A., DUMAIS, S. T., OSUNA, E., PLATT, J., AND SCHOLKOPF, B. Support vector machines. *IEEE Intelligent Systems and their applications 13*, 4 (1998), 18–28.

[52] HENSMAN, P., AND MASKO, D. The impact of imbalanced training data for convolutional neural networks. *Degree Project in Computer Science, KTH Royal Institute of Technology* (2015).

[53] HOELSCHER, J., PETERS, J., AND HERMANS, T. Evaluation of tactile feature extraction for interactive object recognition. In *IEEE-RAS Int. Conf. on Humanoid Robots* (2015), pp. 310–317.

[54] HOSODA, K., AND IWASE, T. Robust haptic recognition by anthropomorphic bionic hand through dynamic interaction. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on* (2010), IEEE, pp. 1236–1241.

[55] HSIAO, P.-H., CHANG, F.-J., AND LIN, Y.-Y. Learning discriminatively reconstructed source data for object recognition with few examples. *IEEE Transactions on Image Processing 25*, 8 (2016), 3518–3532.

[56] IWATA, H., AND SUGANO, S. Design of human symbiotic robot twendy-one. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2009), IEEE, pp. 580–586.

[57] JACOBSEN, S., IVERSEN, E., KNUTTI, D., JOHNSON, R., AND BIGGERS, K. Design of the utah/mit dextrous hand. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (1986), vol. 3, IEEE, pp. 1520–1532.

[58] JANOCH, A., KARAYEV, S., JIA, Y., BARRON, J. T., FRITZ, M., SAENKO, K., AND DARRELL, T. A category-level 3d object dataset: Putting the kinect to work. In *Consumer Depth Cameras for Computer Vision*. Springer, 2013, pp. 141–165.

[59] JAYARAMAN, D., AND GRAUMAN, K. Zero-shot recognition with unreliable attributes. In *Advances in Neural Information Processing Systems* (2014), pp. 3464–3472.

[60] JI, Y., YANG, Y., XU, X., AND SHEN, H. T. One-shot learning based pattern transition map for action early recognition. *Signal Processing 143* (2018), 364–370.

[61] JIA, Y., SHELHAMER, E., DONAHUE, J., KARAYEV, S., LONG, J., GIRSHICK, R., GUADARRAMA, S., AND DARRELL, T. Caffe: Convolutional architecture for fast feature embedding. In *ACM Int. Conf. on Multimedia* (2014), ACM, pp. 675–678.

[62] JIN, M., GU, H., FAN, S., ZHANG, Y., AND LIU, H. Object shape recognition approach for sparse point clouds from tactile exploration. In *IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)* (2013), IEEE, pp. 558–562.

[63] JOHANSSON, R., AND WESTLING, G. Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects. *Experimental brain research 56*, 3 (1984), 550–564.

[64] JOHNSSON, M., AND BALKENIUS, C. Experiments with proprioception in a self-organizing system for haptic perception. *Towards Autonomous Robotic Systems* (2007), 239–245.

[65] JOHNSSON, M., AND BALKENIUS, C. Haptic perception with self-organizing anns and an anthropomorphic robot hand. *Journal of Robotics 2010* (2010).

[66] JOLLIFFE, I. Principal component analysis. In *International encyclopedia of statistical science*. Springer, 2011, pp. 1094–1096.

[67] KABOLI, M., AND CHENG, G. Novel tactile descriptors and a tactile transfer learning technique for active in-hand object recognition via texture properties. In *IEE-RAS Int. Conf. on Humanoid Robots-Workshop Tactile Sensing for Manipulation: New Progress and Challenges* (2016).

[68] KABOLI, M., WALKER, R., AND CHENG, G. Re-using prior tactile experience by robotic hands to discriminate in-hand objects via texture properties. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2016), pp. 2242–2247.

[69] KABOLI, M., WALKER, R., CHENG, G., ET AL. In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors. In *IEEE-RAS Int. Conf. on Humanoid Robots* (2015), pp. 1155–1160.

[70] KANKUEKUL, P., KAWEWONG, A., TANGRUAMSUB, S., AND HASEGAWA, O. Online incremental attribute-based zero-shot learning. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2012), IEEE, pp. 3657–3664.

[71] KAPPASSOV, Z., CORRALES, J.-A., AND PERDEREAU, V. Tactile sensing in dexterous robot hands. *Robotics and Autonomous Systems 74* (2015), 195–220.

[72] KEMP, C., TENENBAUM, J. B., GRIFFITHS, T. L., YAMADA, T., AND UEDA, N. Learning systems of concepts with an infinite relational model. In *AAAI* (2006), vol. 3, p. 5.

[73] KEOGH, E., AND MUEEN, A. Curse of dimensionality. In *Encyclopedia of Machine Learning and Data Mining*. Springer, 2017, pp. 314–315.

[74] KERR, E., MCGINNITY, T. M., AND COLEMAN, S. Material classification based on thermal and surface texture properties evaluated against human performance. In *IEEE Int. Conf. on Control Automation Robotics & Vision (ICARCV)* (2014), pp. 444–449.

[75] KG, S. G. . C. Schunk dexterous hand.

[76] KIM, J. K., WEE, J. W., AND LEE, C. H. Sensor fusion system for improving the recognition of 3d object. In *Cybernetics and Intelligent Systems, 2004 IEEE Conference on* (2004), vol. 2, IEEE, pp. 1207–1212.

[77] KINGMA, D. P., AND BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[78] KLATZKY, R., AND LEDERMAN, S. Human haptics. In *New encyclopedia of neuroscience*, vol. 5. Elsevier Amsterdam, 2009, pp. 11–18.

[79] KLATZKY, R. L., LEDERMAN, S. J., AND MATULA, D. E. Haptic exploration in the presence of vision. *Journal of Experimental Psychology: Human Perception and Performance 19*, 4 (1993), 726.

[80] KLATZKY, R. L., LEDERMAN, S. J., AND REED, C. There's more to touch than meets the eye: The salience of object attributes for haptics with and without vision. *Journal of experimental psychology: general 116*, 4 (1987), 356.

[81] KOHONEN, T. The self-organizing map. *Proceedings of the IEEE 78*, 9 (1990), 1464–1480.

[82] KRIZHEVSKY, A. Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto,, 2009.

[83] KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

[84] KUMAR, N., BERG, A. C., BELHUMEUR, P. N., AND NAYAR, S. K. Attribute and simile classifiers for face verification. In *IEEE Int. Conf. on Computer Vision (ICCV)* (2009), IEEE, pp. 365–372.

[85] LAMPERT, C. H., NICKISCH, H., AND HARMELING, S. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2009), pp. 951–958.

[86] LECUN, Y., BOTTOU, L., BENGIO, Y., AND HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324.

[87] LEDERMAN, S. J., AND KLATZKY, R. L. Hand movements: A window into haptic object recognition. *Cognitive Psychology 19*, 3 (1987), 342–368.

[88] LEDERMAN, S. J., AND KLATZKY, R. L. Extracting object properties through haptic exploration. *Acta psychologica 84*, 1 (1993), 29–40.

[89] LEDERMAN, S. J., AND KLATZKY, R. L. Haptic perception: A tutorial. *Attention, Perception, & Psychophysics 71*, 7 (2009), 1439–1459.

[90] LEI BA, J., SWERSKY, K., FIDLER, S., ET AL. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *IEEE Int. Conf. on Computer Vision (ICCV)* (2015), pp. 4247–4255.

[91] LIAROKAPIS, M. V., CALLI, B., SPIERS, A. J., AND DOLLAR, A. M. Unplanned, model-free, single grasp object classification with underactuated hands and force sensors. In *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems (IROS)* (2015), pp. 5073–5080.

[92] LIU, H., GUO, D., AND SUN, F. Object recognition using tactile measurements: Kernel sparse coding methods. *IEEE Trans. on Instrumentation and Measurement 65*, 3 (2016), 656–665.

[93] LIU, H., SUN, F., FANG, B., AND GUO, D. Cross-modal zero-shot-learning for tactile object recognition. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2018), 1–9.

[94] LIU, H., SUN, F., GUO, D., FANG, B., AND PENG, Z. Structured output-associated dictionary learning for haptic understanding. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2017).

[95] LIU, H., WU, Y., SUN, F., FANG, B., AND GUO, D. Weakly paired multimodal fusion for object recognition. *IEEE Transactions on Automation Science and Engineering 15*, 2 (2018), 784–795.

[96] LIU, H., WU, Y., SUN, F., GUO, D., AND FANG, B. Multi-label tactile property analysis. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2017), pp. 366–371.

[97] LUO, S., MOU, W., ALTHOEFER, K., AND LIU, H. Novel tactile-sift descriptor for object shape recognition. *IEEE Sensors Journal 15*, 9 (2015), 5001–5009.

[98] MADRY, M., BO, L., KRAGIC, D., AND FOX, D. St-hmp: Unsupervised spatio-temporal feature learning for tactile data. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2014), IEEE, pp. 2262–2269.

[99] MARTINEZ-HERNANDEZ, U., LEPORA, N. F., AND PRESCOTT, T. J. Active haptic shape recognition by intrinsic motivation with a robot hand. In *IEEE World Haptics Conference (WHC)* (2015), IEEE, pp. 299–304.

[100] MATHEUS, K., AND DOLLAR, A. M. Benchmarking grasping and manipulation: properties of the objects of daily living. In *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems (IROS)* (2010), IEEE, pp. 5020–5027.

[101] MEIER, M., SCHOPFER, M., HASCHKE, R., AND RITTER, H. A probabilistic approach to tactile shape reconstruction. *IEEE Transactions on Robotics 27*, 3 (2011), 630–635.

[102] MELCHIORRI, C., AND VASSURA, G. Mechanical and control features of the ub hand version 2. In *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'92* (1992), pp. 7–10.

[103] MENSINK, T., GAVVES, E., AND SNOEK, C. G. Costa: Co-occurrence statistics for zero-shot classification. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2014), pp. 2441–2448.

[104] MENSINK, T., VERBEEK, J., PERRONNIN, F., AND CSURKA, G. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 11 (2013), 2624–2637.

[105] METTA, G., SANDINI, G., VERNON, D., NATALE, L., AND NORI, F. The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th workshop on performance metrics for intelligent systems* (2008), ACM, pp. 50–56.

[106] MILLER, G. A. Wordnet: a lexical database for english. *Communications of the ACM 38*, 11 (1995), 39–41.

[107] MISHRA, A., REDDY, M., MITTAL, A., AND MURTHY, H. A. A generative model for zero shot learning using conditional variational autoencoders. *arXiv preprint arXiv:1709.00663* (2017).

[108] MITTENDORFER, P., YOSHIDA, E., AND CHENG, G. Realizing whole-body tactile interactions with a self-organizing, multi-modal artificial skin on a humanoid robot. *Advanced Robotics 29*, 1 (2015), 51–67.

[109] NAIR, V., AND HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (2010), pp. 807–814.

[110] NAVARRO, S. E., GORGES, N., WÖRN, H., SCHILL, J., ASFOUR, T., AND DILLMANN, R. Haptic object recognition for multi-fingered robot hands. In *IEEE Haptics Symposium (HAPTICS)* (2012), IEEE, pp. 497–502.

[111] NILSBACK, M.-E., AND ZISSERMAN, A. Automated flower classification over a large number of classes. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on* (2008), IEEE, pp. 722–729.

[112] ORII, H., TSUJI, S., KOUDA, T., AND KOHAMA, T. Tactile texture recognition using convolutional neural networks for time-series data of pressure and 6-axis acceleration sensor. In *Int. Conf. on Industrial Technology (ICIT)* (2017), IEEE, pp. 1076–1080.

[113] OSHERSON, D. N., STERN, J., WILKIE, O., STOB, M., AND SMITH, E. E. Default probability. *Cognitive Science 15*, 2 (1991), 251–269.

[114] PALATUCCI, M., AND MITCHELL, T. M. Classification in very high dimensional problems with handfuls of examples. In *European Conference on Principles of Data Mining and Knowledge Discovery* (2007), Springer, pp. 212–223.

[115] PALATUCCI, M., POMERLEAU, D., HINTON, G. E., AND MITCHELL, T. M. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems* (2009), pp. 1410–1418.

[116] PARIKH, D., AND GRAUMAN, K. Relative attributes. In *IEEE Int. Conf. on Computer Vision (ICCV)* (2011), pp. 503–510.

[117] PATTERSON, G., AND HAYS, J. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2012), IEEE, pp. 2751–2758.

[118] PEZZEMENTI, Z., PLAKU, E., REYDA, C., AND HAGER, G. D. Tactile-object recognition from appearance information. *IEEE Transactions on Robotics 27*, 3 (2011), 473–487.

[119] RABIN, E., AND GORDON, A. M. Tactile feedback contributes to consistency of finger movements during typing. *Experimental brain research 155*, 3 (2004), 362–369.

[120] RADFORD, A., METZ, L., AND CHINTALA, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).

[121] READ, J., PFAHRINGER, B., HOLMES, G., AND FRANK, E. Classifier chains for multi-label classification. *Machine Learning 85*, 3 (2011), 333–359.

[122] REED, S., AKATA, Z., LEE, H., AND SCHIELE, B. Learning deep representations of fine-grained visual descriptions. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 49–58.

[123] REGOLI, M., JAMALI, N., METTA, G., AND NATALE, L. Controlled tactile exploration and haptic object recognition. In *IEEE Int. Conf. on Advanced Robotics (ICAR)* (2017), IEEE, pp. 47–54.

[124] REYNOLDS, D. Gaussian mixture models. *Encyclopedia of biometrics* (2015), 827–832.

[125] RODNER, E., AND DENZLER, J. Learning with few examples by transferring feature relevance. In *Joint Pattern Recognition Symposium* (2009), Springer, pp. 252–261.

[126] ROHRBACH, M., STARK, M., AND SCHIELE, B. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2011), IEEE, pp. 1641–1648.

[127] ROHRBACH, M., STARK, M., SZARVAS, G., GUREVYCH, I., AND SCHIELE, B. What helps where–and why? semantic relatedness for knowledge transfer. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2010), IEEE, pp. 910–917.

[128] ROMERA-PAREDES, B., AND TORR, P. An embarrassingly simple approach to zero-shot learning. In *Int. Conf. on Machine Learning* (2015), pp. 2152–2161.

[129] RUSSAKOVSKY, O., AND LI, F.-F. Attribute learning in large-scale datasets. In *European Conf. on Computer Vision (ECCV)* (2010), vol. 6553, pp. 1–14.

[130] RUSSELL, B. C., TORRALBA, A., MURPHY, K. P., AND FREEMAN, W. T. Labelme: a database and web-based tool for image annotation. *International journal of computer vision 77*, 1-3 (2008), 157–173.

[131] RUSSELL, R. A. Object recognition by asmarttactile sensor. In *Proceedings of the Australian Conference on Robotics and Automation* (2000), pp. 93–8.

[132] SALAKHUTDINOV, R., TENENBAUM, J., AND TORRALBA, A. One-shot learning with a hierarchical nonparametric bayesian model. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning* (2012), pp. 195–206.

[133] SANCHEZ-FIBLA, M., DUFF, A., AND VERSCHURE, P. F. A sensorimotor account of visual and tactile integration for object categorization and grasping. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2013), IEEE, pp. 107–112.

[134] SCHMITZ, A., BANSHO, Y., NODA, K., IWATA, H., OGATA, T., AND SUGANO, S. Tactile object recognition using deep learning and dropout. In *IEEE-RAS Int. Conf. on Humanoid Robots* (2014), pp. 1044–1050.

[135] SCHNEIDER, A., STURM, J., STACHNISS, C., REISERT, M., BURKHARDT, H., AND BURGARD, W. Object identification with tactile sensors using bag-of-features. In *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems (IROS)* (2009), IEEE, pp. 243–248.

[136] SCHOPFER, M., RITTER, H., AND HEIDEMANN, G. Acquisition and application of a tactile database. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2007), IEEE, pp. 1517–1522.

[137] SCHULZ, S., PYLATIUK, C., KARGOV, A., OBERLE, R., AND BRETTHAUER, G. Progress in the development of anthropomorphic fluidic hands for a humanoid robot. In *Humanoid Robots, 2004 4th IEEE/RAS International Conference on* (2004), vol. 2, IEEE, pp. 566–575.

[138] SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[139] SNELL, J., SWERSKY, K., AND ZEMEL, R. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems* (2017), pp. 4080–4090.

[140] SOCHER, R., GANJOO, M., MANNING, C. D., AND NG, A. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems* (2013), pp. 935–943.

[141] SOH, H., AND DEMIRIS, Y. Incrementally learning objects by touch: Online discriminative and generative models for tactile-based recognition. *IEEE Transactions on Haptics 7*, 4 (2014), 512–525.

[142] SOH, H., SU, Y., AND DEMIRIS, Y. Online spatio-temporal gaussian process experts with application to tactile classification. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012), IEEE, pp. 4489–4496.

[143] SPIERS, A. J., LIAROKAPIS, M. V., CALLI, B., AND DOLLAR, A. M. Single-grasp object classification and feature extraction with simple robot hands and tactile sensors. *IEEE Trans. on Haptics 9*, 2 (2016), 207–220.

[144] SU, Z., FISHEL, J. A., YAMAMOTO, T., AND LOEB, G. E. Use of tactile feedback to control exploratory movements to characterize object compliance. *Frontiers in Neurorobotics 6* (2012).

[145] SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S., ANGUELOV, D., ERHAN, D., VANHOUCKE, V., RABINOVICH, A., ET AL. Going deeper with convolutions. Cvpr.

[146] TANAKA, D., MATSUBARA, T., ICHIEN, K., AND SUGIMOTO, K. Object manifold learning with action features for active tactile object recognition. In *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems (IROS)* (2014), pp. 608–614.

[147] TOMMASI, T., ORABONA, F., AND CAPUTO, B. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 3081–3088.

[148] TURK, A. M. Amazon mechanical turk. *Retrieved August 17* (2012), 2012.

[149] WAH, C., BRANSON, S., WELINDER, P., PERONA, P., AND BELONGIE, S. The caltech-ucsd birds-200-2011 dataset.

[150] WANG, D., LI, Y., LIN, Y., AND ZHUANG, Y. Relational knowledge transfer for zero-shot learning. In *AAAI* (2016), vol. 2, p. 7.

[151] WARRINGTON, E. K., AND TAYLOR, A. M. Two categorical stages of object recognition. *Perception 7*, 6 (1978), 695–705.

[152] WELINDER, P., BRANSON, S., MITA, T., WAH, C., SCHROFF, F., BELONGIE, S., AND PERONA, P. Caltech-ucsd birds 200.

[153] WELINDER, P., BRANSON, S., MITA, T., WAH, C., SCHROFF, F., BELONGIE, S., AND PERONA, P. Caltech-UCSD Birds 200. Tech. Rep. CNS-TR-2010-001, California Institute of Technology, 2010.

[154] WINGERT, J. R., BURTON, H., SINCLAIR, R. J., BRUNSTROM, J. E., AND DAMIANO, D. L. Tactile sensory abilities in cerebral palsy: deficits in roughness and object discrimination. *Developmental Medicine & Child Neurology 50*, 11 (2008), 832–838.

[155] WOLF, L., AND MARTIN, I. Robust boosting for learning from few examples. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (2005), vol. 1, IEEE, pp. 359–364.

[156] WU, H., MIAO, Z., WANG, Y., AND LIN, M. Optimized recognition with few instances based on semantic distance. *The Visual Computer 31*, 4 (2015), 367–375.

[157] XIAN, Y., SCHIELE, B., AND AKATA, Z. Zero-shot learning - the good, the bad and the ugly. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (July 2017).

[158] XIAO, J., HAYS, J., EHINGER, K. A., OLIVA, A., AND TORRALBA, A. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on* (2010), IEEE, pp. 3485–3492.

[159] XU, D., LOEB, G. E., AND FISHEL, J. A. Tactile identification of objects using bayesian exploration. In *IEEE Int. Conf. on Robotics and Automation (ICRA)* (2013), pp. 3056–3061.

[160] YANG, J., LIU, H., SUN, F., AND GAO, M. Tactile sequence classification using joint kernel sparse coding. In *Int. Joint Conf. on Neural Networks (IJCNN)* (2015), IEEE, pp. 1–6.

[161] YU, F. X., CAO, L., FERIS, R. S., SMITH, J. R., AND CHANG, S.-F. Designing category-level attributes for discriminative visual recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2013), pp. 771–778.

[162] YU, X., AND ALOIMONOS, Y. Attribute-based transfer learning for object categorization with zero/one training example. *European Conf. on Computer Vision (ECCV)* (2010), 127–140.

[163] ZHANG, L., ZHANG, S., JIANG, F., QI, Y., ZHANG, J., GUO, Y., AND ZHOU, H. Bomw: Bag of manifold words for one-shot learning gesture recognition from kinect. *IEEE Transactions on Circuits and Systems for Video Technology* (2017).

[164] ZHANG, Y., JIN, R., AND ZHOU, Z.-H. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics 1*, 1-4 (2010), 43–52.

[165] ZHENG, H., FANG, L., JI, M., STRESE, M., ÖZER, Y., AND STEINBACH, E. Deep learning for surface material classification using haptic and visual information. *IEEE Transactions on Multimedia 18*, 12 (2016), 2407–2416.