Learning and Transferring Implicit Knowledge of Object Properties Across Robots via Interactive Perception

A dissertation

submitted by

Gyan Tatiya

In partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in

Computer Science

TUFTS UNIVERSITY

February 2024

Thesis Committee:

Prof. Jivko Sinapov, Tufts University (Chair)

Prof. Elaine Short, Tufts University

Prof. Robert Jacob, Tufts University

Prof. Jason Rife, Tufts University

Dr. Jonathan Francis, Bosch Center for Artificial Intelligence

Abstract

Gyan Tatiya

ADVISOR: Prof. Jivko Sinapov, Tufts University

Integrating personal and service robots into real-world environments necessitates a deep understanding of complex object properties beyond visual attributes. The field of robotics has showcased the capabilities of robots in executing interactive perception tasks involving physical interactions with objects and the acquisition of implicit object properties through non-visual sensory signals such as audio, tactile, and haptic feedback. While learning-based approaches for interactive perception tasks have yielded success, a critical limitation is the time-intensive nature of individual robot learning, impeding the deployment of algorithms across large collections of robots with subtly different morphologies, as in warehouse or factory automation settings. To address this challenge, this dissertation introduces innovative frameworks, both theoretically and practically, facilitating the transfer of multi-sensory object representations among robots, ensuring that newly developed robots can build upon existing knowledge rather than starting from scratch. These frameworks encompass generating features for newly deployed robots by leveraging insights from experienced robots, developing shared latent feature spaces among robots, and acquiring unified multi-sensory object property representations that are transferable across different tasks. This dissertation contributes to advancing robot perception capabilities by enabling robots to share their perceptual knowledge and publishes three large object exploration datasets by robots to facilitate further investigations. Future work should extend the research horizon by delving into autonomous learning mechanisms for exploratory behaviors, refining adaptability with learning-based policies for handling complex tasks, and automating object selection algorithms to enhance the efficiency of perceptual knowledge transfer models across diverse robots.

Acknowledgments

Completing this dissertation has been made possible by numerous individuals and organizations' invaluable contributions and support. Expressing gratitude to each is a heartfelt endeavor, and I strive to convey my appreciation sincerely. Foremost, I extend my deepest thanks to my advisor, Professor Jivko Sinapov. From our initial collaboration during my master's program to his pivotal role in guiding my decision to pursue a Ph.D., Professor Sinapov's mentorship has been transformative. His clear and thorough guidance and insightful advice have significantly shaped my academic journey, making it a rewarding and enriching experience. I am grateful to Jonathan Francis, whose support during my internships at Bosch has been instrumental. His choice of engaging topics, collaborative spirit, and attentive guidance opened doors to interdisciplinary research beyond my university's lab. Jonathan's knowledge, passion, and unwavering support have been pivotal in my professional growth. Special appreciation goes to my other committee members—Elaine Short, Jason Rife, and Robert Jacob—for their invaluable feedback and guidance. Thank you to my co-authors—Ramtin Hosseini, Michael C. Hughes, Yash Shukla, Michael Edegware, Ho-Hsiang Wu, and Yonatan Bisk—for their contributions and insights on captivating research topics. The collaborative synergy these individuals fostered greatly enriched our work's quality.

I hold Tufts University in high esteem for the quality education and unwavering support provided throughout my academic journey. My sincere thanks extend to the institution. Acknowledgment is also due to DARPA's SAIL-ON program and NSF's crucial funding support, enabling my research endeavors. Gratitude is expressed for the opportunities to work as a teaching assistant under the guidance of remarkable professors, including Jivko Sinapov and Fabrizio Santini.

To my current and former lab mates—Shivam Goel and Samay Pashine—thank you for your feedback and collaborative efforts. Your contributions have undeniably enhanced the quality of my work. Finally, heartfelt thanks to my family and friends for their unwavering support. Their encouragement and understanding have been a cornerstone of my academic journey. I am profoundly grateful for their presence in my life. This dissertation is a collective achievement, and I am indebted to each individual and organization that has played a role in its realization.

Gyan Tatiya

TUFTS UNIVERSITY

February 2024

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

The integration of personal and service robots into real-world settings such as homes and offices has long been a vision of the artificial intelligence and robotics communities [Sin13, Sho17, Fra22]. This vision includes the ability of robots to perform daily tasks, such as lifting heavy boxes, cleaning up after a dinner party, or picking up toys from the floor. To achieve this, robotics researchers have been developing object manipulation-based AI tasks, including object grasping pose estimation [DWLZ21], object rearrangement [ZFT+21], and even folding clothes [DSP+16]. An essential aspect of performing these tasks is having a natural understanding of object properties. While most research in this area uses vision to recognize several object properties, such as color and shape, this is not sufficient as there are implicit properties that cannot be perceived with only a visual input. By "implicit properties," we refer to attributes like weight, surface texture, and sound, which require physical interaction to be learned and understood. For example, a robot that is asked to take out empty food containers from the fridge can detect the containers using vision, but cannot determine if they are empty without physically interacting with them and processing non-visual sensory modalities such as haptic force. It is, therefore, essential to interact with objects and observe the resulting outcomes using multiple sensory modalities to learn about these properties. Moreover, interacting

with objects can help a robot to understand their properties better, as the interaction reveals sensory signals that are otherwise not observable. For instance, a water bottle does not make any sound on its own, but shaking it produces a sound that can be useful in determining whether it has water or not.

Research in the field of robotics has demonstrated that robots are capable of completing various interactive perception tasks that require physical interaction with objects, perception of the outcome of these interactions, and the acquisition of implicit object properties that are grounded in non-visual sensory signals such as audio, tactile, and haptic feedback [HXJ+23, FKL+22]. For example, Sinapov et al. [SSS+14a] demonstrated an object category recognition framework in which an upper-torso humanoid robot used multiple exploratory actions (e.g., grasping, lifting, shaking, pushing) and multiple sensory modalities (e.g., vision, proprioception, and audio) to learn machine learning models that categorize 100 household objects to 20 categories (e.g., cups, cans, bottles, balls). This work demonstrated that using multiple sensory feedback and exploratory actions improves the recognition performance of the robot, indicating that different combinations of behavior and sensory modality contain information useful for category recognition. Additionally, Sinapov et al. [SSS14b] proposed a framework that enables robots to learn and describe the relations between objects, such as "heavier than," by performing exploratory behaviors on objects and training recognition models for multimodal perception. This work also showed that this capability to estimate relations between objects boosts the recognition performance of the robot when learning a new category. Similarly, Gemici and Saxena [GS14] presented a learning system that uses haptic data, such as force and tactile data, to manipulate deformable food objects (e.g., tomatoes, bread, cheese, lettuce) and infer a set of material properties, including hardness, brittleness, elasticity, and adhesiveness. The learned models were then used to recognize the properties of the food and decide an appropriate action to perform on a given food item. Learning-based approaches for interactive perception tasks in robots have shown significant success. However, in each of these works, each robot requires excessive time to perform the necessary object exploration to learn

interactive perception tasks, which prohibits rapid learning of non-visual object representations in practice and, by extension, limits the possibility of real-world robot deployments [PGH+16].

To address these limitations, we might consider transferring the representation of object properties to a new robot to enable it to learn faster and complete downstream tasks more efficiently. However, there are no directly transferable general-purpose representations for non-visual features, as these representations are specific to each robot's kinematics properties, including joint configurations, degrees-of-freedom, and dynamic properties such as mass, center of mass, and inertia, as well as different sensors and physical interaction capabilities. Consequently, a robot's machine learning model cannot be naturally applied to another robot especially for interactive perception tasks that requires understanding of non-visual object properties. Therefore, transferring perceptual knowledge of non-visual object properties from one robot to another is challenging, and each individual robot needs to learn its task-specific sensory models from scratch, which is a time-consuming process.

While transfer learning has witnessed significant advancements in various domains, including computer vision [ZYZ+20] and natural language processing [BPT+22], as well as in the realm of reinforcement learning, with notable techniques such as learning from demonstrations [KGS+20], policy distillation [YP17], and learning inter-task mapping [GDL+17], the challenges associated with transferring non-visual object representations for interactive perception tasks in robotics persist and require further attention. To address this challenge, in this dissertation, we aim to accelerate the learning of a newly deployed robot by transferring perceptual knowledge from an experienced robot that has exhaustively explored objects.

1.2 Dissertation Overview and Research Questions

This dissertation focuses on the transfer of multi-sensory perceptual knowledge acquired through interactions with objects between robots. The primary hypothesis is that by leveraging the common objects explored by robots, a projection function

can be learned to facilitate the transfer of knowledge from a more experienced source robot to a less-experienced target robot. This knowledge transfer enables the target robot to learn about object properties faster and with fewer object interactions, leveraging the perceptual knowledge of other robots instead of relying solely on its own object exploration experience. The central research question addressed in this dissertation is:

How can robots transfer perceptual knowledge acquired through object interactions across heterogeneous robot embodiments, behaviors, sensors, tools, and downstream tasks?

To provide a comprehensive answer to this question, we break it down into four subsidiary research questions, each contributing to the development of theoretical problem formulations and practical frameworks that collectively provide insights into this central research question.

1.2.1 How can we use multimodal deep neural networks for object categorization by leveraging interactive behavior?

In understanding how humans learn about object properties, it is evident that children acquire the ability to discern object categories and recognize objects through physical exploration. This process involves not only visual perception but also incorporates the knowledge of object movement, texture, and sound [Pow99]. It is crucial to note that many commonly used nouns and adjectives have non-visual components, highlighting the importance of multimodal understanding in object semantics [LC09]. Motivated by these cognitive processes, Chapter 4 introduces a deep learning methodology for object category recognition, in which a robot interacts with objects and processes multi-sensory data to predict the category of the object. Our methodology combines visual, auditory, and haptic sensory data with exploratory behaviors such as grasping, lifting, and pushing. Remarkably, our approach surpasses previously published baselines on the dataset, which relied on handcrafted features for each modality. Moreover, our findings emphasize that robots do not require complete sensory data throughout the entire interaction. Instead, accurate

predictions can be made early on during the execution of these behaviors, showcasing the efficiency and effectiveness of our approach.

1.2.2 How can robots transfer perceptual knowledge about objects, acquired through interactive behaviors and multimodal perception, from a source robot to a target robot?

Robots have significantly advanced in acquiring knowledge about objects through interactive behaviors and multimodal perception [SBS+11, ANN+12, TSS+16, KR23, ZAS+23]. However, a fundamental challenge arises when transferring this acquired knowledge to other robots with different physical attributes, interaction capabilities, and sensor models. If the new robot possesses different interaction capabilities, such as distinct sensor models or a unique physical embodiment, the implicit knowledge gained by the previous robot is not directly applicable. This challenge persists, although object properties remain invariant, regardless of variations in robot morphologies and sensing capabilities. Chapter 5, 7 and 8 are dedicated to addressing this challenge by proposing frameworks that leverage the potential of generative models to map the sensory data observed by a source robot to a semantically similar feature space for a target robot. A vital aspect of this mapping process involves establishing source-target correspondences based on the invariant object labels or properties provided by humans. Object properties, considered as intrinsic features, form a stable foundation for effective knowledge transfer. This human-provided information about invariant object properties facilitates the establishment of correspondences, proving crucial for effective knowledge transfer between robots, even in the presence of variations in their physical attributes. Consequently, the target robot can learn about objects without requiring direct physical interactions, enabling it to perform tasks like category recognition on novel objects. This research explores the capabilities of generative models, including Encoder-Decoder Networks (EDN), β-Variational Encoder-Decoder Networks (β-VED), and β-Variational Autoencoder Networks (β-VAE), in facilitating seamless knowledge transfer across robots, behav-

iors, and perceptions, laying the foundation for the development of more versatile and adaptable robotic systems in various domains.

1.2.3 How can robots transfer implicit perceptual knowledge, particularly non-visual object properties, among each other using a shared latent feature space?

Humans rely on various non-visual sensory modalities, such as auditory and haptic, and exploratory behaviors to comprehensively understand objects and their properties [TVCO04a, WWCM07, EB04, CHPS21]. While robots can utilize visual data for tasks like shape and color recognition, they cannot often discern non-visual characteristics, such as texture or weight, solely through visual inputs. Instead, they must physically interact with objects and use non-visual sensory modalities to learn about non-visual object properties. However, sharing this knowledge among robots with varying physical attributes and sensor configurations presents a challenge. Traditionally, each robot would need to learn task-specific sensory models through object interaction, which is time-consuming and impractical for wide-scale deployment. Chapters 6 and 9 introduce frameworks, incorporating techniques like kernel manifold alignment (KEMA) and supervised metric learning via triplet loss, aimed at facilitating the transfer of implicit knowledge of non-visual object properties across multiple heterogeneous robots. Our approaches involve the creation of a common latent space derived from the sensory data of multiple "teacher" or "source" robots during interactions with objects, enabling more efficient training of recognition models for various tasks on "student" or "target" robots. This research strives to enhance the efficiency and practicality of transferring non-visual object knowledge among robots, promoting broader applications in robotics, such as facilitating large-scale robot fleet deployments in factories and warehouses.

1.2.4 How can a robot acquire a task-independent, unified multi-sensory object property representation, transferrable across various tasks, via distillation from large pre-trained models, such as foundation models?

In robotics, achieving a holistic understanding of object properties is pivotal for numerous tasks, from object categorization to intricate manipulation. Inspired by the profound role of multi-sensory integration in human perception, Chapter 10 introduces MOSAIC (Multimodal Object Property Learning with Self-Attention and Integrated Comprehension), an innovative framework designed to expedite the acquisition of unified multi-sensory object property representations. These representations encompass insights from diverse sensory modalities such as visual, auditory, and haptic inputs, recognizing that many essential object properties extend beyond the visual domain. MOSAIC accomplishes this by distilling knowledge from large-capacity Vision-Language Models (VLMs), such as the Contrastive Language-Image Pre-training (CLIP) model [RKH$^+$21], aligning these representations not only across visual but also language, haptic and auditory sensory domains. By leveraging language, MOSAIC empowers robots to learn better representations via language grounding and receive instructions from humans, making human-robot interaction more intuitive and efficient. MOSAIC's novel integration of natural language processing and multi-sensory perception constitutes a major stride towards developing more versatile and capable autonomous systems, paving the way for broader applications across various robotic tasks, including language-conditioned fetch object tasks and enhanced object property recognition. This pioneering research introduces CLIP-based sensory grounding, marking a significant advancement in enhancing the multi-sensory perceptual capabilities of autonomous systems while harnessing the power of language.

1.2.5 Outline and Contributions

The remaining of this dissertation is structured as follows:

Chapter 2: Related Work

In this chapter, we provide a review of relevant literature, exploring object exploration in psychology, the integration of multi-sensory perception in robotics, and the application of transfer learning to facilitate knowledge transfer and enhance robot perception, laying the foundation for the core contributions of this dissertation.

Chapter 3: Robotic Platform and Datasets

This chapter offers a detailed overview of the experimental infrastructure, encompassing robots, sensors, exploratory behaviors, objects, and tools employed in this research, providing valuable insights into the diverse robotic platforms and datasets utilized for multi-sensory knowledge transfer experiments.

Chapter 4: Multimodal Object Category Recognition

This chapter introduces a multimodal deep learning methodology for object category recognition. It harnesses visual, auditory, and haptic sensory data combined with exploratory behaviors. This chapter underscores the necessity of transferring perceptual knowledge across robot platforms.

Chapter 5: Knowledge Transfer with EDN

Chapter 5 unveils a framework based on Encoder-Decoder Networks (EDN) for knowledge transfer across different behaviors, specifically targeting grounded category recognition. This approach excels in generating features for novel objects, introducing the "accuracy delta" metric for evaluating knowledge transfer tasks.

Chapter 6: Haptic Knowledge Transfer with KEMA

Here, we present a method employing Kernel Manifold Alignment (KEMA) to facilitate knowledge transfer of haptic information between heterogeneous robots in simulation environments. The results demonstrate accelerated object recognition and improved performance in recognizing novel objects through knowledge sharing.

Chapter 7: β-VAE for Cross-behavior and Cross-perception Transfer

Building upon Chapter 5, this chapter enhances knowledge transfer using the β-Variational Autoencoder Network (β-VAE). It addresses two knowledge transfer scenarios: cross-behavior and cross-perception, with applications in object category and object identity recognition. An innovative algorithm for efficient object selection

for knowledge transfer model learning is provided.

Chapter 8: Learning Projection Functions

Chapter 8 evaluates the effectiveness of two projection functions, EDN and KEMA, for building object property-based and object identity-based correspondences. Real-world heterogeneous robots are employed for tasks related to object properties and identities. A novel data augmentation technique is proposed to enhance knowledge transfer performance.

Chapter 9: Shared Latent Feature Space via Triplet Loss

This chapter introduces the concept of a shared latent feature space using supervised metric learning via triplet loss. Knowledge transfer is applied across different tools and behaviors, with a focus on object identity recognition.

Chapter 10: MOSAIC Framework for Unified Multimodal Representations

Chapter 10 introduces the MOSAIC (Multimodal Object Property learning with Self-Attention and Integrated Comprehension) framework, incorporating a contrastive loss mechanism. This multi-sensory integration system distills foundation models and aligns them with real-world robotic data, enhancing perception capabilities. MOSAIC creates unified representations transferable across various tasks, as demonstrated by its remarkable performance in zero-shot learning scenarios.

Chapter 11: Conclusion and Future Directions

The final chapter summarizes the key findings of this dissertation and proposes avenues for future research.

Chapter 2

Related Work

In this chapter, we delve into the body of literature that underpins the contributions of this dissertation. Our exploration begins by surveying the field of object exploration in psychology and cognitive science, which provides the foundational understanding of how humans interact with and learn about objects. We then transition into the realm of multi-sensory object perception in robotics, where we discuss the pivotal role of non-visual sensory modalities in expanding the capabilities of robotic systems. Following this, we introduce the concept of transfer learning, a fundamental principle in machine learning, which will serve as a bridge to migrate knowledge from one robot to another in the context of interactive object perception. This chapter is intricately connected to the subsequent chapters, where we delve into specific methodologies and frameworks designed to facilitate knowledge transfer across robots, ultimately enhancing their perceptual capabilities.

2.1 Object Exploration in Psychology and Cognitive Science

In psychology and cognitive science, the exploration of objects holds a foundational role in our understanding of human perception and learning. Research within these disciplines has illuminated the pivotal role of interactive object manipulation in acquiring knowledge about objects' tactile, haptic, proprioceptive, and auditory prop-

erties. Groundbreaking studies, such as those by Gibson and Power [Gib88, Pow99], have shown that this interactive process commences early in human development, with infants engaging in a form of exploration that is less goal-oriented and more focused on gaining insights into how objects feel, sound, and move [ST99]. As individuals mature, this ability evolves into a more purpose-driven endeavor, aligning with the desire to acquire specific knowledge about an object's properties.

Central to this exploration is the integration of multiple sensory modalities. Fusing visual, tactile, auditory, and kinesthetic inputs is pivotal in enabling humans to recognize and interact with objects effectively [EB04]. This interplay of sensory modalities, as beautifully articulated by David Katz in 1925, underscores the intrinsic connection between interaction and the revelation of tactual properties:

"The tactual properties of our surroundings do not chatter at us like their colors; they remain mute until we make them speak [Kat25]".

This profound insight highlights the necessity of human engagement with objects for a rich understanding of their attributes, as the sensory dimensions of touch are unveiled through these interactive encounters.

This deep-rooted understanding of object exploration in psychology and cognitive science has been a rich source of inspiration for robotics and artificial intelligence [HXJ+23, FKL+22]. The exploration of objects and the utilization of diverse sensory signals promise to allow robots to acquire comprehensive knowledge about the objects they encounter in their environments. Just as human cognition benefits from multiple sensory modalities when dealing with object recognition, robotic systems aspire to harness these insights for their learning and task-execution processes. However, the challenge lies in scaling the knowledge acquired by robots via object exploration to many robots, each with a unique set of capabilities encompassing its embodiment and sensory modalities. This uniqueness results in distinct interactions with and perceptions of the world.

As we progress through this dissertation, we explore methodologies and frameworks for transferring this invaluable knowledge across robots, enabling them to share their knowledge, reducing the need for new robots to learn from scratch,

and minimizing the time spent exploring objects. This collective knowledge sharing has the potential to advance the field of robotics, pushing its current boundaries.

2.2 Multi-sensory Object Perception in Robotics

Traditionally, the realm of object category acquisition and recognition primarily revolved around the visual domain, leveraging models trained on expansive image datasets without the need for direct physical interaction with objects [ZZC+12, LWL+10, LZD15, Zha16, Cha14]. However, this approach's limitations become apparent when faced with objects that cannot be fully comprehended through visual data alone. Categories like "soft" or "empty" might sound simple, but visually identical objects' materials, internal states, and compliance can differ significantly. Visual data alone may not capture the essence of these distinctions.

The field of robotics recognizes the potential inherent in expanding our perception beyond visual cues. While most object recognition methods in robotics have traditionally been anchored in visual sensing, several innovative research studies have focused on embracing multiple sensory modalities, often coupled with exploratory actions [BHS+17, PGH+16]. This approach advocates that robots, like humans, should harness the richness of sensory information offered by non-visual modalities for an enhanced understanding of object properties. The incorporation of auditory [SWS09, EKSW18], haptic [BRK12a], and tactile feedback [SSSS11, LCL19] has enabled robots to delve into the realm of recognizing objects and their properties.

Furthermore, incorporating non-visual sensory modalities has demonstrated remarkable potential in the learning of object categories [SSS+14a, HBMK16, ECK17, TS19]. This extends to encompassing the comprehension of object relations [SKSS16], enabling robots to not only categorize objects but also understand how these objects relate to each other. Beyond these capabilities, this integration has extended its reach to grasping the nuances of human language in describing objects — a leap in human-robot interaction [RK19]. These additional sensory dimensions, complementing visual perception, significantly enrich the learning process of robotic

systems.

Although significant advances have been made in incorporating multi-sensory cues into robotics, transferring object exploration knowledge across robots remains a considerable challenge [FKL+22]. While some research has been conducted on the transfer of vision-based knowledge [HMG+22, NLWS20], the domain of transferring interactive perception knowledge still needs to be explored. This is akin to robots acquiring a wealth of knowledge about objects through their various senses but unable to share this hard-earned wisdom effectively. In this dissertation, we investigate the underexplored territory of transferring interactive perception knowledge across robots, with the goal of expediting robots' learning of object properties and expanding the horizons of robotic perception. This exploration includes aspects such as language grounding and distilling knowledge from foundation models, crucial elements for acquiring robust multisensory object representations.

2.3 Enhancing Robotic Perception: A Transfer Learning Approach

Transfer learning is a fundamental concept in machine learning, often necessitated by the impracticality of obtaining large, consistent training datasets in real-world scenarios [ZG22, WLL+22, ZLQ+22, YXL22]. This technique relaxes the traditional assumption that training data and test data must be drawn from the same distribution. Instead, transfer learning leverages knowledge learned in one domain, known as the source domain, to improve performance in a related but different domain, referred to as the target domain. This approach substantially diminishes the demand for extensive training data and reduces the time required for data collection in the target domain. Analogously, it mirrors how humans can transfer knowledge across domains. For example, someone with expertise in playing the guitar can efficiently learn to play the piano, as the musical knowledge transfers. Likewise, in transfer learning, the goal is to transfer knowledge from a source domain to enhance learning in a target domain. In this context, the source domain represents a domain where

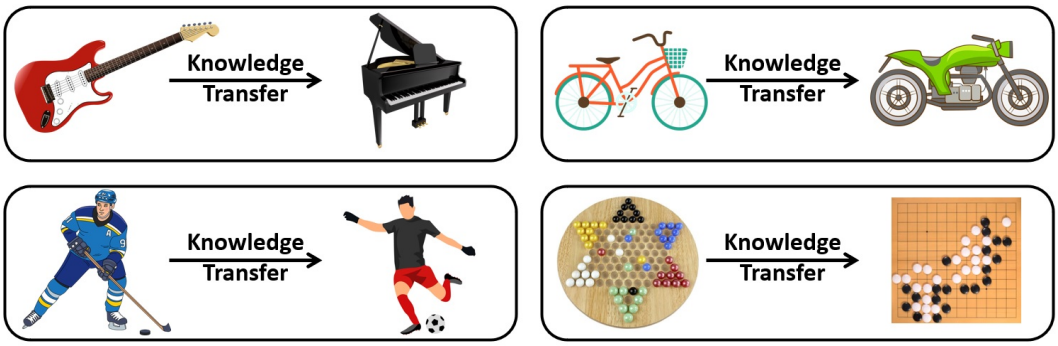


Figure 2.1: Intuitive examples of transfer learning.

learning has occurred, and the target domain is where we aim to apply this learned knowledge. Fig. 2.1 shows some intuitive examples of transfer learning.

Transfer learning has demonstrated remarkable efficacy in numerous computer vision applications [ZYZ+20] like image classification [OSSP23, ZZW+20, CLS+18], human activity classification [LSSVG23, YXC+22, YHSS22], event recognition [DXC12], and face recognition [HYW14], as well as in natural language processing applications [BPT+22] encompassing text sentiment classification [CBL+23, LSJJ19, GBB11], text summarization [TT22, HW17, WC13], and speaker-independent speech recognition [RQD00]. Additionally, there have been several breakthroughs in transfer learning in the context of reinforcement learning, such as learning from demonstrations [KGS+20], policy distillation [YP17], learning inter-task mapping [GDL+17]. However, specific challenges related to transferring non-visual object representations for interactive perception tasks in robots, especially in the context of different morphologies, have yet to be fully addressed. These challenges occur in scenarios where data collection by the target robot proves expensive, and collecting data from scratch for each robot is infeasible. In such situations, the prudent choice is to facilitate knowledge transfer from a source robot to alleviate the resource-intensive task of independent data collection. This dissertation explores how transfer learning can bridge this gap and empower robots to enhance their interactive perceptual capabilities. This dissertation proposes three novel frameworks aimed at addressing the challenges associated with transferring object property representations for interactive perception tasks in robots. Each framework, discussed in detail

below, contributes to advancing the field and lays the foundation for more robust and adaptable robotic systems.

2.3.1 Transfer using Projection to Target Feature Space

In the realm of transfer learning for robotic systems, a pivotal focus of this dissertation lies in addressing the challenge of seamlessly transferring acquired knowledge about objects from one robot to another with disparate physical attributes, interaction capabilities, and sensor models. This framework introduces a novel approach inspired by domain adaptation principles. Encoder-decoder networks, proven successful in various domains [MKK+18, GFL19, HZ93, SVL14], serve as a fundamental component of this framework. The objective is to map sensory data observed by a source robot to a semantically similar feature space for a target robot. Unlike traditional models assuming identical feature spaces, our approach acknowledges potential differences between source and target domains, emphasizing the importance of semantic similarity [BDBC+10, MMR09]. By establishing correspondences based on invariant object labels or properties provided by humans [ZAS+23], our framework capitalizes on intrinsic features, such as object properties, as a stable foundation for effective knowledge transfer. This mapping process, explored in Chapters 5, 7, and 8, facilitates the transfer of multisensory object knowledge, enabling the target robot to learn about objects without necessitating direct physical interactions [KR23]. Through this research, the capabilities of generative models, including Encoder-Decoder Networks (EDN) [HS06], β-Variational Encoder-Decoder Networks (β-VED) [LBL19], and β-Variational Autoencoder Networks (β-VAE) [LWL+17], are harnessed to lay the foundation for more versatile and adaptable robotic systems across diverse domains. This framework uniquely contributes to the field by emphasizing the nuanced challenges of transferring knowledge in the context of interactive object exploration by robots [TSS+16, ZAS+23].

2.3.2 Transfer using Projection to Shared Latent Feature Space

Domain adaptation, a crucial facet of transfer learning, addresses shifts in feature spaces between a source domain (training set) and a related but different target domain (test set). This framework focuses on leveraging kernel manifold alignment (KEMA) for domain adaptation, a method capable of aligning multiple domains of varying dimensionality without the need for paired examples. KEMA, successfully applied in visual object recognition, facial expression recognition, and human action recognition [TCV16, LLL$^+$18], introduces a novel application to haptic data for object recognition in robots. While robots excel at visual tasks, discerning non-visual characteristics like texture or weight often requires physical interaction and exploration. Traditional methods would necessitate individual robots to learn task-specific sensory models, impractical for wide-scale deployment. Chapters 6 and 9 detail our frameworks, incorporating KEMA and metric learning via triplet loss. These approaches create a shared latent space derived from sensory data during interactions, enhancing the efficiency of transferring non-visual object knowledge among robots with varying physical attributes and sensor configurations. Our work strives to contribute to the broader field of robotics by promoting practical and efficient transfer of non-visual object knowledge among heterogeneous robots, addressing a crucial gap in existing methodologies [TVCO04a, WWCM07, EB04, CHPS21].

2.3.3 Transferable Unified Multi-sensory Object Property Representations

Recent strides in contrastive learning, particularly Contrastive Language-Image Pre-training (CLIP) [RKH$^+$21], have demonstrated its efficacy in generating generalized representations for both text and images, excelling in diverse tasks such as zero-shot image classification and image retrieval via text. While CLIP's knowledge has been extended to audio [WSKB22], our MOSAIC (Multimodal Object Property Learning with Self-Attention and Integrated Comprehension) framework stands as the pioneering effort to ground sensory data obtained through robotic object explo-

ration. MOSAIC introduces a novel approach to learning unified multi-sensory object property representations by distilling knowledge from the extensive pre-trained CLIP text model. In Chapter 10, MOSAIC aligns representations not only across visual but also haptic and auditory sensory domains, recognizing the importance of encompassing diverse sensory modalities. By leveraging language, MOSAIC empowers robots to acquire better representations and receive instructions from humans, enhancing human-robot interaction intuitively and efficiently. This pioneering research in CLIP-based sensory grounding contributes significantly to advancing the multi-sensory perceptual capabilities of autonomous systems, marking a crucial step toward versatile and capable robotic systems with applications across various domains.

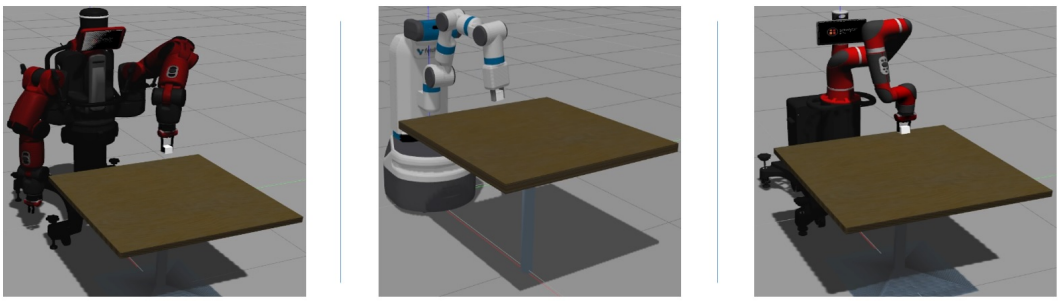
2.4 Summary

In this chapter, we reviewed the relevant literature that informs the core contributions of this dissertation. We commenced our exploration by delving into the world of object exploration in psychology and cognitive science, highlighting the importance of interactive engagement with objects to understand their attributes and integrate multiple sensory modalities in this process. This foundational understanding inspired the integration of multi-sensory perception in robotics, where we discussed the significance of moving beyond visual recognition alone to enhance robotic object perception and understanding. To facilitate knowledge transfer across robots and expedite their learning processes, we introduced the concept of transfer learning, a powerful tool borrowed from machine learning. This chapter serves as the backdrop for our subsequent investigations into transferring interactive perception knowledge across robots and advancing the field of robotic perception.

Chapter 3

Robotic Platform and Datasets

This chapter offers an overview of the robotic platform and datasets employed in the research detailed in this dissertation. The study leveraged a diverse array of robotic platforms, encompassing both simulation-based and real-world robotic systems, each equipped with sensors for data recording. These robots and their sensors played a pivotal role in collecting multiple datasets. These datasets, comprising sensory data recorded during interactions with various objects, were crucial in conducting experiments related to multi-sensory knowledge transfer. All datasets generated as part of this research have been thoughtfully curated and are made publicly available, serving as valuable resources for future research endeavors. Importantly, the datasets curated for this research have been extensively used in experimental evaluations of the knowledge transfer methodologies detailed in Chapter 6, Chapter 8, and Chapter 9. These experiments focus on various aspects of multi-sensory knowledge transfer among robots, utilizing the rich and varied data collected from the robotic platforms. The subsequent sections delve into the specifics of the robots, the array of sensors at their disposal, and a detailed exploration of the datasets collected for this dissertation.



(A) Baxter (B) Fetch (C) Sawyer

Figure 3.1: The three simulated robots employed for object exploration - (A) Baxter, (B) Fetch and (C) Sawyer.

3.1 Robots and Sensors

3.1.1 Simulated Robots

In the context of tabletop manipulation, this research employed three simulated robots—Baxter, Fetch, and Sawyer, as depicted in Fig. 3.1. Baxter has dual arms, with the left arm utilized for object interactions, while Fetch and Sawyer each have a single arm configuration. All three robots, including two grippers, have 9 degrees of freedom (DOF). To enable data collection, these robotic platforms had an array of sensors, encompassing effort, position, and velocity sensors at each joint and the end-effector. Baxter's sensory data acquisition rate was set at 50Hz, while Fetch and Sawyer operated at 100Hz. For a more comprehensive understanding of the objects explored by these robots and the specific exploratory behaviors employed, additional details can be found in Chapter 6.

3.1.2 Real-world Robots

Our research also engaged two real-world robots, Baxter and UR5, in a tabletop manipulation environment (illustrated in Fig. 3.2). Baxter had dual 7-degree-of-freedom arms and a 2-finger gripper. For object exploration, we used Baxter's left arm. UR5, on the other hand, possessed a 6-DOF structure and a 2-finger Robotiq 85 gripper.

Baxter: Baxter was equipped with a PrimeSense camera* mounted on its head,

*https://en.wikipedia.org/wiki/PrimeSense

which captures images at a resolution of 640×480. Additionally, an Audio-Technica PRO 44 microphone[†] was positioned on its workstation. The Baxter hand camera recorded images at a resolution of 480×300. Baxter's sensory suite includes force-torque sensors, measuring effort at each joint and torque at the end-effector.

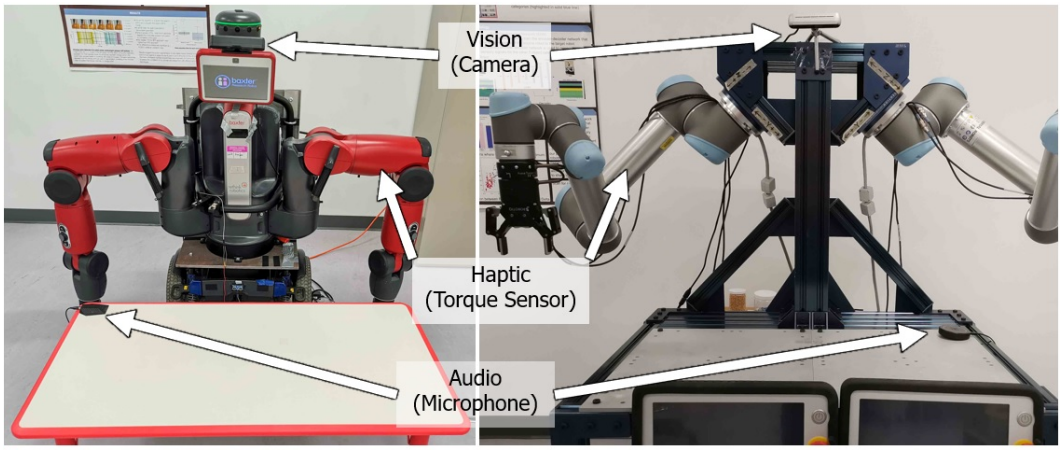
UR5: UR5 utilized a Seeed Studio ReSpeaker microphone[‡] on its workstation. UR5, in the course of object exploration for Chapter 8, employed an Orbbec Astra S 3D Camera[§]. For Chapter 9, we transitioned to using an Intel RealSense Depth Camera D455[¶]. Both of these cameras, securely affixed to UR5's frame, captured images at a resolution of 640×480. Additionally, in Chapter 9, UR5 incorporated the use of the DIGIT tactile sensor[‖], which captures vision-based tactile images at a resolution of 320×240. This sensor was mounted on one of the gripper's fingers. Sample tactile images captured by UR5 while holding various tools can be seen in Fig. 3.3. UR5 also has force sensors that measure effort at each joint and a force-torque sensor at the end-effector.

In summary, Baxter had 14 sensors, while UR5 had 12, covering diverse modalities. A list of these sensory modalities, coupled with their respective sampling rates, can be found in Table 3.1.

3.2 Datasets for Multisensory Knowledge Transfer

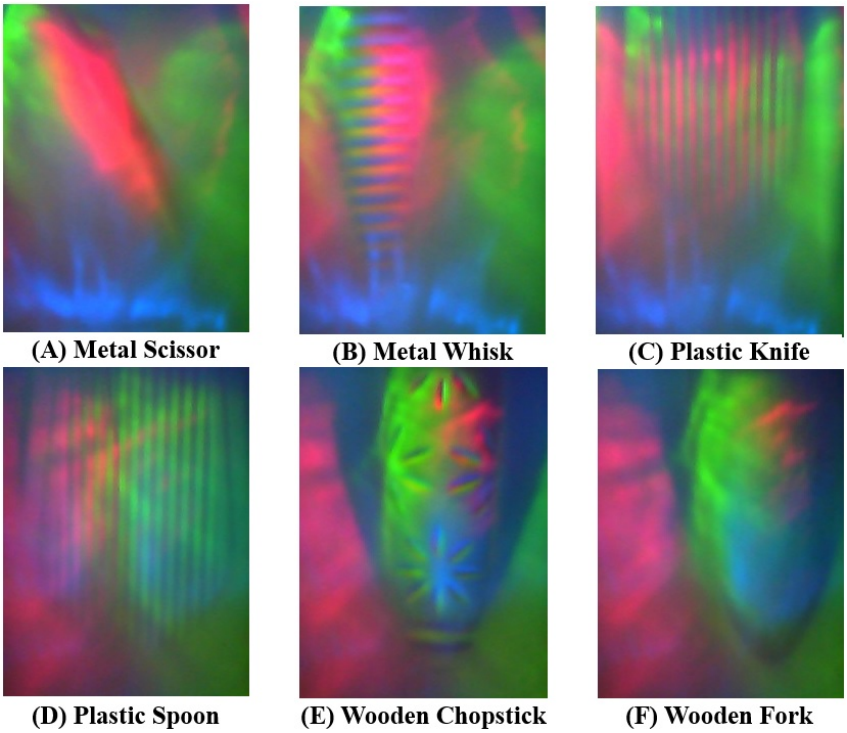
This section encompasses an overview of the datasets utilized in this dissertation, where robots explore objects and record their sensory signals. We begin by describing publicly available datasets employed in this research, followed by the introduction of new datasets collected to facilitate our investigations.

[†] `https://www.amazon.com/dp/B0002BB0OS`
[‡] `https://www.amazon.com/dp/B07ZGZSBS4`
[§] `https://www.amazon.com/dp/B07484SMB8`
[¶] `https://store.intelrealsense.com/buy-intel-realsense-depth-camera-d455.html`
[‖] `lambeta2020digit`



(A) Baxter (B) UR5

Figure 3.2: The two real-world robots employed for object exploration - (A) Baxter and (C) UR5.



(A) Metal Scissor **(B) Metal Whisk** **(C) Plastic Knife**

(D) Plastic Spoon **(E) Wooden Chopstick** **(F) Wooden Fork**

Figure 3.3: Tactile images captured by UR5 holding various tools: (A) metal scissor, (B) metal whisk, (C) plastic knife, (D) plastic spoon, (E) wooden chopstick, and (F) wooden fork.

Table 3.1: Sensors with sampling rates for Baxter and UR5.

Modality	**Baxter**	**UR5**
Accelerometer	100 Hz	-
Audio	44.1 kHz	16 kHz
End-Point State (Force)	100 Hz	125 Hz
End-Point State (Torque)	100 Hz	125 Hz
Gripper State (Force)	20 Hz	-
Gripper State (Position)	20 Hz	30 Hz
Gripper State (Velocity)	-	30 Hz
Hand Camera (RGB)	6 Hz	-
Head Camera (Depth)	30 Hz	30 Hz
Head Camera (Point-Cloud)	30 Hz	30 Hz
Head Camera (RGB)	30 Hz	30 Hz
Joint State (Effort)	140 Hz	135 Hz
Joint State (Position)	140 Hz	135 Hz
Joint State (Velocity)	140 Hz	135 Hz
Range	100 Hz	-
Tactile Images (RGB)	-	30 Hz

3.2.1 Existing Public Datasets

In our research, we have made use of two publicly available datasets: **Sinapov14** [SSS+14a] and **Sinapov16** [SKSS16]. These datasets present valuable resources for understanding multi-sensory knowledge transfer in robotic systems.

Sinapov14 Dataset: Sinapov14 consists of data collected using the Barrett humanoid robot equipped with a 7-DOF arm. This dataset encompasses interactions with 100 household objects from 20 distinct categories. Ten exploratory behaviors were executed, each named as Look, Press, Grasp, Hold, Lift, Drop, Poke, Push, Shake, and Tap. The Look behavior primarily captures visual data, while the interactive behaviors encompass visual, audio, vibrotactile, and haptic sensory data gathered through the robot's sensors. For each object, each behavior was repeated five times, yielding a total of 5,000 interactions (10 behaviors x 5 trials x 100 objects). We have leveraged this dataset in our experiments detailed in Chapters 4, 5, 7 and 10.

Sinapov16 Dataset: Sinapov16 employs the Kinova MICO robot, which explored 32 common household objects using eight distinct exploratory actions: Look, Grasp, Lift, Hold, Lower, Drop, Push, and Press. Similar to Sinapov14, the Look behavior

focused on visual data, while the other behaviors integrated audio, proprioceptive (finger positions for grasp), and haptic (i.e., joint forces) data acquired during interactions with the objects. Each of the eight behaviors was conducted five times on each of the 32 objects, resulting in a total of 1,280 interactions (8 behaviors x 5 trials x 32 objects). This dataset has been used in our experiments outlined in Chapter 7.

3.2.2 Newly Collected Datasets

In the pursuit of conducting research for this dissertation, we recognized the need for datasets specifically tailored to our experimental objectives. We present three novel datasets: **Tatiya20** [TSES20], **Tatiya23** [TFS23], and **Tatita24** [TFS24].

Tatiya20 Dataset: Tatiya20 dataset revolves around interactions with three simulated robots: Baxter, Fetch, and Sawyer. These robots feature heterogeneous embodiments, as described in Section 3.1.1. The dataset encompasses four distinct behaviors - Grasp, Pick, Shake, and place - executed on 25 block objects, each varying in weight from 0.01 kg to 1.5 kg. The robots' behaviors are encoded as joint-space trajectories with joint values randomly sampled within specified ranges for each joint, aiming to simulate real-world variability. Effort feedback from all joints is recorded during each behavior. Each behavior is repeated 100 times on each object, resulting in an extensive dataset of 30,000 examples (3 robots x 4 behaviors x 25 objects x 100 trials). This dataset underpins the experiments detailed in Chapter 6.

Tatiya23 Dataset: Tatiya23 involves interactions with two real-world robots, UR5 and Baxter, each having heterogeneous characteristics, as described in Section 3.1.2. These robots execute eight behaviors: Look, Grasp, Pick, Hold, Shake, Lower, Drop, and Push. The Look behavior records visual modalities (RGB, Depth, and Point-Cloud) from their head camera and serves as a non-interactive behavior. The remaining behaviors are interactive and are encoded as joint-angle trajectories. For all behaviors, Point-Cloud data is recorded for the first 5 frames. The dataset includes 95 objects (cylindrical containers) with variations in color, content, and weight.

Both robots explore these objects, performing five trials on each, resulting in a comprehensive dataset of 7,600 interactions (2 robots x 8 behaviors x 95 objects x 5 trials). This dataset plays a pivotal role in the experiments outlined in Chapter 8.

Tatita24 Dataset: Tatiya24 centers around interactions with the UR5 robot, as described in Section 3.1.2. The robot employs six tools: metal-scissor, metal-whisk, plastic-knife, plastic-spoon, wooden-chopstick, and wooden-fork, to execute six behaviors: Look, Stirring-slow, Stirring-fast, Stirring-twist, Whisk, and Poke. Look is a non-interactive behavior, whereas the others involve dynamic movements encoded as joint-angle trajectories. The robot explores 15 granular food-like objects (e.g., salt, wheat) stored in cylindrical containers. Ten trials are conducted on each object using a tool, resulting in a dataset of 5,400 interactions (6 tools x 6 behaviors x 15 objects x 10 trials). This dataset plays a pivotal role in the experiments outlined in Chapter 9.

For detailed insights into the behaviors, objects, sensory features, and tasks of each dataset, please refer to the specific chapter where each dataset is utilized. These datasets, while distinctive in terms of robot platforms and explored objects, have provided a rich foundation for our research into multisensory knowledge transfer. Table 3.2 provides a summary of both the existing datasets employed and the new datasets we have meticulously collected to facilitate our knowledge transfer experiments.

3.3 Summary

This chapter presents the foundational elements of the research, detailing the robotic platforms and their sensors pivotal to the dissertation's core investigations. The robotic platform comprises both simulated and real-world agents, including Baxter, Fetch, Sawyer, and UR5, each endowed with a unique array of sensory modalities essential for data collection during object interactions. These robots are pivotal in the multi-sensory knowledge transfer explored throughout the dissertation.

The chapter also introduces a critical component - the datasets. It classifies

Table 3.2: This table provides an overview of the sensory modalities captured in each dataset used in this dissertation, along with the number of robots involved, objects explored, distinct behaviors performed, types of tools employed, the number of trials conducted, and the total count of interactions recorded for each dataset.

Datasets	*Modalities*	*Robots*	*Objects*	*Behaviors*	*Tools*	*Trials*	*Interactions*
		Existing Public Datasets					
Sinapov14	vision, auditory, haptic, tactile	1	100	10	-	5	5,000
Sinapov16	vision, auditory, haptic	1	32	8	-	5	1,280
		Newly Collected Datasets					
Tatiya20	haptic	3	25	4	-	100	30,000
Tatiya23	vision, auditory, haptic	2	95	8	-	5	7,600
Tatiya24	vision, auditory, haptic, tactile	1	15	6	6	10	5,400

these datasets into two categories: existing public datasets, including **Sinapov14** and **Sinapov16**, and newly collected datasets, encompassing **Tatiya20**, **Tatiya23**, and **Tatita24**. Table 3.2 shows an overview of each dataset used in this dissertation. These datasets play a pivotal role in the empirical investigations detailed throughout the dissertation. These datasets not only aid in deepening our comprehension of knowledge transfer but also stand as valuable resources for the broader research community, spanning domains beyond the specific investigations in this work. They hold the potential to benefit research in robotics, machine learning, and related fields, extending their usefulness beyond the scope of this dissertation.

Chapter 4

Deep Multi-Sensory Object Category Recognition Using Interactive Behavioral Exploration*

4.1 Introduction

Learning to classify objects into categories is an important skill for a wide variety of robot tasks and an open research challenge in the fields of robotics and computer vision. For example, a domestic service robot that has to clean up a dining table needs to identify semantic categories of objects, like "glass", "full", "open", etc. While some categories can be identified using visual input alone, others cannot and thus satisfactory performance in real-world applications remains a challenge [ZZC+12, LWL+10, LZD15, Zha16, Cha14].

Children learn to discern object categories and recognize objects through physical exploration, where they not only learn what objects look like, but also how they move, feel, and sound [Pow99]. This knowledge is crucial for learning object semantics as the majority of the most common nouns and adjectives humans

*This chapter is based on the following paper: Gyan Tatiya and Jivko Sinapov, "Deep multi-sensory object category recognition using interactive behavioral exploration", *IEEE International Conference on Robotics and Automation (ICRA)*, pages 7872–7878. IEEE, 2019. [TS19]

use have a non-visual component [LC09]. Yet, most robots today rely on pre-trained computer vision models, e.g., [RDGF16], and thus are unable to reason about semantics that cannot be detected using vision alone.

To address these limitations, we propose a deep multimodal learning methodology that enables a robot to categorize novel objects by performing exploratory interactions and processing multi-sensory data input, shown in Figure 4.1. The proposed method is evaluated on a publicly available dataset in which a humanoid robot explored a set of 100 objects using 9 different exploratory behaviors while recording visual, haptic, and auditory data. For all behaviors, the proposed multimodal network architecture either substantially outperformed the previously published baseline, or produced comparable recognition rates. Furthermore, we demonstrate that our approach can produce accurate category estimates with only a fraction of the data produced by an individual behavior, suggesting that exploratory behaviors can be designed to be shorter in duration, allowing a robot to learn multi-sensory object properties quicker in a deployed, realistic setting.

In the context of the broader dissertation, this chapter introduces specialized architectures for processing raw multi-sensory data to predict object categories. However, the models learned in this chapter cannot be directly used by other robots, and each robot must learn its own model by object exploration from scratch, a time-consuming process. Chapters 5 and 7 propose methods for *Transfer using Projection to Target Feature Space*, which were evaluated on the same robot as in this chapter. Chapter 10 introduces a method for learning *Transferable Unified Multi-sensory Object Property Representations* and outperforms the method proposed in this chapter for the object category recognition task. Overall, this chapter lays the foundation for the need to develop knowledge transfer methodologies.

4.2 Related Work

Object category acquisition and recognition has been studied extensively in the visual domain, where models can be trained on large image datasets with no need

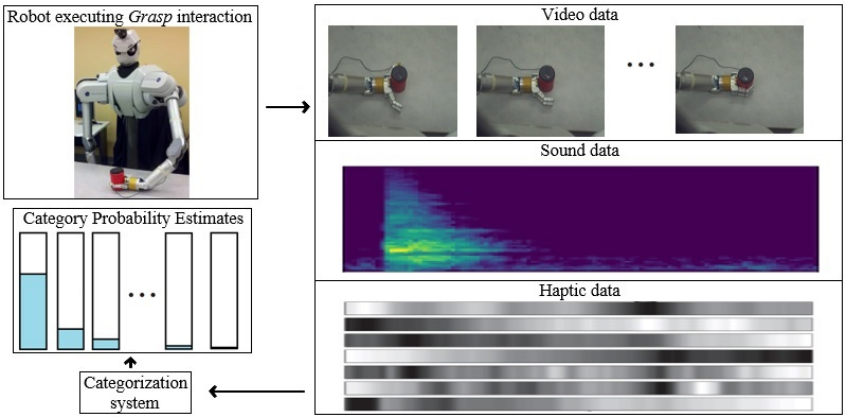


Figure 4.1: Overview of the proposed categorization pipeline.

for robotic interaction with objects [ZZC+12, LWL+10, LZD15, Zha16, Cha14]. For many semantic object categories (e.g., "soft", "empty"), visual information alone may not be sufficient as visually identical objects can differ in material, internal state, and compliance.

To address these cases, several research lines use proprioceptive, haptic, auditory, and/or tactile feedback of robot interaction with objects for category recognition [NANI11, SSS+14a, SSS14c, GZ16]. For example, Nakamura *et al.* in [NANI11] proposed a method that enables the acquisition of object concepts from multiple modalities, such as visual, auditory, and haptic information gathered by robots. Sinapov *et al.* [SSS+14a] demonstrated a category recognition framework in which the robot uses multiple exploratory actions (e.g., grasping, lifting, shaking, pushing) to learn object category models and categorize 100 objects. More recently, Thomason *et el.* [TSS+16, TPS+17, TSMS18] demonstrate how the category recognition method proposed in [SSS14c] can be deployed on a service robot to learn object semantics extracted from human-robot dialog. These examples of multisensory perception used hand-crafted features for different modalities and require some amount of feature engineering, especially when adding new sensory modalities.

Several works have explored deep learning methods for tasks like surface material classification and tactile understanding using visual and haptic modalities [ECK17, GHKD16, ZFJ+16]. Erickson *et al.* [ECK17] presented a semi-supervised

learning approach for material recognition with Generative Adversarial Networks (GANs) that enables a robot to learn from haptic features such as force, temperature, and vibration data from interactions with everyday objects and classify them into six material categories. Gao *et al.* [GHKD16], proposed a deep learning method for tactile understanding using haptic and visual signals. First, individual visual and haptic prediction networks were trained and then they used activations from these networks to train a multimodal network. They demonstrated that combining data from both modalities improves performance. We note that further research work is necessary to use modern learning techniques, which is relatively unexplored in object category recognition. In particular, we present an architecture that uses a larger number of diverse exploratory actions, and consider three types of sensory feedback at the same time: visual, haptic, and auditory.

4.3 Learning Methodology

For each sensory modality, we investigated several network configurations to find ones that achieve high performance on object categorization tasks using visual, audio, and haptic data in a multimodal setting*. Next, we describe these networks along with notation and problem formulation.

4.3.1 Notation and Problem Formulation

Let $\mathcal{B}$ be the set of exploratory behaviors, let $\mathcal{O}$ be the set of objects, and let $\mathcal{M} = \{v, a, h\}$ be the set of modalities (vision, audio, and haptics). During each object exploration trial, the robot applies all of its exploratory behaviors on an object $o \in \mathcal{O}$ and records the 3 different sensory data signals for each modality. Thus, during the i^{th} exploration trial, for each behavior $b \in \mathcal{B}$, the robot observed features:

*Datasets and source code for study replication are available as Jupyter Notebooks at: `https://github.com/gtatiya/Deep-Multi-Sensory-Object-Categorization`. Development environment and network hyper-parameters details are discussed in the README file of the repository. Some alternative network configurations are also discussed with the source code.

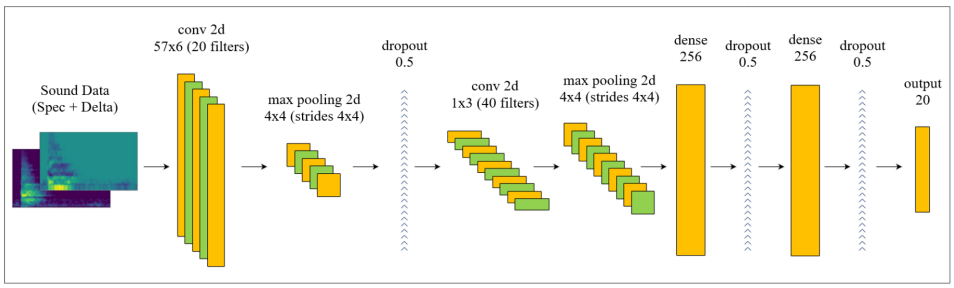


Figure 4.2: The architecture of CNN used for sound classification.

$$\mathrm{X}_i^v \in \mathbb{R}^{w \times h \times t_i^v}, \mathrm{X}_i^a \in \mathbb{R}^{f \times t_i^a}, \mathrm{X}_i^h \in \mathbb{R}^{d \times t_i^h} \tag{4.1}$$

where w and h are the width and height of each image, f is number of frequency bins in the sound spectrogram, d is the number of channels (e.g., number of robot joint-torque sensors) in haptic data, and t_i^v, t_i^a, and t_i^h are the number of frames (e.g., number of images) produced over the course of the interaction for each of the three modalities.

Let the function *label* $(o) \rightarrow y$ be a labeling function that given an object o outputs a label $y \in Y$, where Y is the set of category labels. It is important to note that the labels assigned to objects are provided by a human supervisor. The task of the robot is to learn a category recognition network for each behavior $b \in \mathcal{B}$, that predicts the correct label y, given a sensory signal from modality $m \in \mathcal{M}$ detected while interacting with object o using b. In addition, for each behavior, the robot also learns a multimodal neural network that takes all the modalities of an interaction with an object as input and predicts its category label. Each of the networks estimates a probability for each of the category labels as described below:

$$\begin{aligned} &\Pr(\hat{y} = y | x_i^m), \text{ for a single modality} \\ &\Pr(\hat{y} = y | x_i^v, x_i^a, x_i^h), \text{ for all the modalities} \end{aligned} \tag{4.2}$$

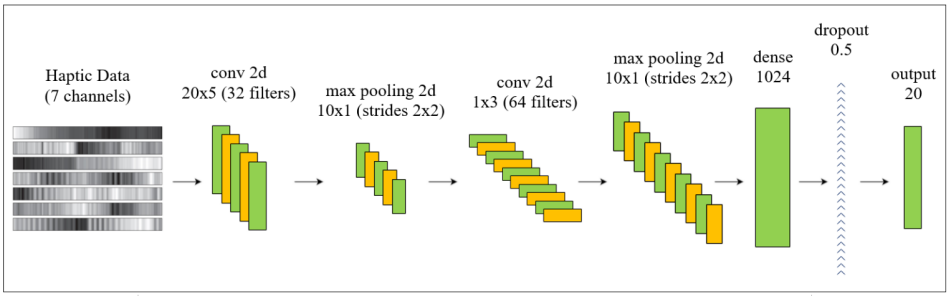


Figure 4.3: The architecture of CNN used for haptic classification.

4.3.2 Visual Network Architecture

4.3.2.1 Image Sequence Pre-processing

For each behavior $b \in \mathcal{B}$, we calculated the average number of image frames per interaction and extracted that many equally-spaced frames from each interaction's image sequence, where each frame was resized to 120 x 90 pixels.[†] For example, the video of a *press* interaction took 48 frames on average for each of 500 trials (100 objects with 5 trials each), so we extracted 48 frames from all the videos of *press* interactions. These pre-processing steps were applied to all the videos of each interaction.

4.3.2.2 Video Network Architecture

Convolutional neural networks (CNNs) have been highly successful in image classification tasks [KSH12, SZ14, LdADSOS17] and Recurrent Neural Networks (RNNs) have been shown to perform well in classifying sequential data [BCB14, SVL14, GJM13, SMS15]. Much work uses the combination of a CNN and an RNN by processing each frame using CNN before feeding it to RNN for video classification [XHD16, MSPGiN16, DAHG$^+$15]. This approach turned out to be impractical for our dataset because the combination of a CNN and an RNN makes a network very deep, which requires a large number of examples to learn all the parameters of the network during training; however, our dataset is very small - there are only 20 ex-

[†]Experimentation with the original image resolution (320 x 240) was also performed, but there was no improvement in accuracy. However, training took a longer time.

amples per category as each object was explored 5 times and the model was trained on 4 out of the 5 objects per category.

We used Tensor-Train Gated Recurrent Unit (TT-GRU), a type of RNN, for video classification proposed by Yang *et al.* [YKT17], which has been shown to achieve results very close to the state-of-the-art networks on various video datasets, despite having a very simple architecture. To reduce the number of weight matrix parameters to be learned, TT-GRU factorizes the input-to-hidden weight matrix using Tensor-Train decomposition which is trained with the weights at the same time. For each frame, a large group of pixel inputs are mapped to the RNN as a latent vector, which is usually lower in dimensionality. This latent vector is then enriched by its predecessor at the last time step recurrently for hidden-to-hidden mapping. In this manner, the RNN is able to learn the inter-frame transition patterns to extract the representation of the entire sequence of frames, and captures the correlation between spatial and temporal patterns because the input-to-hidden and hidden-to-hidden mappings are trained simultaneously. For more details on tensor factorization models and tensor train-decomposition, see [Ose11, NPOV15].

4.3.3 Auditory Network Architecture

4.3.3.1 Sound Pre-processing

We used librosa 0.6.0 [MRL+15], a python package for music and audio analysis, to generate log-scaled mel-spectrograms of the wave files with FFT window length of 1024, hop length of 512 and 60 mel-bands. In addition to the spectrogram, we computed its derivative as a second channel using the default librosa settings. To get the fixed length input, we interpolated both channels of the spectrogram, so that the rate of the audio frames was consistent with that of the visual frames. Specifically, for each frame in a video, we interpolated 5 frames for the corresponding audio file. For example, the video of a *press* interaction has 48 frames, so we interpolated 240 (48 x 5) frames from its audio data.

4.3.3.2 Sound Network Architecture

While CNNs are largely used on image data, they have also shown strong performance in speech [AHMJ+14, AHMJP12] and music analysis [DBS11, VdODS13]. There is abundant research that demonstrates that the ability of finding local features can be successfully applied in sound classification [Pic15, OOF18, SAP17]. Therefore, we used CNN[‡] for the sound dataset depicted in Figure 4.2 and described as follows. The CNN consisted of a total of 6 learned layers including 2 convolutional ReLU, 2 max-pooling and 2 fully connected layers. The first convolutional ReLU layer consisted of 20 filters of kernel size 57 x 6 and stride 1 x 1, and max-pooling with a pool shape of 4 x 4 and stride of 4 x 4. The second convolutional ReLU layers consisted of 40 filters of kernel size 1 x 3 and stride 1 x 1, with max-pooling of shape 4 x 4 and 4 x 4 pool stride. Both the first and the second fully connected layer consisted of 256 nodes.

4.3.4 Haptic Network Architecture

4.3.4.1 Haptic Pre-processing

In our dataset, the haptic signals from 7 joints were sampled at 500 Hz. To get the fixed size input and to synchronize the haptic signals with video and sound data, we interpolated each haptic feedback to 50Hz[§]. For example, the *press* interaction takes 4.8 seconds, so we interpolated 240 (4.8 x 50) frames for each haptic signal of a *press* interaction.

4.3.4.2 Haptic Network Architecture

Several works in the literature have used CNNs to exploit the haptic signal for material classification [ZFJ+16, KANW17]. CNN performed very well because haptic feedback is expected to have temporal correlations with repeating local features in a hierarchical order of scales. For this reason, we used a CNN illustrated in Figure 4.3

[‡]Experiments were also performed using an RNN as well as a CNN-RNN combination, but both produced lower accuracy recognition rates.

[§]Experimentation with the original sampling rate (500Hz) was also performed, but there was no improvement in accuracy. However, training took a longer time.

for the haptic data, which consists of 5 layers that includes 2 convolutional ReLU, 2 max-pooling and 1 fully connected layers. The first convolutional ReLU layer's kernel dimensions are 20 x 5 with 32 filters, and the second convolutional ReLU layer has kernel size 1 x 3 and 64 filters. Both first and second max-pooling layers have a pool size of 10 x 1 and stride of 2 x 2. The fully connected layer has 1024 neurons.

4.3.5 Multimodal Network Architecture

The multimodal network inputs the same pre-processed video, audio and haptic data as described above. We used the same network architecture for each modality and in addition, added a fusion layer shown in Figure 4.4. For each modality-specific network, the last layer outputs 20 values for the 20 categories in the dataset. We activated these 20 outputs for each network using ReLU activation and concatenated them to get a layer of 60 neurons. We again activated these 60 neurons using ReLU activation and connected it to a linear layer of 20 outputs for final predictions. ReLU activation function gives a non-linear component to the network and lets the network find useful patterns, while suppressing the irrelevant features. For example, a *hold* interaction does not produce relevant sound, so the network learns to give more importance to vision and haptic feedback than audio. The multimodal network was trained from scratch which produced better results than training the modality-specific networks first, and then only training the fusion layer. We also considered combining the outputs of the modality-specific networks using a uniform combination, but using a fusion layer increased category recognition accuracy.

4.4 Evaluation and Results

4.4.1 Dataset Description

We used the publicly available dataset of the experiment performed by Sinapov *et al.* [SSS+14a], in which an upper-torso humanoid robot (shown in Figure 4.1) explored 100 different household objects belonging to 20 different categories (shown

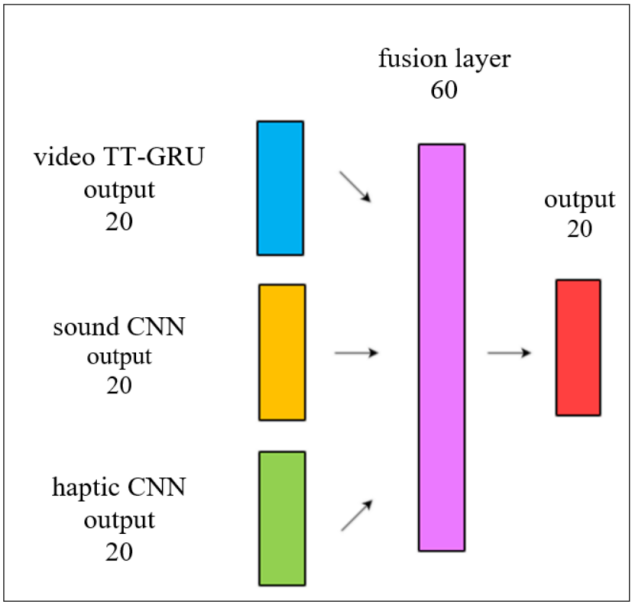


Figure 4.4: The architecture multimodal network.

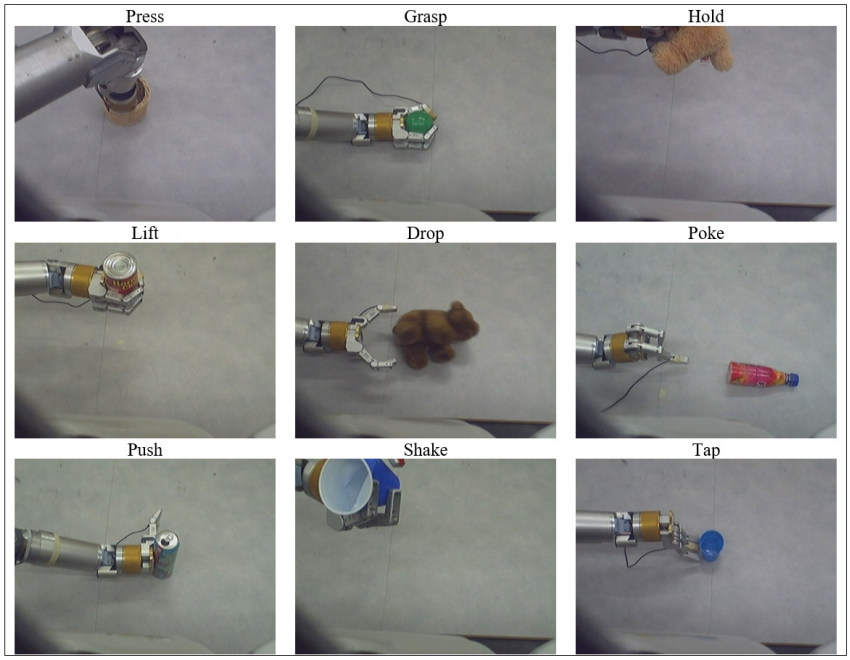


Figure 4.5: The exploratory interactions that the robot performed on all objects. From top to bottom and from left to right: (1) Press, (2) Grasp, (3) Hold, (4) Lift, (5) Drop, (6) Poke, (7) Push, (8) Shake and (9) Tap.

Figure 4.6: The robot along with the 100 objects, grouped in 20 object categories.

in Figure 4.6) using 9 exploratory behaviors performed with its left arm: Press, Grasp, Hold, Lift, Drop, Poke, Push, Shake and Tap (shown in Figure 4.5). During each interaction, the robot recorded visual feedback in the form of RGB images at 10 fps, auditory feedback in the form of a waveform at 44.1 KHz, and haptic feedback consisting of the joint-torque values sampled at 500Hz. Each behavior was performed 5 times on each object, resulting in a total of 9 x 5 x 100 = 4,500 interactions.

4.4.2 Evaluation

We evaluated how well the trained networks perform when recognizing the category of objects that are not found in the training set, via 5-fold object-based cross validation. During each round of evaluation, the training set consisted of the data from 4 objects from each category and the test set consisted of the remaining object for each category. Since the robot explored each object 5 times, there were 400 (80 x 5) examples in the training set, and 100 (20 x 5) examples in the test set. This procedure was repeated 5 times, such that each object was included four times in the training set and once in the test set. We used two metrics to evaluate the category recognition performance. The first metric was accuracy (%) as defined below:

$$Accuracy = \frac{correct\ predictions}{total\ predictions} \times 100\%.$$

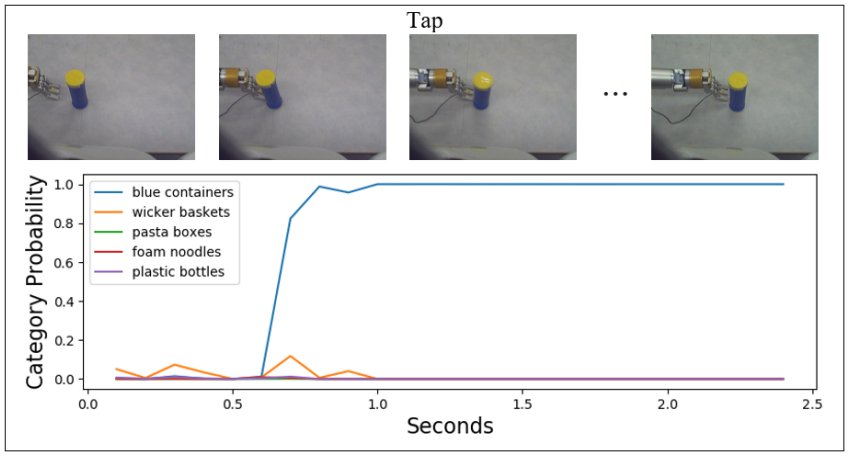


Figure 4.7: An illustrative example of the multimodal network category probability estimates as the robot performs the *tap* behavior on one of the blue container objects. The robot's category estimates converges to the correct category after about 0.7 seconds of interaction.

The second metric was the F-score, which is defined as the harmonic mean between the precision and recall for a given category label. The F-score is given by:

$$F = 2 \times \frac{precision \times recall}{precision + recall}.$$

The F-Score is always in the range of 0.0-1.0. For a given category, a high value of the F-Score indicates that the category is easy to recognize, while a low value shows the opposite.

4.4.3 Results

4.4.3.1 Illustrative Example

An example of the multimodal network category probability estimates as the robot performs a behavior on an object is shown in Figure 4.7. The robot's category estimate converges to the correct category after about 0.7 seconds of interaction. The figure plots the estimates for only 5 of the 20 categories to prevent clutter.

Table 4.1: Category recognition accuracy (%) rates for each behavior

Behavior	SVM Baseline [SSS$^+$14a]	Multimodal Network
Grasp	65.2	**71.4**
Hold	67.0	**76.8**
Lift	**79.0**	77.8
Drop	71.0	**78.0**
Poke	**85.4**	73.8
Push	**88.8**	67.4
Shake	76.8	**83.6**
Tap	**82.4**	81.6
Press	**77.4**	58.8

4.4.3.2 Accuracy Results of Category Recognition

Table 4.1 shows the accuracy for each behavior, compared with the baseline Support Vector Machine (SVM) machine learning approach presented by Sinapov *et al.* [SSS$^+$14a], which used hand-crafted auditory, haptic features, and visual features (bag-of-word SURF and a histogram of optical flow). In general, the multimodal network yields comparable performance to the baseline (chance accuracy is 5%).

In addition, we tested the accuracy of networks trained on individual sensory modalities as a function of time over the course of each interaction. For example, the *hold* behavior's duration was 1.2 seconds but we hypothesized that the robot would not need all 1.2 seconds of sensory signals to make a good prediction. Figure 4.8 shows the accuracy curve for every combination of interaction and sensory modality. The results show that for many behaviors, accurate predications can be made without needing to execute the entire behavior. This result is important as behavioral exploration of objects can be costly in terms of time and suggests that in future work, exploratory behaviors can be designed not only to maximize accuracy but also to minimize their duration such that a robot can learn object properties quicker.

4.4.3.3 F-Score Results of Category Recognition

F-scores shown in Figure 4.9 indicate which modality and behavior work better for each category. For categories in which all the objects have similar shape and color,

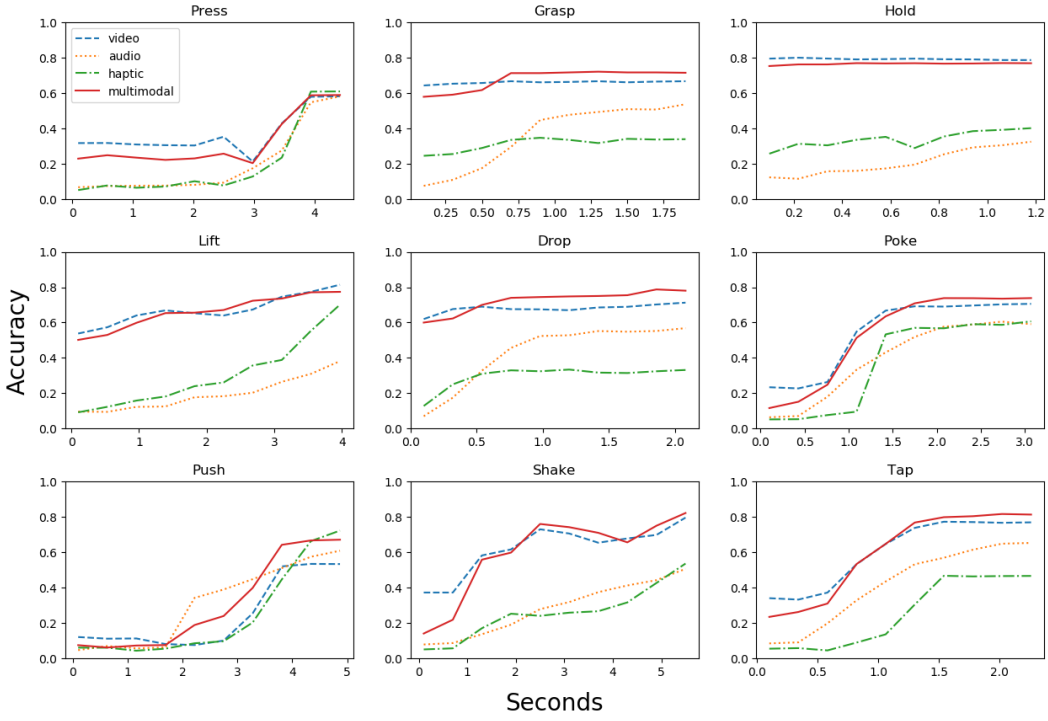


Figure 4.8: Accuracy curve for all the interactions and sensory modalities. The x-axis is duration (seconds) and the y-axis is accuracy.

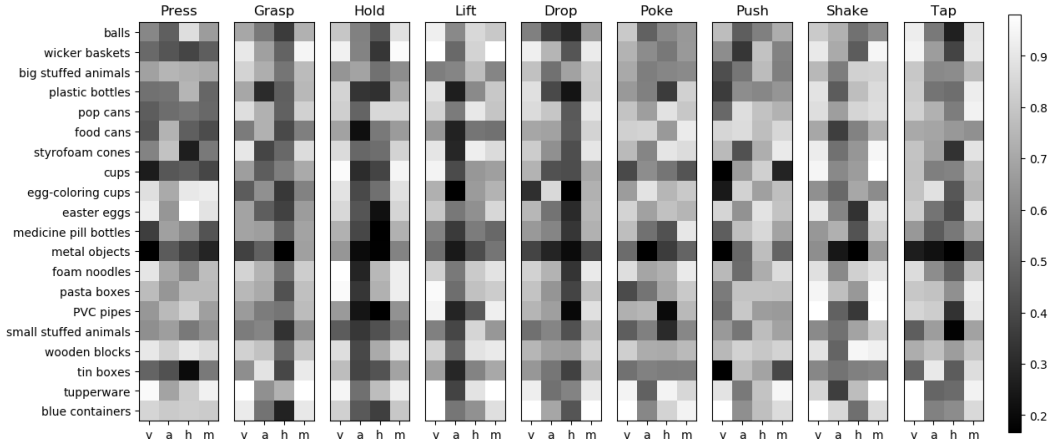


Figure 4.9: Recognition F-score for each category behavior, and sensory modality: (v)isual, (a)uditory, (h)aptic and (m)ultimodal.

the visual modality network performs better than the auditory and haptic models. For *hold* and *lift* interactions, the haptic network detects categories better than the sound network. Overall, the results show that different modalities and behaviors are relevant for different categories and suggests that robots need to purposefully select relevant actions when learning new categories.

4.5 Summary

Recognizing the category of objects is an important task for robots operating in human inhabited environments. We proposed deep learning techniques for object categorization using visual, auditory and haptic data acquired through behavioral interactions that a humanoid robot can perform on objects. We demonstrated how the robot learns to detect an object's category using a neural network for each of the sensory modalities individually. In addition, we propose a novel strategy that efficiently combines sensory modalities in a single classifier. Furthermore, unlike previous work, we showed that a robot does not need data from the entire interaction, but instead can make a good prediction early on during behavior execution.

In ongoing and future work, we are investigating the spectrum of early vs. late sensory integration in the context of category learning. In our experiments, we found that adding a fusion module, consisting of one layer, increased performance as compared to training separate modality-specific networks and combining their outputs; yet, it is an open question how deep the fusion module should be to achieve optimal performance. Another open question to be pursued in future work is how to incrementally learn new categories instead of learning all categories at the same time. The ability to acquire new categories on the fly would enable this approach to be used in grounded language learning settings, where a robot in a human inhabited environment encounters new words describing objects over time as it interacts with the people around it. Finally, to test the proposed method in a more complex scenario, we can keep multiple objects in the working space of the robot.

Chapter 5

Sensorimotor Cross-Behavior Knowledge Transfer for Grounded Category Recognition*

5.1 Introduction

From an early stage in development, humans and many other species use exploratory behaviors (e.g., shaking, lifting, pushing) to learn about objects [Pow99]. Such behaviors produce not only visual but also auditory and haptic feedback [SS08], which is fundamental to grounding the meaning of many nouns and adjectives that cannot be represented using vision alone [LC09]. For example, to perceive whether an object is full or empty, a human may lift it; to perceive whether it is soft or hard, a human may press it [Gib88]. In a sense, the behavior acts as the question which is subsequently answered by the sensory signal produced during its execution.

Recent advances in robotics have shown that robots too can use such exploratory actions for a variety of tasks, including object recognition [SBS+11], category acquisition [ANN+12], and language grounding [TSS+16]. Despite the sig-

*This chapter is based on the following paper: Gyan Tatiya, Ramtin Hosseini, Michael C. Hughes, and Jivko Sinapov, "Sensorimotor cross-behavior knowledge transfer for grounded category recognition", *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, IEEE, 2019. [THCHS19]

nificant advancement in interactive and multisensory object perception for robots [BHS+17], one challenge is that multisensory representations such as haptic, proprioceptive, auditory, and tactile perceptions cannot be easily transferred from one robot to another, as different robots may have different behaviors, bodies, and sensors. Since each robot has a unique morphology and sensor suite, each individual robot needs to learn its task-specific multisensory models of objects from scratch and cannot use models learned by a robot with different embodiment. Even in the case of two physically identical robots, it is not always possible to transfer multisensory object models as the robots' behaviors may be different.

To address these existing limitations, this chapter proposes using an encoder-decoder neural network to project sensorimotor features that the source robot has observed when interacting with an object to a semantically similar feature space that the target robot would observe when it interacts with the same object. For example, if the source robot and the target robot had observations of what the same objects feel like when grasped and shook, the pair of datasets would be used to learn a shared latent space which in turn can be used to generate observations of new objects using the source robot's observations to teach the target robot. This generated feature space can be used to train a task-specific recognition model allowing the target robot to identify objects of novel classes that it has not previously interacted with. The benefit of this approach is that the target robot would not have to learn the recognition task from scratch, but instead could use the generated features obtained from the source robot.

The proposed method is evaluated on a dataset in which a humanoid robot explored a set of 100 objects, corresponding to 20 categories using 9 exploratory behaviors while recording haptic and auditory data. The results show that certain combinations of the sensory modality and the behavior performed by the source and the target robot to learn the encoder-decoder network can generate features that achieve recognition accuracy almost as good as if the target robot learned by actually interacting with the objects.

In the context of the broader dissertation, this chapter proposes an Encoder-

Decoder based method for *Transfer using Projection to Target Feature Space* and evaluates it on knowledge transfer across different behaviors. Furthermore, in Chapter 7, the architecture proposed in this chapter is improved to support multiple source robots that performed different behaviors to generate the target robot's features.

5.2 Related Work

5.2.1 Object Exploration in Cognitive Science

Cognitive neuroscience shows that it is important for humans to interact with objects in order to learn their tactile, haptic, proprioceptive and auditory properties [Gib88, Pow99, CSS+04]. Studies show that infants start learning how objects feel, sound, and move at an early stage and this ability becomes more goal-driven as we grow older [ST99]. Research has also shown that humans are able to integrate multiple sensory modalities to recognize objects and each modality contributes to the final decision [WWCM07, EB04]. Inspired by these findings, we propose a method of knowledge transfer from the source robot to the target robot to facilitate the learning process of the target robot, as collecting multiple sensory data by interacting with objects is an expensive process.

5.2.2 Multisensory Object Perception in Robotics

While most of the object recognition methods in robotics use visual sensing, several research studies have considered multiple sensory modalities coupled with exploratory actions [BHS+17]. A number of approaches and feature extraction techniques have been proposed for recognizing objects and their properties using auditory [SWS09, EKSW18], haptic [BRK12a], and tactile feedback [SSSS11, LCL19]. Besides recognizing objects, non-visual sensory modalities have also proven useful for learning object categories [SSS+14a, HBMK16, ECK17, TS19], object relations [SKSS16], and more generally, grounding language that humans use to describe objects [RK19]. Despite all of these advances, current work in this area is limited

by the fact that each new robot is required to learn object models from scratch as different robots have different embodiment and sensors, resulting in excessive time required for individual robots to carry out the necessary object exploration, prohibiting rapid learning. In our work, we propose a method that would enable multisensory object knowledge learned by one robot to be transferred to another, thus reducing time spent on object exploration.

5.2.3 Encoder-Decoder Networks

Encoder-decoder networks consist of two feed-forward neural networks: an *encoder* and a *decoder* [HZ93, HS06]. The encoder transforms an input feature vector (the sensory input from the source robot) into a fixed-length code vector. The decoder takes a code vector as input and produces a target feature vector as output (e.g. the sensory information for the target robot). Often, encoder-decoder architectures are used for dimensionality reduction by forcing the intermediate code vector to be a much smaller size than either input or output. When input and output vectors are identical, they are referred to as "autoencoder" networks [LWL$^+$17]. When inputs and outputs differ, the more general term "encoder-decoder" applies. Encoder-decoder approaches have enjoyed success in applications such as translating sentences written in two different languages [SVL14] or learning multi-scale features for image representation tasks [KSB$^+$10]. We propose using encoder-decoder networks to predict sensorimotor features produced by an interaction with an object by one robot (the target robot) given such features produced by another robot (the source robot). Such an ability enables the target robot to use sensorimotor experience from the source robot and drastically reduce the amount of interaction and data collection needed for learning multisensory recognition models.

5.3 Learning Methodology

5.3.1 Notation and Problem Formulation

For the source robot, let $\mathcal{B}_s$ be the set of exploratory behaviors (e.g. *push*, *drop*), let $\mathcal{M}_s$ be the set of sensory modalities (e.g. *audio*, *haptic*), and let $\mathcal{C}_s$ be the set of sensorimotor contexts such that each context $c_s \in \mathcal{C}_s$ refers to a combination of a behavior $b_s \in \mathcal{B}_s$ and a sensory modality $m_s \in \mathcal{M}_s$ (e.g., each context c_s could be *push-audio*, *drop-haptic*, etc.). Similarly, for the target robot, let $\mathcal{B}_t$ be the set of exploratory behaviors, let $\mathcal{M}_t$ be the set of sensory modalities, and let $\mathcal{C}_t$ be the set of sensorimotor contexts.

For each exploration trial, the source robot and the target robot perform exploratory behaviors $b_s \in \mathcal{B}_s$ and $b_t \in \mathcal{B}_t$, respectively, on a specific object and record a sensory signal for each modality in $\mathcal{M}_s$ and $\mathcal{M}_t$, respectively. Thus, during the i^{th} exploration trial, the source robot observed features $x_i^{c_s} \in \mathbb{R}^{D_{c_s}}$ and the target robot observed features $x_i^{c_t} \in \mathbb{R}^{D_{c_t}}$. Here, D_{c_s} and D_{c_t} are the dimensions of the features observed by the source robot and the target robot, respectively, under contexts c_s and c_t.

We divide our total set of possible object categories $\mathcal{Y}$ into two mutually exclusive subsets: $\mathcal{Y}_{\text{shared}}$ and $\mathcal{Y}_{\text{source-only}}$. Categories in $\mathcal{Y}_{\text{shared}}$ are *shared*; both source and target robots have access to multiple examples from these categories during the exploration or training phase. Categories in $\mathcal{Y}_{\text{source-only}}$ are only experienced by the source robot during the training phase. The goal of our work is to effectively train the *target robot* to recognize an object at test time from one of the categories in $\mathcal{Y}_{\text{source-only}}$, even though it has never experienced any object from these categories before.

5.3.2 Knowledge Transfer Model

Our proposed encoder-decoder approach is designed to transfer knowledge from the source robot to the target robot. First, the encoder neural network transforms the observed feature vector of the source robot $x_i^{c_s}$, to a lower-dimensional, fixed-size

code vector $z_i \in \mathbb{R}^{D_z}$ of size D_z. We denote this non-linear mapping by an encoder function f: $z_i = f_\theta(x_i^{c_s})$, which takes network parameter weights θ. Next, a decoder neural network maps an input code vector z_i to create a vector of "reconstructed" target feature vector $\hat{x}_i^{c_t}$. We denote this non-linear mapping by a decoder function g: $\hat{x_i}^{c_t} = g_\phi(z_i)$, which takes network parameter weights ϕ.

Training the encoder-decoder for a context pair c_s, c_t requires observing features from both source and target robot across a set of N total objects. Given a dataset of source-target feature pairs $\{x_i^{c_s}, x_i^{c_t}\}_{i=1}^N$, we wish to find parameters $(\theta,\ \phi)$ that minimize the error between the real features $x_i^{c_t}$ observed by the target robot and the model's "reconstructed" target features $\hat{x}_i^{c_t}$ obtained by applying the encoder-decoder to the corresponding source features $x_i^{c_s}$. We use root mean square error (RMSE) as the error to minimize:

$$\theta^\star, \phi^\star = \arg\min_{\theta,\phi} \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i^{c_t} - \underbrace{g_\phi(\underbrace{f_\theta(x_i^{c_s})}_{z_i})}_{\hat{x}_i^{c_t}})^2} \tag{5.1}$$

We emphasize that the objects used to train the encoder-decoder come from the set of shared categories $\mathcal{Y}_{\text{shared}}$.

5.3.3 Category Recognition Model using Transferred Features

Given a pre-trained encoder-decoder for a source context c_s (e.g. *push-audio* or *drop-haptic*), we can train the target robot to classify objects from several categories it has never experienced before, as long as examples of these categories are seen by the source robot under context c_s. We denote this set of categories $\mathcal{Y}_{\text{source-only}}$. We assume the source robot has seen J total feature-label pairs from these categories: $\{x_j^{c_s}, y_j\}_{j=1}^J$, where $y_j \in \mathcal{Y}_{\text{source-only}}$. We can transfer this labeled dataset to the target robot by creating a "reconstructed" training set: $\{g_\phi(f_\theta(x_j^{c_s})), y_j\}_{j=1}^J$. This dataset can be used to train a standard multi-class classifier. Then, when the target robot is deployed in an environment with novel objects without category label, the

target robot can measure observed features x^{ct} and feed these features into its pre-trained classifier to predict which category within the set $\mathcal{Y}_{\text{source-only}}$ it has observed. Throughout, we will assume that at test time, only categories from $\mathcal{Y}_{\text{source-only}}$ are possible for the target robot to encounter. However, it is straightforward to extend our approach for the combined set of possible categories $\mathcal{Y}_{\text{source-only}}$ and $\mathcal{Y}_{\text{shared}}$ by combining a target robot's real and reconstructed training datasets.

5.4 Experiments and Results

5.4.1 Dataset Description

We used the dataset described in [SSS$^+$14a], in which an upper-torso humanoid robot used a 7-DOF arm to explore 100 different objects belonging to 20 different categories using 9 behaviors: *Crush*, *Grasp*, *Hold*, *Lift*, *Drop*, *Poke*, *Push*, *Shake* and *Tap* (shown in Fig. 5.1). During each behavior the robot recorded auditory and haptic feedback using two sensors: 1) an Audio-Technica U853AW cardioid microphone that captures audio sampled at 44.1 KHz, and 2) joint-torque sensors that capture torques from all 7 joints at 500 Hz. Each behavior was performed 5 times with each of the 100 objects, resulting in a total of 9 x 5 x 100 = 4,500 interactions.

We used the auditory and haptic features computed from raw sensory signals as described in [SSS$^+$14a]. For audio, the discrete Fourier transform was performed using 129 log-spaced frequency bins and a spectro-temporal histogram was computed by discretizing both time and frequencies into 10 equally spaced bins, resulting in a 100-dimensional feature vector. Haptic data was similarly discretized into 10 temporal bins, resulting in a 70-dimensional feature vector (the arm had 7 joints). Fig. 5.2 shows an example of *audio* and Fig. 5.3 shows an example of *haptic* features.

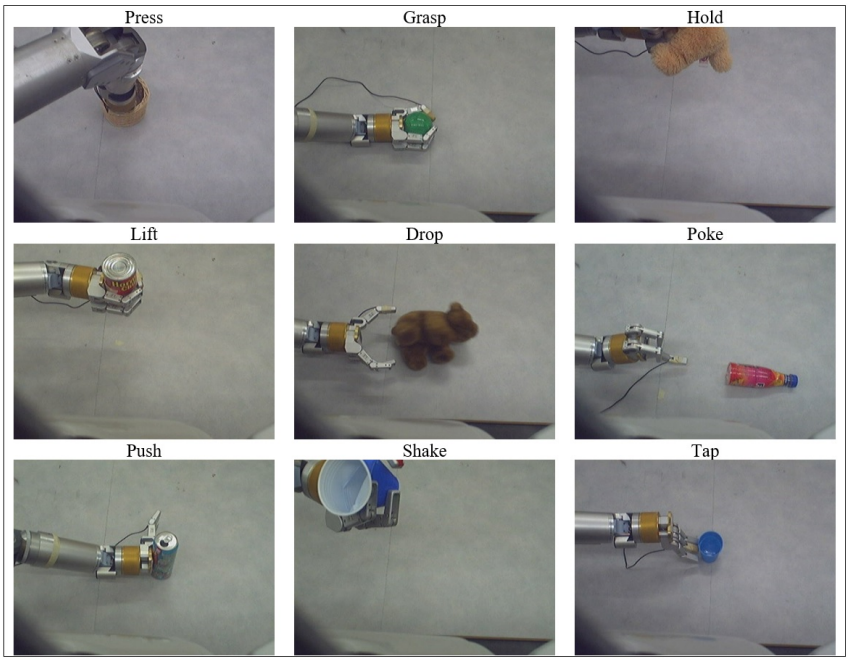


Figure 5.1: The exploratory interactions that the robot performed on all objects. From top to bottom and from left to right: (1) *Press*, (2) *Grasp*, (3) *Hold*, (4) *Lift*, (5) *Drop*, (6) *Poke*, (7) *Push*, (8) *Shake* and (9) *Tap*.

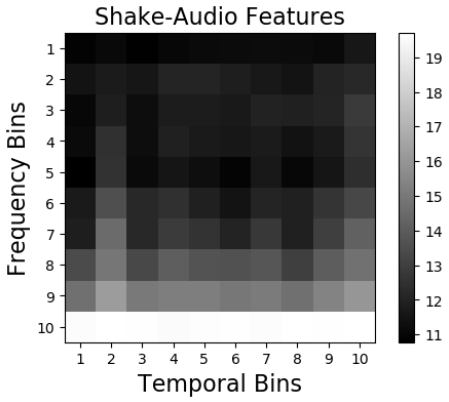


Figure 5.2: Example *audio* features using *shake* behavior performed on an object from the *medicine bottles* category.

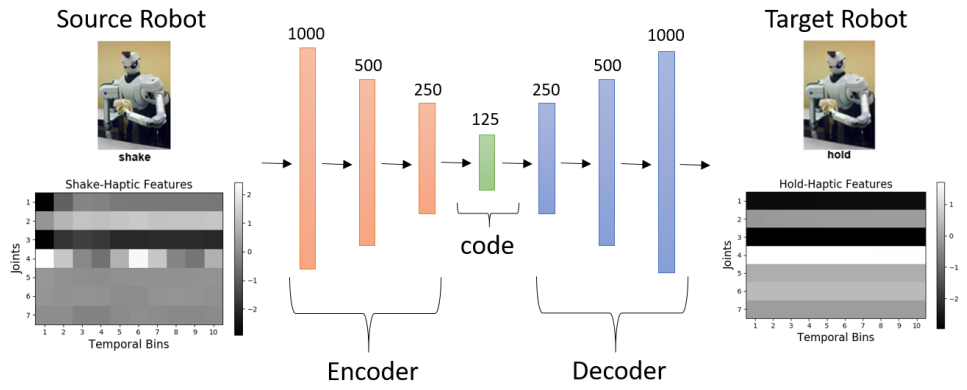


Figure 5.3: Encoder-decoder network architecture and an example of a *shake-haptic* to *hold-haptic* projection.

5.4.2 Knowledge Transfer Model Implementation

The encoder-decoder network* used consists of a multilayer perceptron (MLP) architecture of three hidden layers for both encoder and decoder, with 1000, 500, 250 hidden units and Exponential Linear Units (ELU) [CUH16] as an activation function, and a 125-dimensional latent code vector as depicted in Fig. 5.3. The network parameters are initialized randomly and updated for 1000 training epochs using Adam optimization [KB15] with learning rate 10^{-4}, and was implemented using TensorFlow 1.12 [ABC+16].

5.4.3 Category Recognition Model Implementation

At test time, we performed classification of objects into categories from the set $\mathcal{Y}_{\text{source-only}}$ via a multi-class Support Vector Machine (SVM) [Bur98]. Using the kernel trick, an SVM maps training examples to an (implicit) high-dimensional feature space where examples from different classes may be closer to linearly separable. We used the Radial Basis Function (RBF) kernel SVM implementation in the open-source scikit-learn package [PVG+11], with default hyperparameters. We also tested a k-nearest neighbors classifier (not shown) [AKA91], which performed similarly to the SVM.

*Datasets and source code for study replication are available at: `https://github.com/gtatiya/Knowledge-Transfer-in-Robots`. The experiment pipeline is visually explained and complete results of SVM and K-NN are available on the GitHub page of the study.

5.4.4 Evaluation

We assume that the source robot interacts with all 20 object categories, but the target robot interacts with only 15 randomly selected object categories. The objects of the 15 categories shared by both robots are used to train the encoder-decoder network that projects the sensory signal of the source robot to the target robot. Since the dataset we used has only one robot, we assume that the source and the target robots are physically identical, but they perform different behaviors on shared objects.[†] Subsequently, the trained encoder-decoder network is used to generate "reconstructed" sensory signals for the other 5 object categories in $\mathcal{Y}_{\text{source-only}}$ that the target robot did not interact with. Each sensory signal from objects in these categories experienced by the source robot is thus "transferred" to a target feature vector.

We consider two possible category recognition approaches: our proposed transfer-learning pipeline using the projected data from the source context (i.e., how well it would do if it transferred knowledge from the source robot), and a non-transfer ideal baseline using ground truth features produced by the target robot (i.e., the best the target robot could do if it had explored all the objects itself during the training phase). In both cases, real features observed by the target robot are used as input to the classifier at test time. We used 5-fold object-based cross-validation, where the training set consisted of 4 objects from each of the 5 categories the target robot did not interact with and the test set consisted of the remaining objects. Since the robot explored each object 5 times, there were 100 (4 objects x 5 categories x 5 trials) examples in the training set, and 25 (1 objects x 5 categories x 5 trials) examples in the test set. This procedure was repeated 5 times, such that each object was included 4 times in the training set and once in the test set.

We used two metrics to evaluate the category recognition performance of the target robot on the object categories it did not explore. First, we consider accuracy,

[†]Note that the proposed transfer learning method makes no such assumption and is applicable in situations where the two robots are physically different and/or use different feature representations for a given modality.

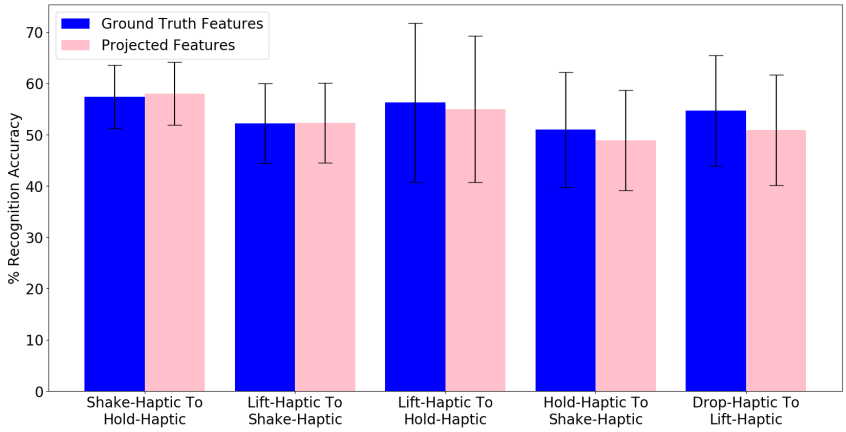


Figure 5.4: Projections where the Accuracy Delta (SVM) is minimum.

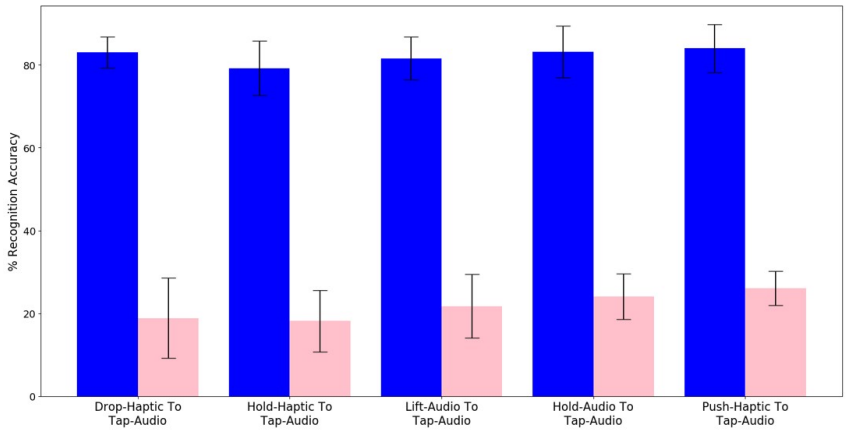


Figure 5.5: Projections where the Accuracy Delta (SVM) is maximum.

defined as $A = \frac{\text{correct predictions}}{\text{total predictions}}$ (often reported as a percentage). The process of selecting 15 categories randomly to train the encoder-decoder network, generating the features of the other 5 categories, training two classifiers using projected and ground truth features, and computing accuracy for both classifiers on ground truth features by 5-fold cross validation is repeated 10 times to compute statistics for each projection.

The second metric was accuracy delta (%), which measures the drop in classification accuracy as a result of using the projected features for training as opposed to the ground-truth features. We define this loss as $A\Delta = A_{truth} - A_{projected}$, where A_{truth} and $A_{projected}$ are the accuracies obtained when using real and projected features, respectively. Smaller accuracy delta indicates that it is easy for the source

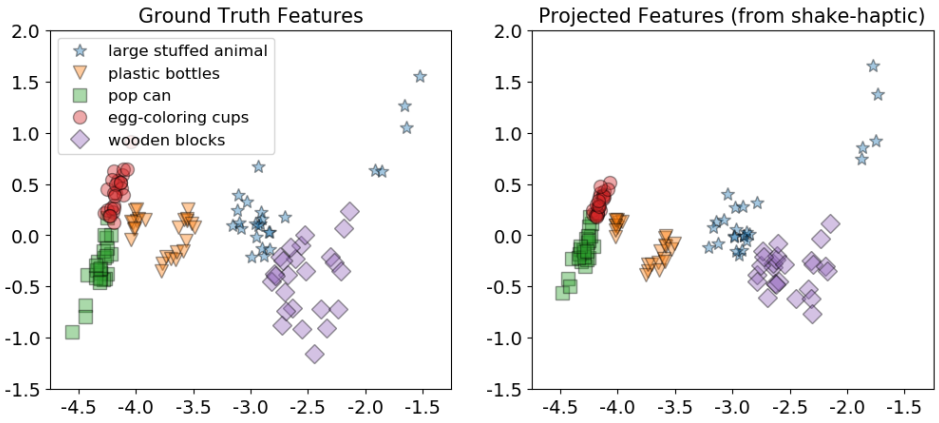


Figure 5.6: Target robot's *hold-haptic* ground truth features (left) and the projected features (right) in 2D space using Principal Component Analysis.

robot to project its sensory features in the target robot feature space, and the target robot can use these projected features to learn a classifier that can achieve comparable performance as if the target robot actually explored the objects.

5.4.5 Results

5.4.5.1 Illustrative Example

Consider the case where the source robot performs *shake* behavior and the target robot performs *hold* behavior. Projecting *haptic* features from *shake* to *hold*, enables the target robot to achieve 58% recognition accuracy[‡], compared with 57.36% when using features from real interactions (shown in Fig. 5.4). In other words, the target robot's category recognition model is as good as it would have been had it been trained on real data.

To visualize *shake-haptic* to *hold-haptic* projection, we reduced the dimension of the ground truth and the projected features of the 5 categories the target robot did not interact with into 2D space (shown in Fig. 5.6) by Principal Component Analysis implemented in scikit-learn [PVG+11]. As shown in Fig. 5.6, the clusters of projected features look very similar to the ground truth features indicating that the "reconstructed" features generated by the source robot are realistic.

[‡]Chance accuracy for 5 categories is 20%. Note that accuracy can be boosted to nearly 100% by combining multiple behaviors and sensory modalities [SS10] but this is out of scope for this chapter.

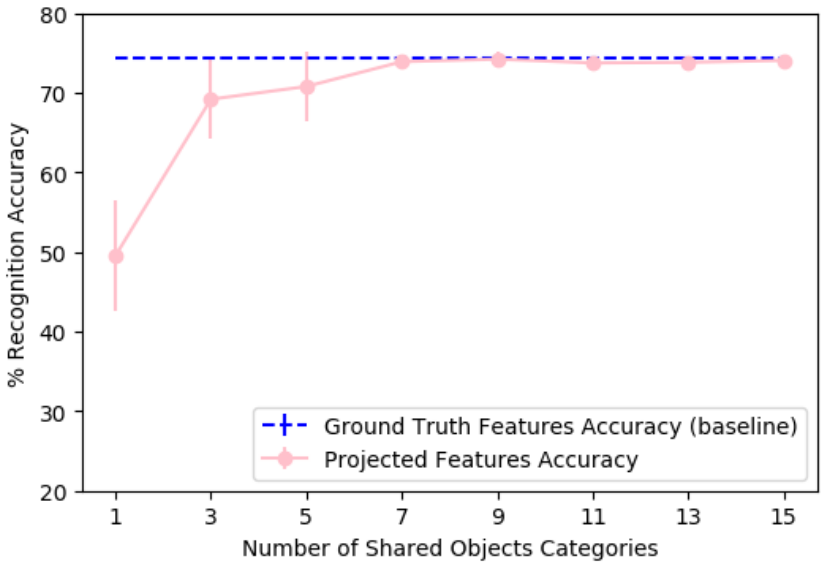


Figure 5.7: Accuracy (SVM) achieved by the target robot for different number of shared objects classifier for *shake-haptic* to *hold-haptic* projection.

To find the minimum number of object categories both robots need to interact with to train an encoder-decoder network that achieves good performance, we varied the number of shared categories for *shake-haptic* to *hold-haptic* projection. As shown in Fig. 5.7, performance saturates at about 7 shared object categories (i.e., using 5 objects per class, the robot needs 35 shared objects out of 100 possible).

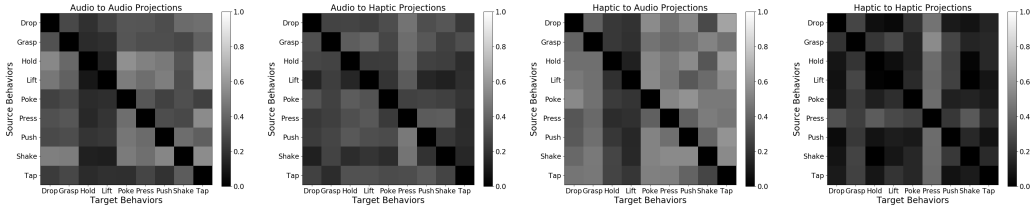


Figure 5.8: Accuracy Delta (SVM) for 4 mappings: *audio* to *audio*, *audio* to *haptic*, *haptic* to *audio*, *haptic* to *haptic*. Darker color means smaller Accuracy Delta (better) and lighter color means larger Accuracy Delta (worse).

5.4.5.2 Accuracy Results of Category Recognition

Since there are 2 modalities (*audio* and *haptic*) there are 4 possible mappings from the source to the target robot: *audio* to *audio*, *audio* to *haptic*, *haptic* to *audio*, and *haptic* to *haptic*. Each of the 9 behaviors are projected to all of the other 8

behaviors, so for each mapping, there are 72 (9 x 8) projections. Fig. 5.4 shows the 5 projections where the accuracy delta is minimum, and Fig. 5.5 shows the 5 projections where the accuracy delta is maximum among all 288 (4 x 72) projections.

Overall, mappings within same modality (*audio* to *audio* and *haptic* to *haptic*) achieve higher accuracy than mapping to a different modality. This is intuitive, as knowing what an object feels like when performing a behavior can inform what it would feel like better than what it will sound like given another behavior.

5.4.5.3 Accuracy Delta Results

Fig. 5.8 shows the accuracy delta for all 4 possible modality mappings. Darker color indicates smaller accuracy delta, thus the diagonal is black as there is no accuracy drop when both robots perform the same behavior. Comparatively, *haptic* to *haptic* projections achieve smallest accuracy delta. *Audio* to *audio* is the second best performing mapping, indicating that mappings within the same modality achieve less accuracy delta. Some specific projections that support this observation are shown in Fig 5.4. However, when both robots perform actions using different modalities, the accuracy delta is relatively higher. For example, *drop-haptic* to *tap-audio* and *hold-haptic* to *tap-audio* are the two projections where the accuracy delta is highest.

When both robots perform behaviors that capture similar object properties, the projected features are more realistic. For example, lifting an object provides a good idea how it would feel to hold that object as indicated by smaller accuracy delta. Producing *hold-audio* features from most of the source robot's features is an easy task, possible because holding an object does not produce much sound.

The relation between the RMSE loss of features used to train the encoder-decoder network and the accuracy delta is shown in Fig. 5.9 for all of the mappings. RMSE is the Euclidean distance between the ground truth and the projected features. Each dot in the plot corresponds to a projection from the source to the target robot. Generally, the accuracy delta increases with the increase in RMSE loss. This means when the "reconstructed" features are more realistic, the accuracy delta is expected to be smaller, and as the reconstruction gets worse, the accuracy delta

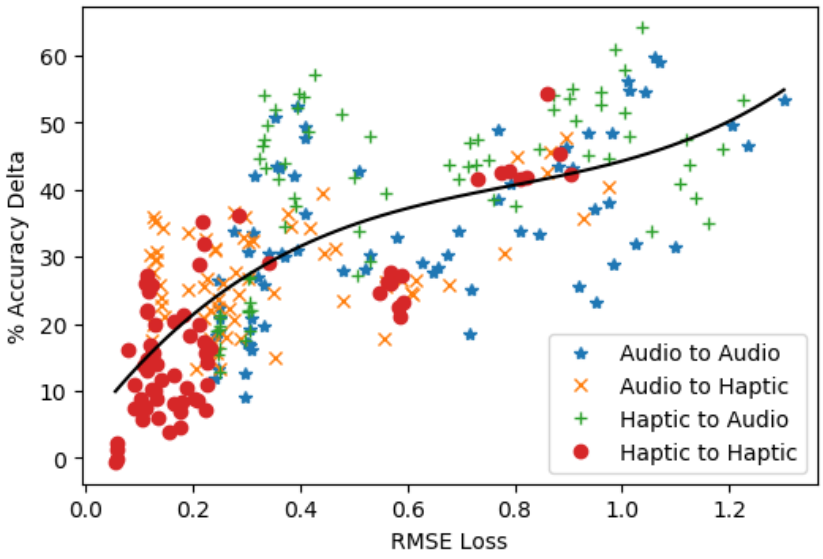


Figure 5.9: Relation between RMSE Loss of the features on the training set and Accuracy Delta (SVM) computed using the trained encoder-decoder network. The solid line represents a polynomial with degree 3 that fits all the dots.

increases.

5.5 Summary

Non-visual sensory object knowledge is specific to each robot and depends on its unique embodiment, sensors, and actions. We proposed a framework for knowledge transfer that uses an encoder-decoder network to project sensory features from one robot to another robot across different behaviors. The framework enables a target robot to use knowledge from a source robot to classify objects into categories it has never seen before. In this way, the target robot does not have to learn a classifier from scratch, but instead starts immediately with a model nearly as accurate as what can be achieved if the target robot could afford to collect its own labeled training set via exploration. This result addresses some of the biggest challenges in deploying behavior-grounded multi-sensory perception models, namely that they require a lot of interaction data to train and cannot be easily transferred from one robot to another.

In future work, we will test our proposed framework on robots that not

only perform different actions, but also are morphologically different and use unique feature representations. Extending the framework to allow for more than two robots to share information is also an outstanding challenge which has the potential to enable any new robot to use multi-sensory knowledge transferred from other robots that had previously interacted with a shared set of objects.

Chapter 6

Haptic Knowledge Transfer Between Heterogeneous Robots using Kernel Manifold Alignment*

6.1 Introduction

To recognize objects and their properties, humans use a variety of non-visual sensory modalities coupled with exploratory behaviors. While robots can use vision to recognize the shape and color of an object, camera input alone cannot determine its haptic and tactile properties, such as whether it is soft or hard, or whether it is full or empty. To perceive non-visual information, a robot must interact with the object and interpret the feedback to detect the object's characteristics. Previous works have indeed shown that robots can use non-visual sensory feedback of interaction with objects such as haptic, tactile, and/or auditory senses to perform tasks, including object recognition, object category acquisition, and language grounding (see [BHS+17, LKS+20] for a review).

A major challenge when learning non-visual object representations is that

*This chapter is based on the following paper: Gyan Tatiya, Yash Shukla, Michael Edegware, and Jivko Sinapov, "Haptic knowledge transfer between heterogeneous robots using kernel manifold alignment", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2020. [TSES20]

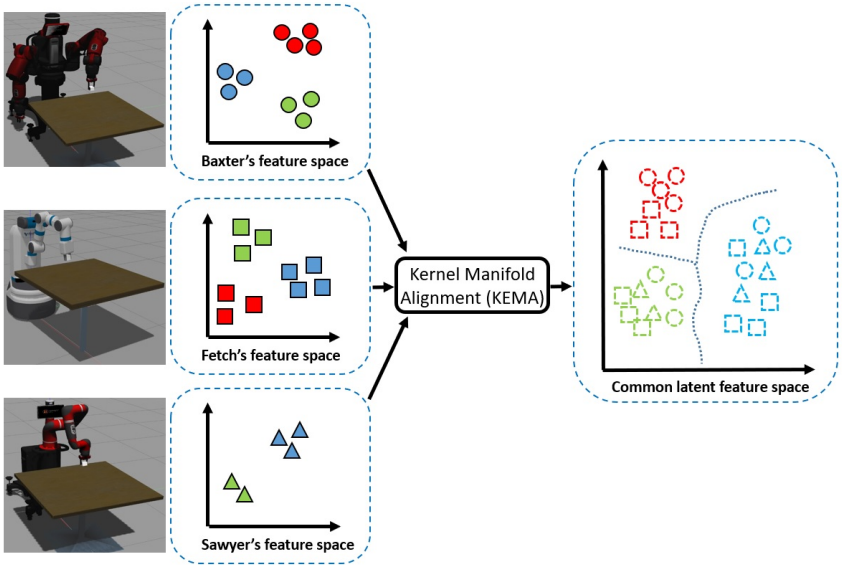


Figure 6.1: Overview of the proposed framework. Feature space of different robots depict datapoints collected during object interaction. Each shape represents a robot and each color represents an object. Once each datapoint is projected into a common latent space, the decision function for a classifier is grounded in the latent space rather than the robot's own feature space.

each robot requires excessive time to perform the necessary object exploration for data collection, which prohibits rapid learning and makes it difficult to deploy non-visual object representations in practice. There is no general purpose sensory knowledge representations for non-visual features as different robots have different embodiments and sensors. As a result, it is not easy to transfer knowledge of non-visual object properties from one robot to another, so each individual robot needs to learn its task-specific sensory models from scratch.

To address this challenge, we propose a framework for haptic knowledge transfer, shown in Fig. 6.1, using kernel manifold alignment (KEMA) for sharing knowledge between multiple, heterogeneous robots. Our method projects the sensorimotor features of object interaction from multiple robots into a common latent space and use this latent space to train the recognition models to solve various tasks, as opposed to using each robot's own sensorimotor feature space. To test our method, we collected a dataset of 3 simulated robots that performed 4 behaviors on 25 objects, and we used this dataset to transfer knowledge from two source robots

to a target robot for training the target robot with less examples. The results of our experiments show that robots can bootstrap their haptic object perception skills by leveraging experience from other robots in a way that speeds up learning and allows the target robot to recognize novel objects that it has not interacted with before test time.

In the context of the broader dissertation, this chapter proposes a KEMA-based method for *Transfer using Projection to Shared Latent Feature Space* and evaluates it on the dataset collected by three simulated robots described in Chapter 3. Additionally, the method proposed in this chapter serves as the baseline for comparison with methods proposed in Chapter 7, Chapter 8, and Chapter 9.

6.2 Related Work

Research in psychology and cognitive science has highlighted the significance of multiple sensory modalities used by humans to recognize objects [WWCM07, EB04] and interact with them in order to learn their haptic and tactile properties [Gib88]. Traditionally, object recognition approaches are based solely on the visual modality. More recently, several lines of recent research have proposed integrating exploratory actions with haptic modality, which has also been shown useful for learning object categories [SSS$^+$14a, HBMK16, ECK17, TS19, JLWS19, BGS$^+$20], object relations [SKSS16, TSS$^+$20], and grounding language used to describe objects [CMR$^+$15, TSMS18, RK19]. A remaining challenge is that non-visual sensory representations cannot be easily transferred from one robot to another, as each robot has a unique embodiment in terms of its morphology and sensor suite. As a result, each robot must interact with objects to learn its models from scratch. This work presents a knowledge-transfer framework for multiple robots that enables them to not only recognize objects with less interactions, but also to recognize novel objects without exploratory training.

To transfer knowledge, Tatiya *et al.* [THCHS19] proposed using encoder-decoder neural network to project sensorimotor features from a source robot's fea-

ture space to a target robot's feature space, allowing the target robot to classify novel objects into categories using the source robot's knowledge. One limitation was that the dataset used contained only a single robot, and thus they transferred knowledge between two physically identical robots across different behaviors. Furthermore, the method proposed would work only for two robots: the source and the target. To deal with these shortcomings, we propose a method that enables more than two robots of different embodiments to project their sensory features into a common latent space, such that the decision function for a given recognition task is grounded in the latent space rather than each individual robot's own feature space.

Domain adaptation is a transfer learning method that deals with shifts in the feature spaces of a source domain (training set) and a different but related target domain (test set). The main goal of such methods is to reduce the domain shift so that a machine learning classifier trained on the source domain can make better predictions about the target domain. Manifold alignment is a domain adaptation strategy that aligns datasets and projects them into a common latent space. Manifold alignment preserves the local geometry of each manifold and captures the correlations between manifolds, which allows knowledge transfer from one domain to another. The projected datapoints are comparable and can be used to train a single classifier for different domains.

We propose to use the kernel manifold alignment (KEMA) [TCV16] for domain adaptation, which can align an arbitrary number of domains of different dimensionality without needing paired examples. KEMA [TCV16] has been successfully applied to visual object recognition [TCV16], facial expression recognition [TCV16], and human action recognition [LLL$^+$18]. However, KEMA has never been applied to the haptic data that robots can use for object recognition. We evaluated the performance of KEMA to adapt the sensory signals of multiple robots and obtain their aligned feature representations in a common latent space.

6.3 Learning Methodology

6.3.1 Notation and Problem Formulation

Let a robot perform a set of exploratory behaviors (e.g., *grasp*, *pick*), $\mathcal{B}$, on a set of objects, $\mathcal{O}$, while recording a non-visual sensory modality m (e.g. *effort*). Let the robot perform each behavior n times on each object. Let us consider $\mathcal{R}$ such robots' datasets with $\mathcal{B}_r$, m_r and n_r, where $r = 1, ..., R$. Each robot interacts with the same set of objects $\mathcal{O}$. During the i^{th} exploratory trial, the robot r observation feature is represented as $x_r^i \in \mathbb{R}^{D_r}, i = 1, ..., n_r$ where D_r is the dimensionality of the feature space for robot r.

Our main goal is to learn a common latent feature space for all the $\mathcal{R}$ robots, such that the robots can be trained to recognize objects in that latent space, as opposed to each robot's own feature space. This will enable an individual robot to use the observation features collected by other robots to learn a recognition model and perform better than a model trained only using its own observation features. In addition, learning a common latent feature space would also enable a robot to recognize objects it has never interacted with, as long as other robots have. While learning the latent space, it is assumed that all the robots perform the same behavior and interact with the same set of objects.

6.3.2 Kernel Manifold Alignment (KEMA)

KEMA [TCV16] extended the work of Wang *et al.* [WM11] by kernelization of the original data by transforming it into a high dimensional Hilbert space $\mathcal{H}$ with the mapping function $\phi(.) : x \mapsto \phi(x) \in \mathcal{H}$ to ensure that the transformed data is linearly separable. Due to the high dimensional feature space, the computational load would increase significantly and thus, kernel trick is used in which the problem is expressed in terms of dot products within $\mathcal{H}$. A Kernel function $K_{ij} = K(x_i, x_j) =< \phi(x_i), \phi(x_j) >$ is used to compute the kernel matrix that encodes the similarity between training examples using pair-wise inner products between mapped examples without computing $\phi(.)$ directly. We adopted Radial Basis Function (RBF) kernel

as the kernel function. As there are multiple robots, R different robots' datasets are mapped into R different Hilbert spaces of dimension $\mathcal{H}_r, \phi_r(.) : x \mapsto \phi_r(x) \in \mathcal{H}_r, r = 1, ..., R$.

KEMA constructs a set of domain-specific projection functions, $\mathcal{F} = [f_1, f_2, ...f_R]^T$ that project data from R robots into a common latent space such that the examples of a same object class would locate closer while examples of different object classes would locate distantly. To achieve this, KEMA finds the data projection matrix $\mathcal{F}$ that minimizes the following cost function:

$$\begin{aligned} \{f_1, f_2, ...f_R\} &= \underset{f_1, f_2, ...f_R}{\arg\min} \left(C(f_1, f_2, ...f_R)\right) \\ &= \underset{f_1, f_2, ...f_R}{\arg\min} \left(\frac{\mu GEO + (1-\mu)SIM}{DIS}\right) \end{aligned} \tag{6.1}$$

where geometry (GEO) and class similarity (SIM) terms are minimized and class dissimilarity (DIS) term is maximized. The parameter $\mu \in [0, 1]$ controls the contribution of the geometry and the similarity terms. The three terms are explained as follows:

1. **Geometry (GEO)** is a matrix that represents the geometry of a domain. GEO is minimized to preserve the local geometry of each domain by penalizing projections in the input domain that are far from each other:

$$\begin{aligned} GEO &= \sum_{r=1}^{R} \sum_{i,j=1}^{n_r} W_g^r(i,j) \left\| f_r^T \phi_r(x_r^i) - f_r^T \phi_r(x_r^j) \right\|^2 \\ &= tr(F^T \Phi L_g \Phi^T F) \end{aligned} \tag{6.2}$$

where W_g^r in a similarity matrix representing the similarity between x_r^i and x_r^j, which is typically computed by k-nearest neighbor graph (k-NNG). $L_g \in \mathbb{R}^{(\sum_r n_r) \times (\sum_r n_r)}$ is a graph Laplacian matrix computed by $L_g = D_g - W_g$, where D_g is a diagonal matrix with entries $D_g(i,i) = \sum_j W_g(i,j)$.

2. **Similarity (SIM)** is a matrix that represents the class similarity of a

domain. SIM is minimized to encourage examples with the same object class to be located close to each other in the latent space by penalizing projections of the same object class far from each other:

$$\begin{aligned} SIM &= \sum_{r,r'=1}^{R} \sum_{i,j=1}^{n_r,n_{r'}} W_s^{r,r'}(i,j) \left\| f_r^T \phi_r(x_r^i) - f_{r'}^T \phi_{r'}(x_{r'}^j) \right\|^2 \\ &= tr(F^T \Phi L_s \Phi^T F) \end{aligned} \tag{6.3}$$

where $W_s^{r,r'}$ in a similarity matrix that has components set to 1 if the two examples from robots r and r' belong to the same object class, and 0 otherwise. The graph Laplacian matrix is computed by $L_s = D_s - W_s$, where D_s is a diagonal matrix with entries $D_s(i,i) = \sum_j W_s(i,j)$.

3. **Dissimilarity (DIS)** is a matrix that represents the class dissimilarity of a domain. DIS is maximized to encourage examples with different object classes to be located far apart in the latent space by penalizing projections of the different object class that are close to each other:

$$\begin{aligned} DIS &= \sum_{r,r'=1}^{R} \sum_{i,j=1}^{n_r,n_{r'}} W_d^{r,r'}(i,j) \left\| f_r^T \phi_r(x_r^i) - f_{r'}^T \phi_{r'}(x_{r'}^j) \right\|^2 \\ &= tr(F^T \Phi L_d \Phi^T F) \end{aligned} \tag{6.4}$$

where $W_d^{r,r'}$ in a dissimilarity matrix that has components set to 1 if the two examples from robots r and r' belong to different objects, and 0 otherwise. The graph Laplacian is computed by $L_d = D_d - W_d$, where D_d is a diagonal matrix with entries $D_d(i,i) = \sum_j W_d(i,j)$. By combining Eqs. (6.2), (6.3), and (6.4), the optimization problem can be formulated as:

$$\underset{f_1,f_2,...f_R}{\arg\min}\, tr\left(\frac{F^T \Phi(\mu L_g + (1-\mu) L_s) \Phi^T F}{F^T \Phi L_d \Phi^T F} \right) \tag{6.5}$$

The latent features that minimize the cost function $C(f_1, f_2, ...f_R)$ are given

by the eigenvectors corresponding to the last eigenvalues of the generalized eigenproblem derived from Eq. (6.5) [WM11]:

$$\Phi(\mu L_g + (1-\mu)L_s)\Phi^T F = \lambda \Phi L_d \Phi^T F \tag{6.6}$$

where Φ is a block diagonal matrix containing the datasets $\Phi_r = [\phi_r(x_1), ..., \phi_r(x_{n_r})]^T$, F contains the eigenvectors organized in rows for the particular domain defined in Hilbert space $\mathcal{H}_r$, where $\mathcal{F} = [f_1, f_2, ...f_H]^T$, $H = \sum_{r=1}^{R} H_r$, and λ is the eigenvalues of the generalized eigenproblem. F is in a high dimensional space that might be costly to compute. Thus, the eigenvectors are expressed as a linear combination of mapped examples using the Riesz representation theorems [RN55] as $f_r = \Phi_r \alpha_r$ (or $F = \Phi\Lambda$ in matrix notation). By multiplying both sides by Φ^T in Eq. (6.6) and replacing the dot products with the corresponding kernel matrices, $K_r = \Phi_r^T \Phi_r$, the final problem is formalized as:

$$K(\mu L_g + (1-\mu)L_s)K\Lambda = \lambda K L_d K \Lambda \tag{6.7}$$

where K contains kernel matrices K_r in a block diagonal form. The projection matrix Λ can be expressed in a block structure of size $n \times n$:

$$\Lambda = \begin{bmatrix} \boldsymbol{\alpha_1} \\ \vdots \\ \alpha_R \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha_{1,1}} & \boldsymbol{\cdots} & \boldsymbol{\alpha_{1,n}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\alpha_{n_1,1}} & \boldsymbol{\cdots} & \boldsymbol{\alpha_{n_1,n}} \\ \alpha_{n_1+1,1} & \cdots & \alpha_{n_1+1,n} \\ \vdots & \ddots & \vdots \\ \alpha_{n,1} & \cdots & \alpha_{n,n} \end{bmatrix} \tag{6.8}$$

where the eigenvectors are highlighted in bold for the first domain, and $n = \sum_r n_r$ is the total number of examples in the kernel matrices. A new test example x_r^i can be projected to the new latent space by first mapping it to its corresponding kernel form K_r^i and then applying the corresponding projection vector α_r formulated as:

$$P(x_r^i) = f_r^T \Phi_r^i = \alpha_r^T \Phi_r^T \Phi_r^i = \alpha_r^T K_r^i \tag{6.9}$$

where K_r^i is a kernel evaluations vector between example x_r^i and all examples of rth robot used to compute the projections α_r. For more details on KEMA, readers can refer [TCV16, WM11].

6.3.3 Object Recognition Model using Latent Features

Once the data is transferred to the latent space from multiple robots, we used the transferred data on the latent manifold to train a multi-class Support Vector Machine (SVM) [Bur98] model with the RBF kernel to recognize different object classes. We trained two types of models: speeding up object recognition model and novel object recognition model.

To build the manifold alignment for the speeding up object recognition model, we used two source robots that are assumed to have explored the objects extensively and one target robot that is assumed to have relatively less experience with objects. To train this model, we used the transferred data from all the robots, but incrementally varied the number of examples per object used for the target robot. To test this model, we used the examples of the target robot that were not used to build the manifold alignment.

To build the manifold alignment for the novel object recognition model, we used two source robots that are assumed to have explored all the objects and one target robot that is assumed to have never explored a few objects. To train this model, we used the transferred data from two source robots of the objects that the target robot never explored. To test this model, we used the examples of the objects that are novel to the target robot.

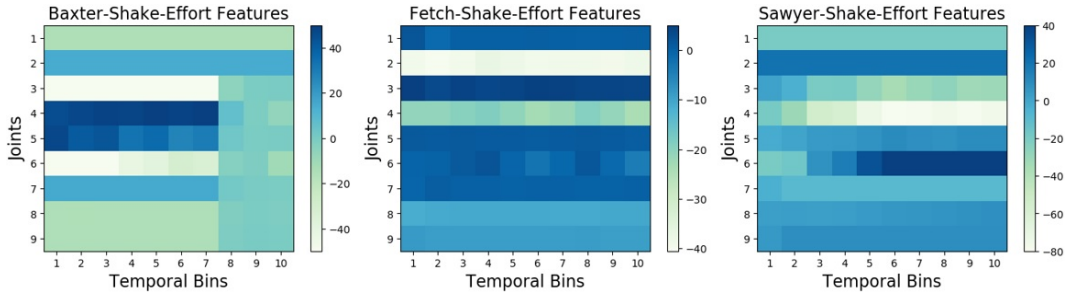


Figure 6.2: Examples of *effort* features using *shake* behavior performed on an 0.62 kg block object by *Baxter*, *Fetch*, and *Sawyer* (right to left).

6.4 Evaluation

6.4.1 Data Collection and Feature Extraction

A dataset was collected in which 3 simulated robots (*Baxter*, *Fetch* and *Sawyer*) perform 4 behaviors (*grasp*, *pick*, *shake* and *place*) on 25 block objects (each vary by weight from 0.01 kg to 1.5 kg). The behaviors of each robot were encoded as joint-space trajectories where the joint values are randomly sampled within a specified range of joint values for each joint of the robot. Thus, each interaction of the robot is expected to be different, which is what we would expect in the real world. During each behavior the robots recorded *effort* feedback from all joints [*]. Each behavior was performed 100 times on each object, resulting in a total of 10,000 examples (4 behaviors x 25 objects x 100 trials) per robot. Effort data was discretized into 10 temporal bins, where each bin consists of mean of effort values in that bin. Fig. 6.2 visualizes examples of effort features of all the robots.

6.4.2 Evaluation

To evaluate the performance of manifold alignment for knowledge transfer, we considered two tasks. In the first task, the target robot has less interaction with objects, and in the second task, the target robot has never interacted with a few objects. In both tasks, we assume both source robots have explored all the objects extensively.[†]

[*]The sampling rate of *Baxter* is 50Hz, and *Fetch* and *Sawyer* is 100Hz. All the robot's arm have 9 joints including 2 grippers.

[†]Datasets, source code and complete results for study replication are available at: `https://github.com/gtatiya/Haptic-Knowledge-Transfer-KEMA`.

6.4.2.1 Speeding up object recognition

In this task, the main goal is to improve the object recognition performance of the less experienced target robot, by aligning the data from all the 3 robots, and then using this aligned data to train the target robot. For the baseline condition, the target robot is trained to recognize objects by using its own data collected during object interactions. For the transfer condition, the target robot is trained to recognize objects by using the aligned data in the latent feature space corresponding to all the 3 robots. We incremented the number of examples per object used to train the target robot from 1 to 80, and we used the held-out 20 examples for testing. For both conditions, we performed 5-fold cross validation such that each example is included in test set once and computed accuracy $A = \frac{\text{correct predictions}}{\text{total predictions}}\%$, and reported average accuracy of all the folds.

6.4.2.2 Novel object recognition

In this task, the goal is to enable the target robot to recognize n objects it never interacted with. Both source robots interact with all the 25 objects, while the target robot interacts with only $25 - n$ randomly selected objects. The $25 - n$ objects shared by all 3 robots are used to build the manifold alignment that transfers the sensory signal of the robots to the latent space. Then a classifier is trained using the transferred data of the source robot corresponding to the objects that are novel to the target robot. Subsequently, to test this classifier, the transferred data of the n objects that the target robot did not interact with is used that were not used to build the alignment. Similar to speeding up object recognition, we reported the accuracy of this classifier to evaluate its performance and compared it with the chance accuracy of the classifier. The process of selecting $25 - n$ objects randomly to build the manifold alignment, training the classifier using transferred data of the source robots and testing the classifier on n novel objects was repeated 10 times to produce an accuracy estimate.

6.5 Results

6.5.1 Illustrative Example

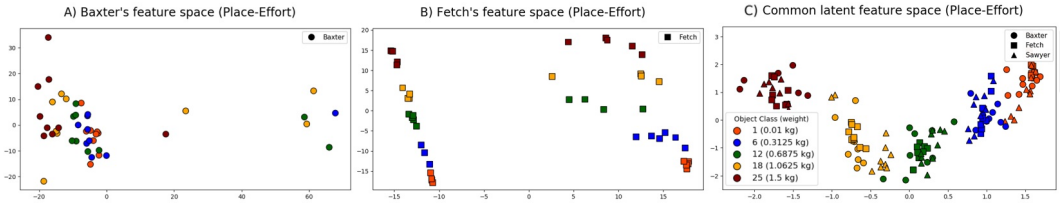


Figure 6.3: Original sensory features of (A) Baxter and (B) Fetch for *place-effort* performed on 5 objects in 2D space, and first 2 dimensions of corresponding features in the common latent feature space (C).

Consider the case where the 3 robots perform the *place* behavior on all 25 objects 10 different times while recording *effort* signals, which were used to build the manifold alignment using KEMA and generate latent features. We plotted the first two dimensions of the latent features, and reduced the dimensionality of the original sensory signal to 2 by Principal Component Analysis. As shown in Fig. 6.3, the datapoints collected by the 3 robots of 5 different objects are clustered together in the common latent space.

6.5.2 Speeding up object recognition results

Fig. 6.4 shows the object recognition performance, where *Baxter* and *Sawyer* serve as the source robots and *Fetch* serves as the target robot. To build the manifold alignment, we incrementally varied the number of interactions of the target robot from 1 to 80, and to test the classifier, held-out 20 examples are used. Note that to choose the amount of source robot data for building alignment and number of dimensions of latent features used to train the model, we performed a grid search, in which we experimented with different amount of source robot data and different number of dimensions and used the optimal parameters for the final results. Generally, if the target robot interacts less with objects, using more source robots' data generates better latent features, and using the first 1 or 2 dimensions of the latent features achieves high accuracy as they are the most correlated dimensions among

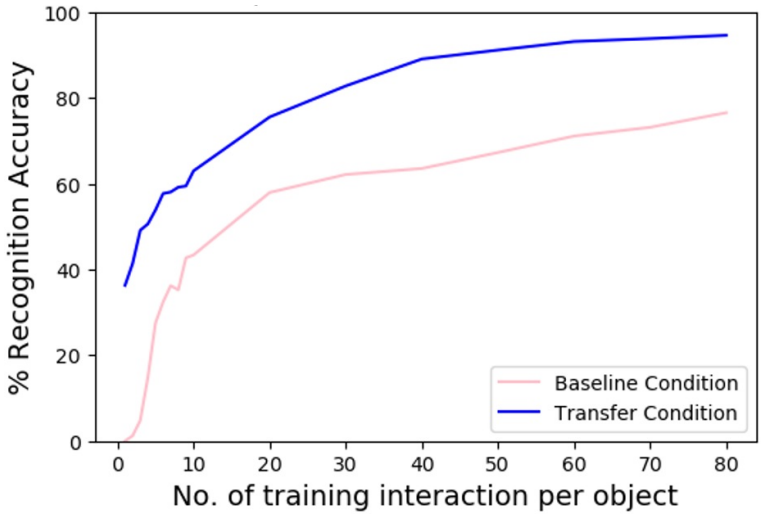


Figure 6.4: Accuracy of the baseline and transfer conditions, where *Fetch* serves as the target robot, and *Baxter* and *Sawyer* serve as the source robots.

all the robots.[‡] Fig. 6.4 compares the recognition accuracy of the baseline condition, where the target robot learns to recognize objects using only its own features, and the transfer condition, where the target robot learns to recognize objects using its own as well as the source robots' latent features. In both conditions, the recognition accuracy is computed by performing a weighted combination of all the behaviors based on their performance on the training examples.

For most behaviors, the transfer condition performs consistently better than the baseline condition. A significant boost in performance is observed with a fewer number of the target robot's interactions per object. Fig. 6.4 shows that by performing all the behaviors with each object only once, the target robot achieves around 0% accuracy in the baseline condition, whereas it achieves 36.28% accuracy in the transfer condition. This result indicates that in cases where the target robot has limited time to learn the task, transferring knowledge from other robots can speed up as well as improve the classification performance. We also experimented with *Baxter* and *Sawyer* as the target robot, and the other 2 robots as the source robot, and observed similar boost in performance in the transfer condition.

[‡]Note that using entire source robots' data and latent features for training the target robot did not perform better than using optimal amount of source robot data and number of latent features.

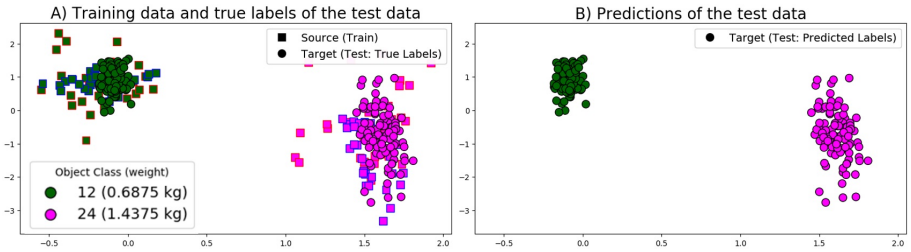


Figure 6.5: Visualization of the training and testing datapoint used to train the target robot (*Fetch*) to detect 2 novel objects in 2D space. (A) shows the training data in squares corresponding to the source robots (*Baxter* and *Sawyer*) latent features of *place* behavior, and the test data in circles corresponds to the true labels of the target robot (*Fetch*). (B) shows the predictions of the test data, which is 100% correct.

6.5.3 Novel object recognition results

For a case where the *Fetch* robot has not interacted with 2 of the objects, we trained a classifier using the latent features of the source robots (*Baxter* and *Sawyer*) performing the *place* behavior on those objects. Fig. 6.5 visualizes the data used to train and test the classifier. In Fig. 6.5A, squares with blue and red outline show the source robots' training data and circles show the true labels of the target robot's data used to test the classifier. Each color represents a different object. Fig. 6.5B shows the predictions of the classifier, which is able to correctly classify 100% of the test data.

Fig. 6.6 shows the results when the target robot (*Fetch*) was trained to recognize 2 and 5 novel objects by incrementing the number of objects explored by the target robot used to build the manifold alignment. To build the manifold alignment, 30% of the source robots' data (*Baxter* and *Sawyer*) was used. In most cases, the target robot achieves better than chance accuracy, and as the target robot interacts with more objects, its performance to recognize novel objects improves. Thus, the target robot can learn to recognize objects it never interacted with by using the knowledge transferred by the source robots. Similar results were observed when the *Baxter* and *Fetch* serve as the target robot.

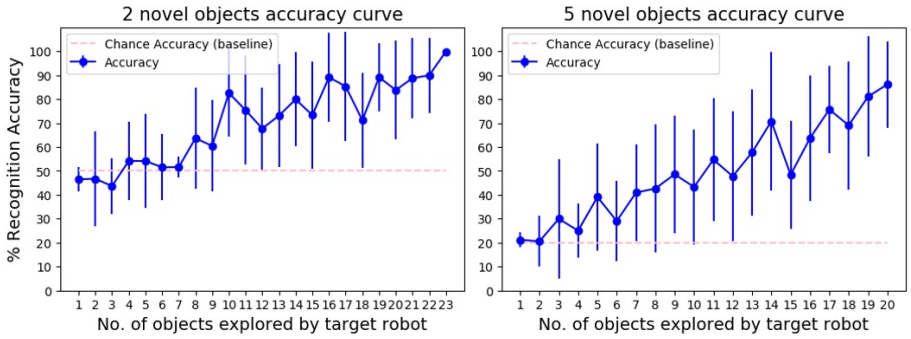


Figure 6.6: Accuracy curve of the target robot (*Fetch*) for detecting 2 and 5 novel objects (left to right) for different number of objects explored by it using the knowledge transferred by the source robots (*Baxter* and *Sawyer*).

6.5.4 Heterogeneous Feature Representation

A robot's sensory features can be represented in different ways depending on the feature extraction method. To evaluate our framework with different feature representations used by the individual robots, we discretized the effort data into 15 temporal bins, where each bin consists of effort values' range computed by subtracting the minimum effort value from the maximum effort value in that bin. Fig. 6.7 shows the results of the speeding up object recognition and the novel object recognition tasks on this new representation, where *Baxter* and *Sawyer* serve as the source robots and *Fetch* serves as the target robot. Fig. 6.7A indicates that the transfer condition enables the target robot to perform better than the baseline condition especially with less experience with objects. Moreover, Fig. 6.7B suggests that the target robot learned to recognize novel objects with knowledge transferred by the source robots. These results are consistent with the results of the previous feature representation we presented, which means knowledge can be transferred using KEMA for different representations.

6.6 Summary

To enable robots to work in human-inhabited environment, they would need to recognize objects' properties through interaction. Non-visual sensory signals (e.g. haptic) collected by a robot's interaction cannot be used to train another robot as

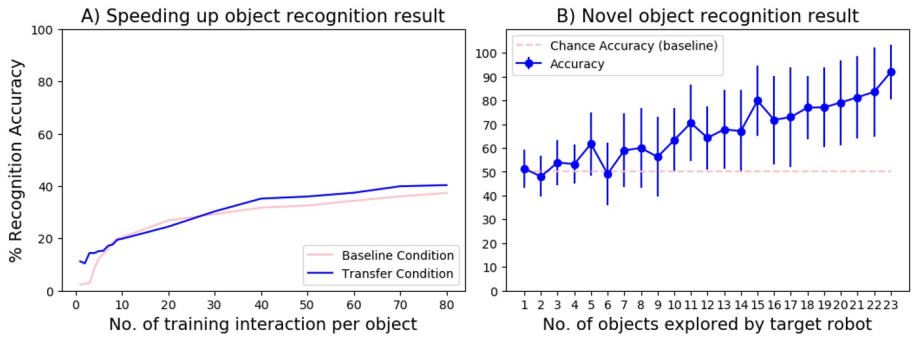


Figure 6.7: Results of a different feature representation, where *Baxter* and *Sawyer* serve as the source robots and *Fetch* serves as the target robot. (A) shows the results of the speeding up object recognition task, where predictions of all the behaviors are combined. (B) shows the accuracy curve of 2 novel objects recognition task.

the feature space of such data is different for robots with different embodiments. In addition, collecting interaction based sensory signals is a time consuming process. Thus, we propose using kernel manifold alignment, to align the feature spaces of different robots into a common feature space, and use it to train the robots. We showed that our approach can enable the target robot to not only speed up the learning process by learning with less interaction, but also perform better by using aligned features from other robots rather than learning just from its own features. Moreover, we showed that the target robot can learn to recognize novel objects by knowledge transferred by the source robots.

A limitation of our experiment is that the dataset we used contains simulated robots, thus in future work, we plan to test our proposed knowledge transfer method on real robots. A kernel function that is designed to specifically capture time series data such as haptics is also a promising avenue for future exploration. Moreover, we would adapt our knowledge transfer method to a larger variety of non-visual sensors other than effort such as audio, temperature, and vibration. Finally, in our experiments, we addressed the object recognition task. In future work, we plan to extend our method to handle sensory knowledge transfer for other tasks, such as object manipulation, and language grounding.

Chapter 7

A Framework for Sensorimotor Cross-Perception and Cross-Behavior Knowledge Transfer for Object Categorization*

7.1 Introduction

From an early stage in cognitive development, humans, as well as other species, use exploratory behaviors (e.g., shaking, lifting, pushing) to learn about the objects around them [Pow99]. Such behaviors produce visual, auditory, haptic, and tactile sensory feedback [SS08], which is fundamental for learning object properties and grounding the meaning of linguistic categories and descriptors that cannot be represented using static visual input alone [LC09]. For example, to detect whether a container is full or empty, a human may lift it; to perceive whether a ball is soft or hard, a human may squeeze it [Gib88]. In other words, the behavior acts as a medium to find the answer, in the form of a sensory signal, to a question about

*__This chapter is based on the following paper:__ Gyan Tatiya, Ramtin Hosseini, Michael Hughes, and Jivko Sinapov, "A Framework for Sensorimotor Cross-Perception and Cross-Behavior Knowledge Transfer for Object Categorization", *Frontiers in Robotics and AI*, 7:137, 2020. [THHS20]

object properties.

Recent research in robotics has demonstrated that robots can also use multisensory feedback from interaction with objects (e.g., vision, proprioceptive, haptic, auditory, and/or tactile) to perform several tasks, including language grounding [TSS+16], object recognition [SBS+11], and object category acquisition [ANN+12]. One of the challenges in interactive multisensory object perception is that there is no general purpose multisensory knowledge representations for non-visual features such as haptic, proprioceptive, auditory, and tactile perceptions, as different robots have different embodiments, sensors, and exploratory behaviors. Because each robot has a unique embodiment and sensor suite, it is not easy to transfer knowledge of non-visual object properties from one robot to another. In existing work, each robot must learn its task-specific multisensory object models from scratch. Even if there are two physically identical robots, it is still not easy to transfer multisensory object knowledge as the two robots' exploratory behaviors may be implemented differently. Furthermore, sensors may fail over the course of operation and thus, an object classifier that relies on the failed sensor's input would become unusable until the sensor is fixed.

To address these limitations, this chapter proposes a framework for sensoirmotor knowledge transfer across different behaviors and different sensory modalities. The framework is designed to allow a robot to recover a failed sensor's input given sensor data from one or more of the robot's other sensory modalities. The framework also affords transfer from one robot to another across behaviors such that a source robot can transfer knowledge obtained during object exploration to a target robot that may have different actions and sensory modalities. This means that if the source robot and the target robot had observations of what the same objects feel like when lifted and pressed, the pair of observations could be used to learn a function that maps observations from the source robot's feature space to that of the target robot. Such generated observations (i.e., features) can be used to train task-specific recognition models for the target robot to identify novel objects that only the source robot has interacted with. The advantage of this method is that

the target robot does not need to learn the perceptual recognition task from scratch as it can use the generated observations obtained from the source robot. Similarly, knowledge can be mapped from one sensory modality to another, such that if a sensor fails, modules that require its input can still operate, or if a new sensor is added, the robot would not have to exhaustively explore all objects in its domain from scratch to learn models that use the new sensor's output.

We evaluated the proposed framework on a publicly available dataset in which a robot explored 100 objects, corresponding to 20 categories using 9 exploratory behaviors coupled with auditory, haptic, vibrotactile and visual data. We consider the object category recognition task in which the robot has to recognize the category of a novel object given labeled examples on a training set of objects. The task is closely related to grounded language learning and other applications where a robot may need to identify object properties that cannot be inferred based on static visual input alone. We evaluate two different approaches for knowledge transfer, 1) variational encoder-decoder networks, which allows one or more source feature spaces to be mapped into a target feature space; and 2) variational auto-encoder networks, which are trained to reconstruct their input features and can be used to recover features from a missing sensor or new behavior-modality combination. The results show that both approaches are able to effectively map data from one or more sensory modalities to another, such that a target robot with a different morphology or a different set of sensors can achieve recognition accuracy using the mapped features almost as good as if it had learned though actual interaction with the objects.

In the context of the broader dissertation, this chapter expands the preliminary framework proposed in Chapter 5 and introduces a β-Variational Autoencoder Network (β-VAE) architecture, which is a method for *Transfer using Projection to Target Feature Space*. This architecture supports multiple source robots that performed different behaviors to generate target robot's features. In the β-VAE architecture, the basic idea is to learn a non-linear probabilistic mapping to construct the target robot features from the input source features and use β to control the

information capacity of the latent representation. β can be set to any nonnegative value. A high β value would impose high constraint, and the latent representation would have low capacity. This chapter addresses two knowledge transfer scenarios: cross-behavior and cross-perception, with applications in object category and object identity recognition, as compared to only cross-behavior and object category recognition in Chapter 5. Additionally, this chapter used the Kernel Manifold Alignment (KEMA)-based method for *Transfer using Projection to Shared Latent Feature Space* proposed in Chapter 6 as a baseline for comparison.

7.2 Related Work

7.2.1 Object Exploration in Cognitive Science

Previous cognitive science studies show that it is fundamental for humans to interactively explore objects in order to learn their auditory, haptic, proprioceptive and tactile properties [Gib88, Pow99, CSS+04]. For example, in [SLM00] the effect of perception was put into a test by presenting kids with a sponge painted to adapt the visual characteristics of a rock. The kids perceived the sponge as a rock until they came in contact with it by touch, at which point they recognized that it was not a rock, but rather, a sponge. The case illustrates an example of how haptic and tactile data can supplement visual perception in inferring objects' characteristics [Hel92]. Studies have also demonstrated that infants commonly use tactile exploratory behaviors when exploring a novel object [Ruf84]. For example, [ST99] found that 7-month-old infants can perform tactile surface recognition using tactile exploratory strategies in the absence of visual information. In early stages of development, object exploration is less goal driven and serves the primary purpose of learning how objects feel, sound, and move; as we get older, we apply this learned knowledge by performing specific exploratory behaviors to identify the properties of interest, e.g., lift an object to perceive its weight, touch it to perceive its temperature, etc. [Gib88, ST99].

Studies have also shown that humans are capable of integrating multiple

sensory modalities to detect objects and each modality contribute towards the final decision [EB04]. [WWCM07] have reported that combining multiple sensory signals such as visual and tactile with exploratory behaviors on objects produces more accurate object representation than using only a single sensory signal. Moreover, several lines of research in psychology have shown that object exploration, when performed in a natural setting, is a multimodal process. For example, consider a simple action of touching an object. In Chapter 4 of "Tactual Perception: A Sourcebook", Lederman writes:

"*Perceiving the texture of a surface by touch is a multimodal task in which information from several different sensory channels is available. In addition to cutaneous and thermal input, kinesthetic, auditory, and visual cues may be used when texture is perceived by touching a surface. Texture perception by touch, therefore, offers an excellent opportunity to study both the integrated and the independent actions of sensory systems. Furthermore, it can be used to investigate many other traditional perceptual functions, such as lateralization, sensory dominance, and integration masking, figural aftereffects, and pattern recognition. [SF82]*"

[LC09] have demonstrated that humans rely on multiple sensory modalities to learn and detect many object properties (e.g., roughness, hardness, slippery, and smooth). In their studies, that over half of the most common adjectives and nouns have a strong non-visual component in terms of how humans represent each word. Inspired by these findings, this chapter proposes a knowledge transfer framework so that the robots in our factories and workplaces can appropriately learn from and reason about multimodal sensory information produced during physical interaction with objects.

7.2.2 Multisensory Object Perception in Robotics

Vision-based recognition of an object is the commonly adopted approach; however, several research studies show incorporating a variety of sensory modalities is the key to further enhance the robotic capabilities in recognizing multisensory object properties (see [BHS+17] and [LKS+20] for a review). Previous work has shown that

robots can recognize objects using non-visual sensory modalities such as the auditory [TJNF05, SWS09, LZAL17, EKSW18, JLWS19, GGP20], the tactile [SSSS11, FL12, BRK12b, KSG+19] and the haptic sensory modalities [NMS04, BSO+09, BGS+20]. In addition to recognizing objects, multisensory feedback has also proven useful for learning object categories [SSS+14a, HBMK16, TYT18, TS19], material properties [ECK17, ERCM18, ELCK19], object relations [SSS14c, SKSS16], and more generally, grounding linguistic descriptors (e.g., nouns and adjectives) that humans use to describe objects [TSS+16, RK19, APR+20].

A major limitation of these methodologies is that they need large amounts of object exploration data, which may be prohibitively expensive to collect. In other words, the robot must perform a potentially large number of behaviors on a large number of objects, multiple times, to collect enough data to learn accurate models. To address this, some work has focused on learning to optimize the exploratory behavior as to minimize the number of explorations needed to identify the object [FL12]. Other research has proposed learning object exploration policies when attempting to identify whether a set of categories apply to an object [AWZ+18]. In addition, methods have also been proposed to select which behaviors to be performed when learning a model for a given category based on its semantic relationship to the categories that are already known [TSMS18].

In spite of all of these advances in robotics, a major outstanding challenge is that multisensory information, as perceived by one robot, is not directly useful to another robot that has a different body, different behaviors and possibly different sensory modalities. In other words, if a robot learns a classifier for the word "soft" based on haptic input produced when pressing an object, that classifier cannot directly be deployed on another robot that may have a different body, different number or type of haptic sensors, or a different encoding of the behavior. Furthermore, existing methodologies rarely try to learn the relationships between different sensory modalities in a way that can handle sensor failure. This chapter addresses these limitations by expanding a preliminary framework [THCHS19] as to afford sensorimotor knowledge transfer between multiple sensory modalities and exploratory

behaviors.

7.2.3 Domain Adaptation

Most machine learning models assume that both training and test data are drawn from the same distribution and are in the same feature space. However, in many cases, the training and the test distributions could be different, making it crucial to adapt the examples from different distributions. The process of adapting one or more source domains to transfer knowledge for the goal of improving the performance of a target learner is called domain adaptation [BDBC$^+$10, MMR09]. In domain adaptation, the training examples are obtained from the source domain with labels, while the test examples are obtained from the target domain with no labels or only a few labels. In these settings, while the source and target domains are different, they are in a semantically similar feature space. Our goal is to train a model for the target robot using one or more semantically similar source robot feature spaces.

Encoder-decoder networks have recently shown promising results in facilitating domain adaptation [MKK$^+$18, GFL19]. Encoder-decoder networks are composed of two feed-forward neural networks: an *encoder* and a *decoder* [HZ93, HS06]. The encoder maps an input feature vector (the source robot sensory input) into a fixed-length code vector. Give a code vector as input, the decoder produces a target feature vector as output, such that it minimizes the reconstruction loss between the produced output and a ground truth observation. Frequently, such architectures are used for dimensionality reduction, i.e., the intermediate code vector size is much smaller than the size of either input or output. If the input and output data points are identical, they are referred to as "autoencoder" networks [LWL$^+$17]. Autoencoders have been successfully applied to vision domains, such as image reconstruction [MM17] and image super-resolution [ZYW$^+$15]. The term "encoder-decoder" applies when the input and output are different. Encoder-decoder approaches have been shown successful in applications such as language translation, in which the input language is different than the output language [SVL14], as well as in extracting multi-scale features for image representation tasks [KSB$^+$10]. As tactile signals

can complement visual information, both modalities have been used to learn shared features for texture recognition problems [LYA+18], and encoder-decoder networks have been proposed for predicting visual data from touch (and vice versa) [LBL19]. We hypothesize that encoder-decoder networks can be used to generate the sensory features that would be produced by one robot (the target robot) when it interacts with an object given features produced by another robot (the source robot) that has already explored the object. This mapping would enable multisensory object knowledge learned by the source robot to be transferred to the target robot, which would reduce the need for exhaustive object exploration necessary for producing multisensory observations of objects.

7.3 Learning Methodology

7.3.1 Notation and Problem Formulation

Consider the case where two or more robots are tasked with recognizing object properties using sensory data produced when performing a behavior on an object. For a given robot r, let $\mathcal{B}_r$ be its set of exploratory behaviors (e.g., grasp, lift, press, etc.). Let $\mathcal{M}_r$ be its set of sensory modalities (e.g., audio, tactile, vision, etc.) and let $\mathcal{C}_r$ be the set of sensorimotor contexts where each context denotes a combination of a behavior and modality (e.g., *grasp-tactile*, *lift-haptic*).

Let $\mathcal{O}$ denote the set of objects in the domain and let $\mathcal{Y}$ denote the discrete set of categories such that each object maps to particular category $y \in \mathcal{Y}$. When performing an action on an object $o \in \mathcal{O}$, the robot records sensory features for all contexts associated with the behavior, i.e., during the i^{th} exploration trial, the robot observes features from context $c \in \mathcal{C}_r$ represented as $x_i^c \in \mathbb{R}^{n_c}$ where n_c is the dimensionality of the features space associated with context c. For a given context $c \in \mathcal{C}_r$, let $\mathcal{X}_c$ be the n_c-dimensional feature space associated with that context. For the category recognition problem, the robot needs to learn a classifier decision function $d_c : \mathcal{X}_c \rightarrow \mathcal{Y}$ that maps the sensory feature vector to one of the discrete set of categories $y \in \mathcal{Y}$. In our framework the robot learns a classifier d_c for each

sensorimotor context c using supervised learning with labeled examples.

Consider the case where one robot, the *source* robot, has explored all objects in $\mathcal{O}$ multiple times such that it can learn accurate classifiers for the category recognition task. Another robot, the *target* robot, however, has only explored a subset of the objects from categories $\mathcal{Y}_{shared} \subset \mathcal{Y}$ and needs to learn a category recognition model for a different set of categories $\mathcal{Y}_{target} \subset \mathcal{Y}$ where $\mathcal{Y}_{shared} \bigcap \mathcal{Y}_{target} = \emptyset$. In other words, the target robot must learn to categorize objects according to the labels $\mathcal{Y}_{target}$ without having interacted with any objects from those categories. Below, we describe our knowledge transfer model that enables the target robot to solve this task.

7.3.2 Knowledge Transfer Model

To transfer sensory object representations learned by one robot to another, we need a function that predicts what the target robot would observe in a particular feature space when interacting with an object, given what the source robot has observed with that object in one of its own feature spaces. More specifically, let $c_s \in \mathcal{C}_s$ and $c_t \in \mathcal{C}_t$ be two sensorimotor contexts, one from the source robot s and the other from the target robot t. Thus, the task is to learn a function $map_{c_s,c_t} : \mathcal{X}_{c_s} \rightarrow \mathcal{X}_{c_t}$ which takes as input an observed feature vector $x_i^{c_s}$ from the source context and produces $\hat{x}_i^{c_t}$, the estimated sensorimotor features in context c_t that the target robot would have observed if it interacted with the object that produced sensorimotor features $x_i^{c_s}$ for the source robot. We considered two knowledge transfer scenarios:

Cross-perception transfer: A knowledge transfer model that maps the feature spaces across different modalities of the robot performing the same behavior is referred as *cross-perception transfer*. This transfer can be useful in a scenario where one of the robot's sensors fails and its signal is recovered from the available set of sensors. Another application is the situation where a new sensor is added to the robot at a time after the robot has explored an initial set of objects for a recognition problem.

Cross-behavior transfer: A knowledge transfer model that maps the feature spaces across different exploratory behaviors performed by the robot is referred as *cross-behavior transfer*. This transfer can be useful in a scenario where a new robot with less experience with objects is required to learn from a more experienced robot that has thoroughly explored the objects in the recognition domain. Note that such a mapping can also be cross-perceptual as not only the behaviors, but the sensors as well, may be different across the source and the target robots.

We can further extend this model to take input from multiple contexts (e.g., tactile and visual data) and output a reconstruction for some other context (e.g., haptic data). Further, we also consider mappings which take inputs from a fixed set of sensorimotor contexts and simply reconstruct the observations in the same feature spaces. We refer to mappings whose input and output contexts are identical as *autoencoders*. Mappings for which the output contexts are distinct from the input ones are referred to as *encoder-decoders*. We propose that such mappings can be learned via two probabilistic approaches, the β-variational encoder-decoder (β-VED) and β-variational autoencoder (β-VAE), which we describe below. While the core ideas behind the VAE [KW13] and its extension to the β-VAE [HMP$^+$17] have been widely-used across machine learning, we specialize them to encoder-decoder architectures to solve transfer learning problems across robot contexts.

7.3.2.1 β-Variational Encoder-Decoder Network

Our proposed β-VED approach (shown in Fig. 7.1) is designed to transfer knowledge from the source robot to the target robot. This β-VED learns a non-linear probabilistic mapping to construct the target robot features $x_i^{c_t}$ from the input source features $x_i^{c_s}$ while compressing the data in the process to discover an efficient representation in a "learned" latent code space. We denote the lower-dimensional, fixed-size encoding of the data for example i by the code vector $z_i \in \mathbb{R}^{D_z}$ of size D_z.

The β-VED is defined by two related probabilistic models, fully described below. The first model is fully generative, producing latent codes and target features. The second model is conditional, producing latent codes giving source features.

These are trained together, related by the fact that the second model should be an accurate approximation of the posterior over latent codes given target data for the first model. We will describe how to coherently fit the model to observed data using the same well-motivated training objective as [HMP+17], but specialized to our robot context.

First, the generative model defines a joint distribution over latent codes and target features:

$$p(z_i) = \text{MultivariateNormal}(z_i|0, I_{D_z}) \tag{7.1}$$

$$p_\theta(x_i^{c_t}|z_i) = \text{MultivariateNormal}(x_i^{c_t}|\text{decode}(z_i, \theta), \sigma^2 \cdot I_{n_{c_t}}) \tag{7.2}$$

Here, the standard Normal prior distribution on code vectors $p(z_i)$ is designed to encourage mild independence among its entries, while the likelihood $p_\theta(x_i^{c_t}|z_i)$ is designed so its mean is the output of a flexible "decoder" neural network with weight parameters θ. Given each distinct latent code, the decoder will map to a distinct mean in target feature space.

Second, the conditional model of our proposed β-VED defines a probability distribution $q_\phi(z_i|x_i^{c_s})$, which allows probabilistic mapping from the source features to a latent code vector:

$$q_\phi(z_i|x_i^{c_s}) = \text{MultivariateNormal}(\text{encode}(x_i^{c_s}, \phi), \hat{\sigma}^2 \cdot I_{D_z}) \tag{7.3}$$

Again, we use a flexible "encoder" neural network with weight parameters ϕ to define a non-linear mapping from any source features to a mean vector in latent code space. A specific code vector is then drawn from a Normal distribution with that mean and a diagonal covariance with learned scale. For both encoder and decoder neural networks, we use multi-layer perceptron architectures with non-linear activation functions.

Training the β-VED for a context pair c_s, c_t amounts to learning the weight parameters of the two neural networks, θ and ϕ, as well as the variance parameters

σ^2 and $\hat{\sigma}^2$. Henceforth, we will use notation θ and ϕ to represent *all* parameters we need to learn (both the weights and the variances), for compact notation.

Our training problem requires observing features from both source and target robot across a set of N total objects where both robots interact with each object M times. The objects used to train the β-VED come from the set of shared categories $\mathcal{Y}_{\text{shared}}$. Given a dataset of source-target feature pairs $\{x_i^{c_s}, x_i^{c_t}\}_{i=1}^{N\times M}$, where each pair comes from the same object, we find the parameters $(\theta,\ \phi)$ that *maximize* the following evidence lower bound variational objective function:

$$\mathcal{L}(\theta, \phi; x^{c_s}, x^{c_t}, z, \beta) = \sum_{i=1}^{N\times M} \mathbb{E}_{q_\phi(z_i|x_i^{c_s})}[\log p_\theta({x_i}^{c_t}|z_i)] - \beta D_{KL}(q_\phi(z_i|x_i^{c_s})||p(z_i)) \tag{7.4}$$

This objective, which comes from the β-VAE work by [HMP+17], is based on well-known lower bounds on marginal likelihood used to motivate variational inference in general [BKM17]. We can interpret the two terms here in justifiable ways. The first term seeks to maximize the likelihood that the real observed target features $x_i^{c_t}$ are similar to the model's "reconstructed" target features $\hat{x}_i^{c_t}$. Recall that reconstruction occurs in two steps: first sampling a code vector from the conditional model ("encoder"), then sampling the target features from the generative likelihood ("decoder") given that code vector. The second term in Eq. (7.4) is a Kullback-Leibler (KL) divergence used to quantify the distance between our learned conditional distribution q over latent code vectors z_i given source features $x_i^{c_s}$, and the prior distribution over codes, denoted $p(z_i)$. The KL-divergence acts as a regularizer on the learned code space, encouraging the approximate posterior distribution to be close to the prior distribution, which is a Normal with mean zero and identity covariance.

The coefficient $\beta > 0$ was introduced to the objective by [HMP+17] to control the model's emphasis on the information capacity of the latent code space. Large $\beta > 1$ lead to low capacity (but highly interpretable representations), while low $\beta < 1$ value demphasizes the KL divergence and allows higher fidelity reconstructions (at the expense of the interpretability of the latent space). Note that $\beta = 1$ with

target and source domains the same recovers the standard variational inference objective used to train VAEs by [KW13]. For implementation details, readers can refer section 7.4.2.

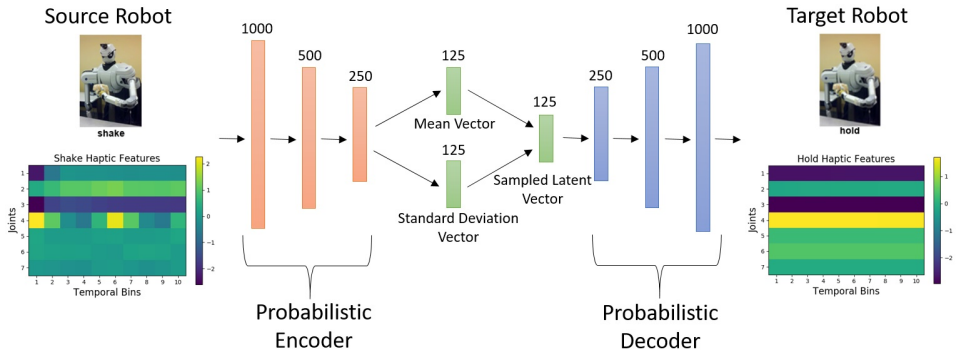


Figure 7.1: The proposed β-VED network architecture. In this example, an input data point from the *shake-haptic* context is projected to the *hold-haptic* context.

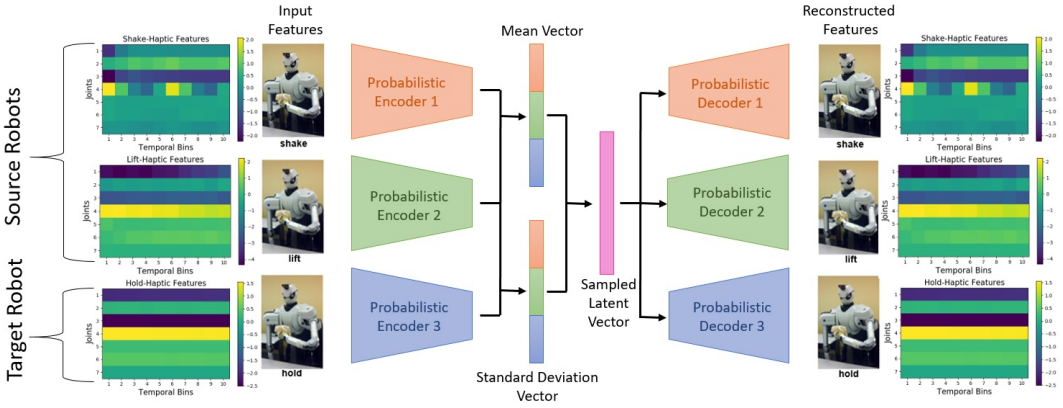


Figure 7.2: The proposed β-VAE network architecture. In this example, the network is trained to reconstruct data points from the *hold-haptic* context given data points from the *shake-haptic* and *lift-haptic* contexts.

7.3.2.2 β-Variational Autoencoder Network

The major difference between β-VED and β-VAE is that in β-VED the input is different than the output, and in β-VAE the input is same as the output. Because our goal is to generate the target robot's features using the source robot's features, we used both source and target robot's data as the input as well as the output for β-VAE. The benefit of using β-VAE over β-VED is that we can have more than one source robot projecting into the target robot's feature space rather than just one

source robot.

Our proposed β-VAE is shown in Fig. 7.2. First, the features of each robot go through their private encoder and project into a common latent distribution between all the robots. Then a code is sampled from the latent distribution, and passed through the private decoder for each robot. The latent distribution is learned to reflect the categorical information of the input, and the private encoder and decoder is learned to compress and generate robot specific features. The objective function of β-VAE is same as for the β-VED discussed in 7.3.2.1.

7.3.3 Using Transferred Features for Category Recognition

Once we have a trained knowledge transfer model (e.g. β-VED, β-VAE) for one or more source context c_s (e.g. *push-haptic* or *drop-audio*), we can then train the target robot to recognize novel object categories it has never experienced before, as long as examples of these categories are experienced by the source robot under context c_s. We refer to this novel set of categories as $\mathcal{Y}_{\text{target}}$. We assume that the source robot has experienced a total of J feature-label pairs from these categories: $\{x_j^{c_s}, y_j\}_{j=1}^{J}$, where $y_j \in \mathcal{Y}_{\text{target}}$. We project this labeled dataset to the target robot by producing a "reconstructed" training set: $\{\hat{x}_j^{c_t}, y_j\}_{j=1}^{J}$, which is then used for supervised training of a multi-class classifier appropriate for the target context. We produce reconstructed features by sampling from our pre-trained probabilistic knowledge transfer models. This involves two steps of sampling: a sample from the encoder followed by a sample from the decoder. The resulting reconstructed target feature vector (and its associated known label) can then be used to train a classifier. In the experiments below, we generally found that a single sample of the target feature vector worked reasonably well in terms of downstream classification performance, so we use that throughout. Future work could explore how multiple samples might improve robustness.

Subsequently, at test time when the target robot interacts with novel objects without category labels, the target robot observes features x^{c_t} and feeds these features directly to its pre-trained classifier to predict which category within the

set $\mathcal{Y}_{\text{target}}$ it has observed. While we assume that at test time, the target robot encounters objects only from categories $\mathcal{Y}_{\text{target}}$, it is straightforward to extend our approach for the combined set of possible categories $\mathcal{Y}_{\text{target}}$ and $\mathcal{Y}_{\text{shared}}$ by combining the target robot's both real and reconstructed training sets.

7.4 Experiments and Results

7.4.1 Dataset Description

We used the publicly available dataset introduced by [SSS+14a], in which an upper-torso humanoid robot used a 7-DOF arm to explore 100 different objects belonging to 20 different categories using 9 behaviors: *press*, *grasp*, *hold*, *lift*, *drop*, *poke*, *push*, *shake* and *tap* (shown in Fig. 7.3). During each behavior the robot recorded audio, haptic, vibrotactile and visual feedback using four sensors: 1) an Audio-Technica U853AW cardioid microphone that captures audio sampled at 44.1 KHz; 2) joint-torque sensors that capture torques from all 7 joints at 500 Hz, 3) vibrotactile sensor consisting an ABS plastic artificial fingernail with an attached ADXL345 3-axis digital accelerometer, and 4) a Logitech webcam that captures 320 x 240 RGB images. Thus, there are 36 sensorimotor contexts, i.e., each combination of a behavior and sensory modality serves as a context. The robot performed each behavior 5 times on each of the 100 objects, thus there were 4,500 interactions (9 behaviors x 5 trials x 100 objects). We used the auditory, haptic, and visual features as described by [SSS+14a]. The parameters regarding the feature extraction routines (e.g., the number of frequency bins) were left identical to those in the original dataset as to be consistent with other papers that use the same dataset. Next, we briefly discuss the feature extraction methodology used by [SSS+14a] to compute features from the raw sensory signal.

For audio, first, the spectrogram was computed by Discrete Fourier Transformation using 129 log-spaced frequency bins. Then, a spectro-temporal histogram was produced by discretizing both time and frequencies into 10 equally spaced bins, thus producing a 100-dimensional feature vector. An example spectrogram of a de-

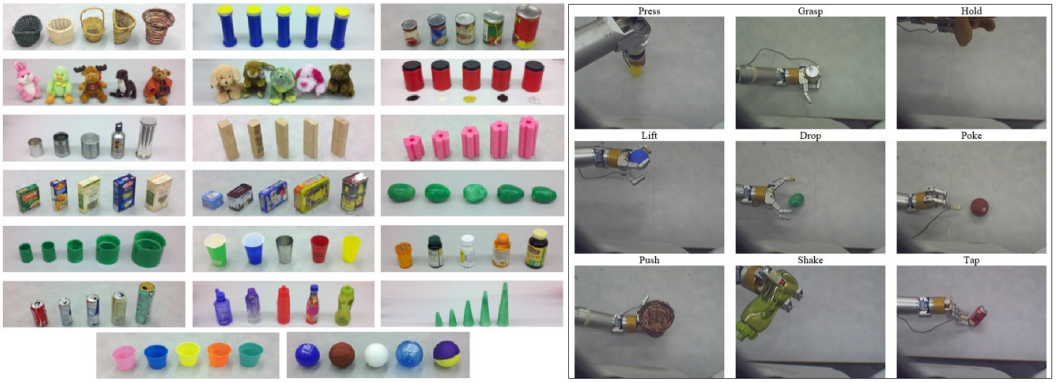


Figure 7.3: Left: 100 objects, grouped in 20 object categories. Right: The interactive behaviors that the robot performed on the objects. From top to bottom and from left to right: (1) *press*, (2) *grasp*, (3) *hold*, (4) *lift*, (5) *drop*, (6) *poke*, (7) *push*, (8) *shake* and (9) *tap*.

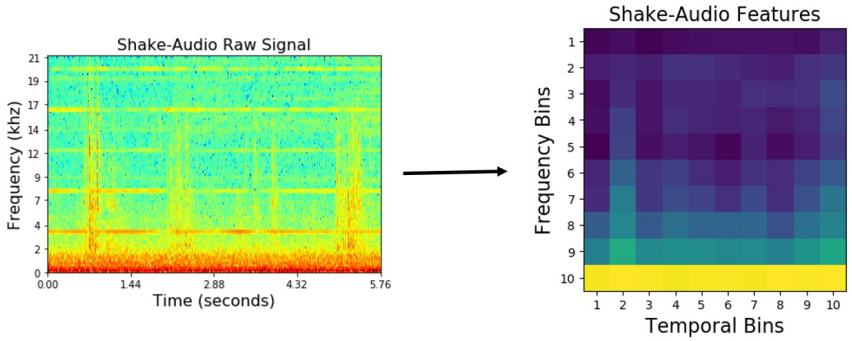


Figure 7.4: *Audio* features using *shake* behavior performed on an object from the *medicine bottles* category.

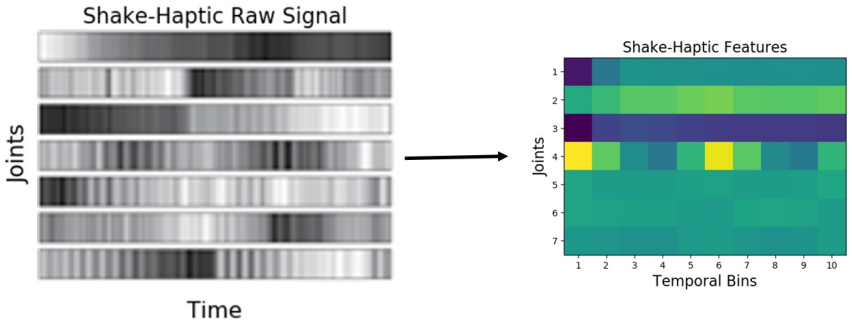


Figure 7.5: *Haptic* features produced when the robot performed the *shake* behavior on an object from the *medicine bottles* category.

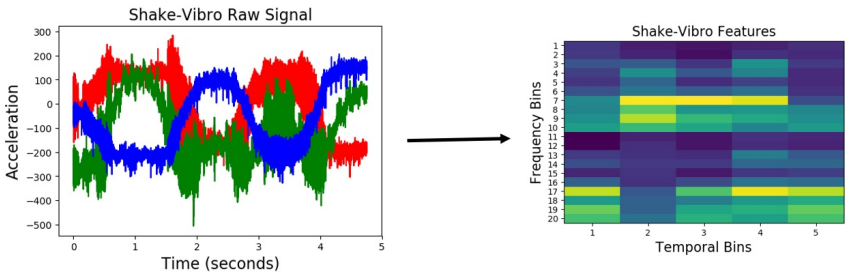


Figure 7.6: *Vibrotactile* features produced when the robot performed the *shake* behavior on an object from the *medicine bottles* category.

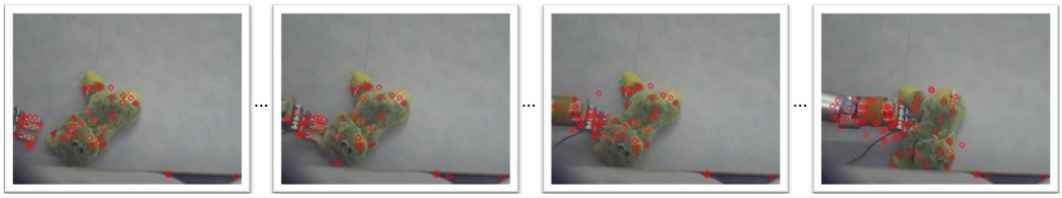


Figure 7.7: *Visual (SURF)* features detected when the *tap* behavior was performed on an object from the *large stuffed animals* category. The feature descriptors of the detected interest points over the entire interaction were represented using bag-of-words.

tected sound, and the resulting low-dimensional feature representation are shown in Fig. 7.4.

Similar to audio, haptic data was discretized into 10 equally spaced temporal bins, resulting in a 70-dimensional feature vector (the arm had 7 joints). Fig. 7.5 shows an example raw joint-torque data and the resulting feature representation. Vibrotactile features were computed from the raw data using frequency-domain analysis as described by [SSSS11]. The 3-axis accelerometer time series were converted into a univariate magnitude deviation series, on which the Discrete Fourier Transform was performed, resulting in a spectrogram with 129 frequency bins denoting intensities of different frequencies over time. This spectrogram was discretized into 5 temporal bins, and 20 frequency bins (an example representation is shown in Fig. 7.6).

The robot also recorded the raw RGB images from its camera as it performed a behavior on an object. For each interaction, the Speeded-Up Robust Features (SURF) features were computed on each image (a sample set of SURF features detected over an image are shown in 7.7). SURF consisted of 128-dimensional

feature vector representing the distribution of the first order Haar wavelet responses within the interest point neighborhood.

7.4.2 Knowledge Transfer Model Implementation

The β-VED network consisted of a multi-layer perceptron (MLP) architecture with three hidden layers for both the encoder and the decoder, with 1000, 500, 250 hidden units respectively, Exponential Linear Units (ELU) [CUH16] as an activation function, and a 125-dimensional latent code vector as shown in Fig. 7.1. The latent layer and the output layer used a linear activation function. The network parameters are initialized using Glorot uniform initializer [GB10] and updated for 1000 training epochs using the Adam optimizer [KB15] with a learning rate of 10^{-4}, implemented using TensorFlow 1.12 [ABC+16]. The prior distribution of the latent representation used a normal distribution with a mean of zero and a standard deviation set to one. The β value was set to 10^{-4}. We performed network hyper-parameter tuning by trying different numbers of layers in the network within the range of 1 to 5 and different numbers of units in each layer within the range of 100 to 1000. Then, we choose the minimum number of layers and units after which increasing them did not improve the performance. We performed this network hyper-parameter tuning experiments on 10 randomly selected projections (e.g. *shake-haptic* to *hold-haptic*, *poke-vision* to *poke-haptic*) and then used the selected hyper-parameters for the entire set of projections. Note that the hyper-parameters and the network architecture we used may not be optimal for a different dataset that may have a much larger input dimensionality or a larger set of datapoints.

For β-VAE (shown in Fig. 7.2), we used the same network architecture for all the private encoders and decoders as of β-VED discussed above. The output of all the encoders were concatenated and connected to the mean and the standard deviation vector. The sampled latent vector was used as an input to the decoders. The rest of the implementation details and the hyper-parameters of β-VAE are same as of β-VED.

7.4.3 Category Recognition Model Implementation

At test time, we used multi-class Support Vector Machine (SVM) [Bur98] to classify objects into the categories from the set $\mathcal{Y}_{\text{target}}$. SVM uses the kernel trick to map the training examples to a high-dimensional feature space where the data points from different classes may be linearly separable. We used the SVM implementation in the open-source scikit-learn package [PVG+11], with the Radial Basis Function (RBF) kernel and default hyperparameters.

7.4.4 Evaluation

We consider the setting where the source robot interacts with all 20 object categories, while the target robot interacts with 15 randomly selected object categories. The objects of the shared 15 categories experienced by both robots are used to train the knowledge transfer model that projects the sensory signals of the source robot to that of the target robot. Subsequently, the trained knowledge transfer model is used to generate "reconstructed" sensory signals of the other 5 object categories in $\mathcal{Y}_{\text{target}}$ that the target robot never interacted with. Each sensory signal experienced by the source robot from objects in these categories is thus "transferred" to a target feature vector. Since the dataset we used has only one robot, we evaluated our framework in two scenarios: *cross-perception* knowledge transfer, in which one of the robot's sensors fail and its signal is recovered from the set of available sensors, and *cross-behavior* knowledge transfer, in which the source and the target robots are physically identical, but they perform different behaviors on shared objects. *

We consider three possible category recognition training cases: (1) our proposed transfer-learning framework using the generated data from the source context (i.e., how well the target robot performs if it uses transferred knowledge from the source robot), (2) a domain adaption method, KEMA (kernel manifold alignment) [TCV16, TSES20] that aligns two different robots' feature spaces into a common

*Note that the proposed transfer learning methodology does not make this assumption and is applicable in situations where the two robots are morphologically different and/or use different sensors and feature representations for a given modality.

space and then trains the target robot using the aligned features, and (3) a non-transfer baseline using the target robot's ground truth features produced by actual interaction (i.e., the best the target robot could perform if it had experienced all the objects itself during the training phase). In all three cases, ground truth features detected by the target robot are used as inputs to the category recognition model when testing. We used 5-fold object-based cross-validation, where each training fold consisted of 4 objects from each of the 5 object categories in $\mathcal{Y}_{\text{target}}$ that the target robot never interacted with, while the test fold consisted of the remaining objects. Since the robot interacted with each object for a total of 5 times, there were 100 (5 categories x 4 objects x 5 trials) data points in the training set, and 25 (5 categories x 1 objects x 5 trials) data points in the test set. This process was repeated 5 times, such that each object occurred 4 times in the training set and once in the test set.

The performance of the target robot at recognizing novel categories of objects it never explored was evaluated using two metrics. The first, accuracy, is defined as:

$$\%\ Recognition\ Accuracy = \frac{Correct\ predictions}{Total\ predictions}.$$

The process of selecting the 15 random categories to train the knowledge transfer model, generating the features of the remaining 5 categories, training the two classifiers using generated and ground truth features, and calculating accuracy for both classifiers on ground truth observations by 5-fold object-based cross validation was repeated for a total of 10 times to produce an accuracy estimate.

The second metric that we used was accuracy delta (%), which measures the loss in recognition accuracy as a result of using the generated features for training when compared to using the ground truth features. We define this loss as:

$$Accuracy\ Delta = Accuracy_{truth} - Accuracy_{generated}$$

where $Accuracy_{truth}$ and $Accuracy_{projected}$ are the accuracies obtained when using ground truth and generated features, respectively. Smaller accuracy delta suggests

that the features generated by the learned mapping are similar to the target robot's real features, and that the target robot can use these generated features to learn a classifier that achieves comparable performance as if the target robot learned by actually exploring the objects.

7.4.5 Results

7.4.5.1 Cross-Perception Sensorimotor Transfer

First, we consider the case where a robot is tasked with learning a mapping from one of its sensory modalities (e.g., *vision*) to another (e.g., *haptic*) for the same behavior. Such a mapping would be needed if the modality sensor associated with the target context c_t fails at test time, or if a new sensor is added such that there is limited data produced with objects with that sensor.

Illustrative Example Consider the case where the robot performs *poke* behavior while the *haptic* sensor is not working. Projecting *haptic* features from *vision*, enables the robot to achieve 42.5% recognition accuracy using β-VED and 35.6% using β-VAE, compared with 49.6% when using features from real interactions (shown in Fig. 7.10). In other words, the robot's category recognition model trained on the reconstructed signal of a failed sensor performs very close to the model that been trained on real signal. Chance recognition accuracy for 5 categories is 20% and the accuracies of individual sensorimotor contexts are typically in the 40-60% range. Note that the overall recognition accuracy can be boosted to nearly 100% by using multiple behaviors and sensory modalities [SS10] but this is out of scope for this chapter.

To visualize how the projected features look as compared to the ground truth features, we plotted an example of *tap-vibro* to *tap-haptic* projection using β-VED. Fig. 7.8 shows a feature vector from the source feature space, the projected observation in the target features, and a ground truth feature vector captured by performing the *tap* behavior on the same object. The projected and the ground truth features are very similar. Note that this is a special case and there are certainly

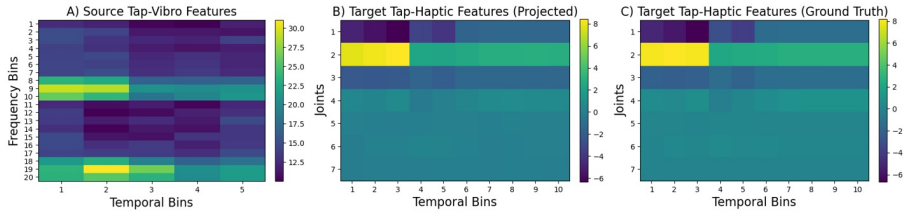


Figure 7.8: Visualizations of: (A) the source robot's features; (B) the target robot's projected features using β-VED, and (C) the corresponding ground truth features captures by performing *tap* behavior on an object from the *bottles* category.

pairs of source-target contexts which do not produce accurate projections.

Now, consider a case where the robot performs *push* behavior while the *haptic* sensor is not working. Generating *haptic* features using *audio* and *vision* by β-VAE as two sources, enables the robot to achieve 38.6% recognition accuracy. This is a significant boost in accuracy as projecting *vision* alone achieves 27.8%, and projecting *audio* alone achieves 23.9%.

To find the effect of the amount of data used to train a two sources β-VAE and corresponding two single source β-VEDs on the recognition performance, we varied the number of shared object categories for a projection. Fig. 7.9 shows the recognition performance for different number of number of shared categories for β-VAE *push-audio* and *push-vision* to *push-haptic* projection, β-VED *push-vision* to *push-haptic* projection and β-VED *push-audio* to *push-haptic* projection. As demonstrated combining *vision* and *audio* features improves the generation of *haptic* features for most number of shared categories, and the performance of two sources β-VAE reaches very close to the ground truth features accuracy.

Accuracy Results of Category Recognition Since there are 4 modalities (*audio*, *haptic*, *vibro* and *vision*), if a sensor fails, there are 3 possible mappings that take a single sensory modality as input, each from an available sensor to a failed sensor, so there are $4 \times 3 = 12$ possible mappings (e.g. if the *haptic* sensor fails, the 3 possible mapping would be *audio* to *haptic*, *vibro* to *haptic*, *vision* to *haptic*). There are 9 behaviors, so there are $12 \times 9 = 108$ projections (e.g. *poke-vision* to *poke-haptic*, *tap-vision* to *vision-haptic*). Fig. 7.10 shows the 5 β-VED cross-perception

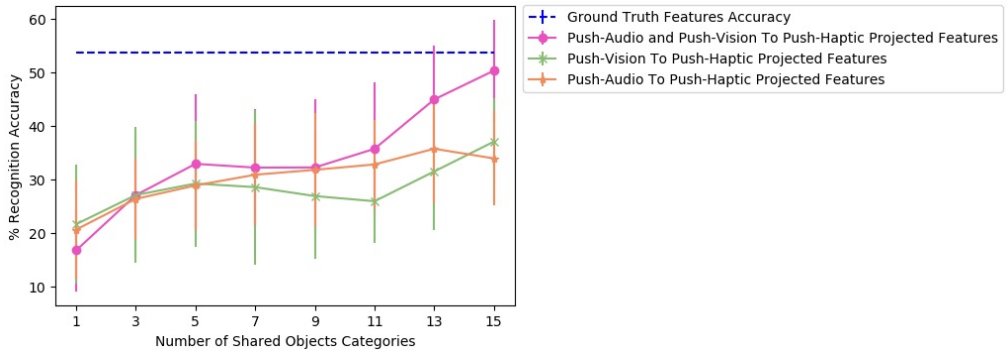


Figure 7.9: Accuracy achieved by the projected features of the robot for different number of shared objects classifier for β-VAE *push-audio* and *push-vision* to *push-haptic* projection, β-VED *push-vision* to *push-haptic* projection and β-VED *push-audio* to *push-haptic* projection.

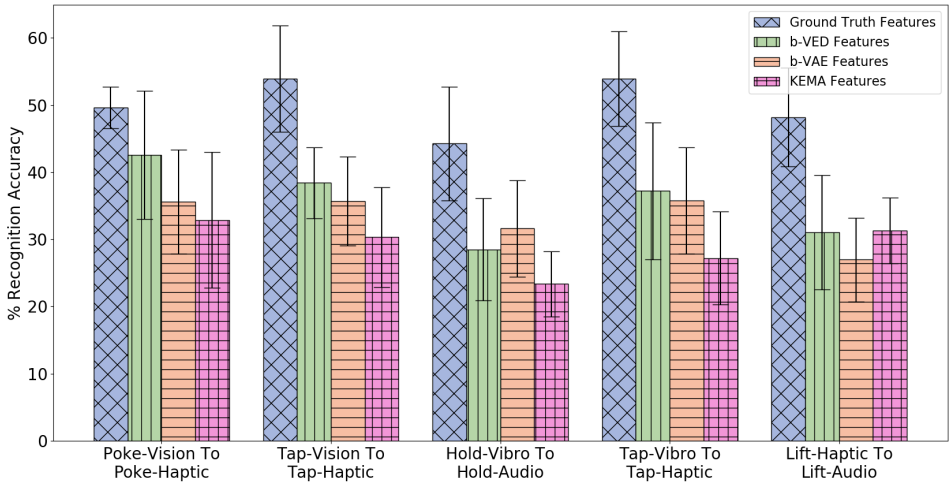


Figure 7.10: β-VED cross-perception projections where the Accuracy Delta is minimum and corresponding β-VAE projections and KEMA projections.

projections with the least accuracy delta and corresponding single source β-VAE projections and KEMA projections. Recovering *haptic* features from *vibrotactile* and *vision* was the easiest task indicating that knowing what an object's surface feels and looks like when interacting with it can inform how much force would be felt when performing that behavior. Fig. 7.10 also shows that the single source β-VAE produce comparable recognition rates as β-VED.

A statistical analysis of the projections shown in Fig. 7.10 was performed using a two-sample t-test. The t-test produced a p-value when a knowledge transfer method is compared with another, and p-value < 0.05 was considered statistically significant. For all the projections the p-value is less than 0.05 when KEMA is compared with β-VED and β-VAE except *lift-haptic* to *lift-audio*, where the p-value is 0.94 for KEMA and β-VED, and 0.11 for KEMA and β-VAE. This shows that the performance of encoder-decode methods is significantly better than KEMA in most cases.

For 2 sources β-VAE, we evaluated 3 mappings: *audio* and *vision* to *haptic*, *audio* and *vision* to *vibro*, and *haptic* and *vibro* to *vision*. Results in Fig. 7.11 indicate that by knowing how an object looks like and sounds like when performing a behavior gives a good idea of how its surface would feel and how much force would be felt performing that behavior. However, it is hard to predict how an object looks like by knowing its *haptic* and *vibro* signal, which is intuitive as objects in different category may have similar weights, but look very different. For all projections shown in Fig. 7.11, the p-value is less than 0.05 when β-VAE (2 sources) is compared with the better performing source robot among the two corresponding source robots using β-VED method.

Accuracy Delta Results Fig. 7.12 shows the accuracy delta for all 9 behaviors for β-VED model. Darker color indicates lower accuracy delta, and thus the diagonal is black. If a particular sensor fails Fig. 7.12 informs which source sensor would be better to recover its sensory signal, depending on the behavior. For example for the *poke* behavior, if the *haptic* sensor fails, using the *vision* sensor to recover its signal

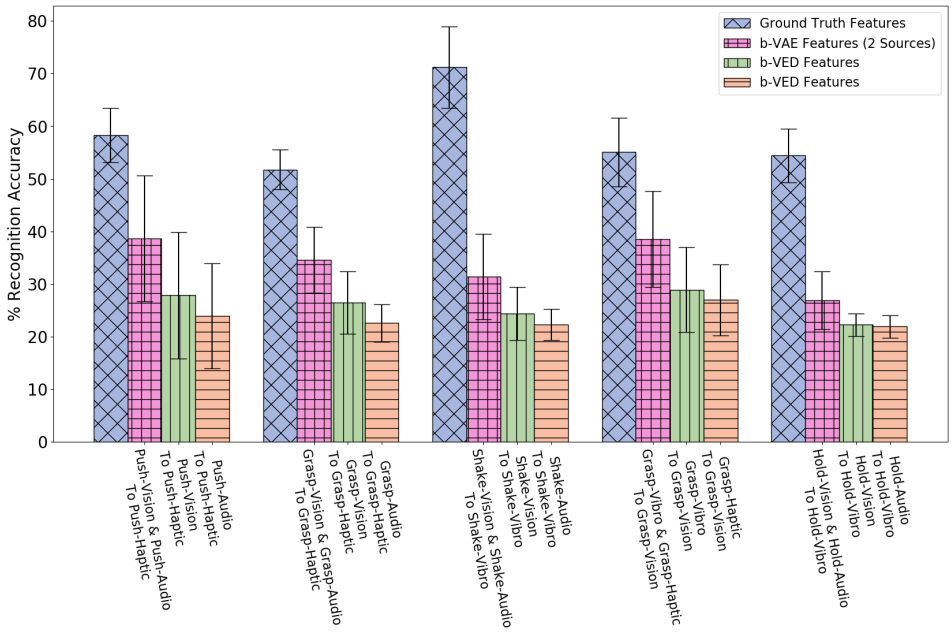


Figure 7.11: Two sources β-VAE cross-perception projections where the recognition accuracy improves as compared with corresponding β-VED projections.

would be better than other source contexts as it achieves the smallest accuracy delta. Similarity, for the *hold* behavior, if the *audio* sensor fails, the *vibrotactile* sensor is a good source context to recover its signal. These results also show that the best source modality for reconstructing features from another modality varies by behavior. The recognition accuracy of some of these projections is shown in Fig 7.10.

7.4.5.2 Cross-Behavioral Sensorimotor Transfer

Next, we consider the case where a robot is tasked with learning a mapping from one of its behaviors (e.g., *shake*) to another (e.g., *hold*) for different or same modality. Such a mapping would be useful if a new robot that has limited experience with objects needs to learn from more experienced robots that have thoroughly explored the objects in the domain.

Illustrative Example Suppose the source robot performs *shake* while the target robot performs *hold*. Projecting the *haptic* features from *shake* to *hold*, allows the

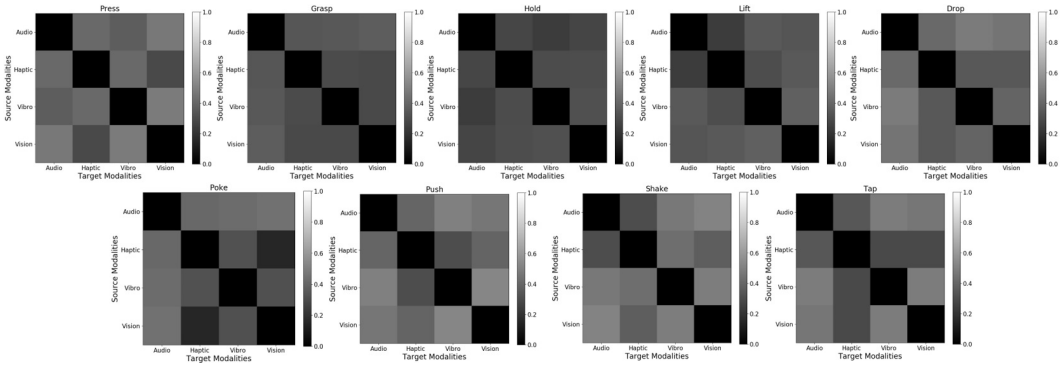


Figure 7.12: Cross-perception Accuracy Delta for 9 behaviors using β-VED. From top to bottom and from left to right: (1) *press*, (2) *grasp*, (3) *hold*, (4) *lift*, (5) *drop*, (6) *poke*, (7) *push*, (8) *shake* and (9) *tap*. Darker color means lower Accuracy Delta (better) and lighter color means higher Accuracy Delta (worse).

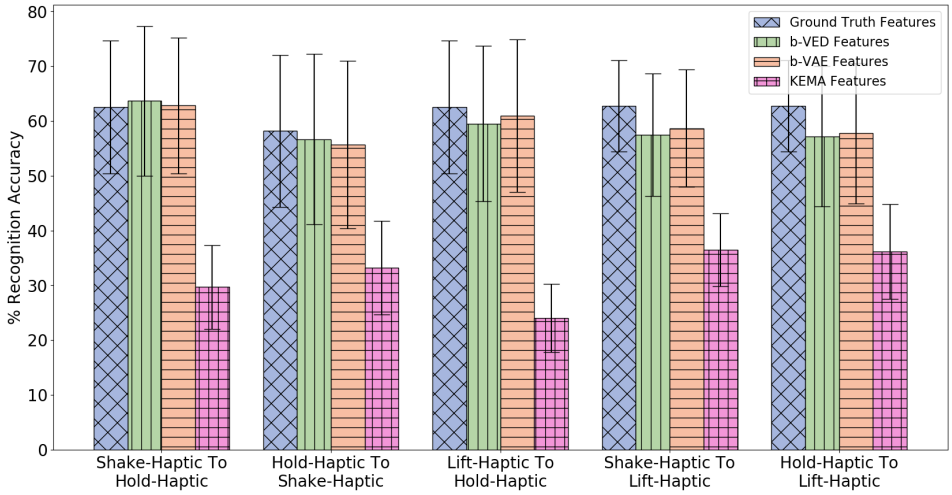


Figure 7.13: β-VED cross-behavior projections where the Accuracy Delta is minimum and corresponding β-VAE projections and KEMA projections.

target robot to attain 63.3% recognition accuracy compared with 62.5% when using ground truth features from real interactions (shown in Fig. 7.13). In other words, the target robot's recognition model is as good as it could have been if it were trained on real data.

To visualize the projection between the *shake-haptic* and *hold-haptic* contexts, we reduced the dimensionality of the generated and the ground truth features of the 5 categories the target robot never interacted with to 2 (shown in Fig. 7.14) using Principal Component Analysis [TB99] implemented in scikit-learn [PVG+11]. Fig. 7.14 shows the clusters of the ground truth features (top-left) and five plots

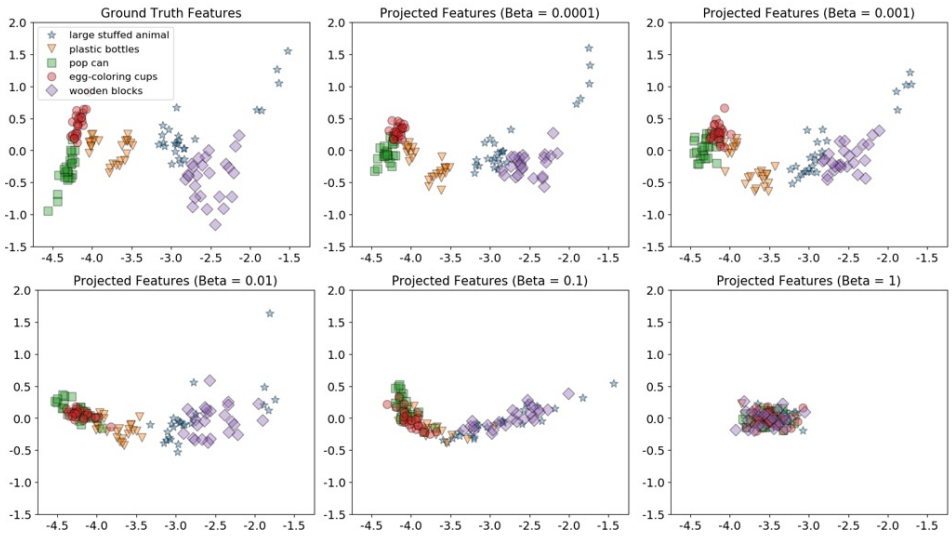


Figure 7.14: 2D visualizations using Principal Component Analysis of the target robot's *hold-haptic* ground truth features (top-left) and five β-VED projected features' (from *shake-haptic*) clusters for different β values (in increasing order from top to bottom and left to right).

that show β-VED projected features for different β values (in increasing order from top to bottom and left to right). The plots clearly show that, as the model was less constrained, the model learned better representations of the 5 categories indicated by the 5 clusters. The clusters of projected features ($\beta = 0.0001$) look structurally very similar to the ground truth data, indicating that the "reconstructed" features generated by the source robot are realistic. In the remaining experiments, we used 0.0001 as the β value for β-VED and β-VAE.

Now, consider a case of two source robots: one performs *lift* behavior and another performs *press* behavior, while the target robot performs *poke* behavior. Projecting *lift-haptic* and *press-haptic* features to *poke-haptic* by β-VAE as two sources, enables the target robot to achieve 39.3% recognition accuracy. This is a significant boost in accuracy as projecting *lift-haptic* alone to *poke-haptic* achieves 30.2%, and projecting *press-haptic* alone to *poke-haptic* achieves 28.7% (shown in Fig. 7.15).

To find the effect of the amount of data used to train a two sources β-VAE and corresponding two β-VEDs on the recognition performance, we varied the number of shared categories used to learn a projection. Fig. 7.16 shows the

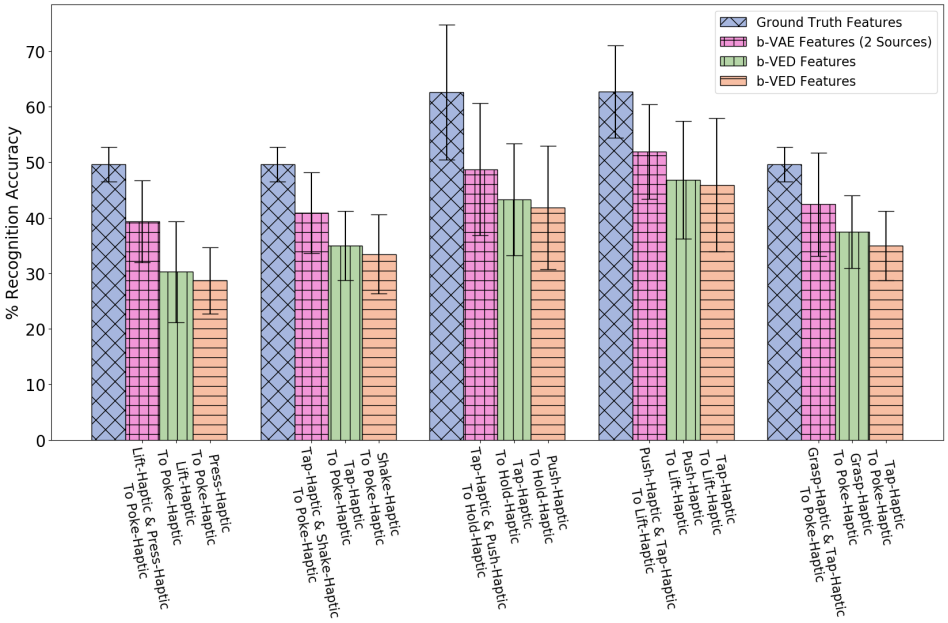


Figure 7.15: Two sources β-VAE cross-behavior projections where the recognition accuracy improves as compared with corresponding β-VED projections.

recognition performance for different numbers of shared object categories for β-VAE *lift-haptic* and *press-haptic* to *poke-haptic* projection, β-VED *lift-haptic* to *poke-haptic* projection and β-VED *press-haptic* to *poke-haptic* projection. Combining *lift-haptic* and *press-haptic* features improves the generation of *poke-haptic* features, especially with more shared categories, and the performance of two sources β-VAE reaches very close to the accuracy achieved when using ground truth features.

Accuracy Results of Category Recognition Since there are 4 modalities (*audio*, *haptic*, *vibro* and *vision*) there are $4 \times 4 = 16$ possible mappings from the source to the target robot (e.g. *audio* to *audio*, *audio* to *haptic*, *audio* to *vibro*, *audio* to *vision*, etc.). Each of the 9 behaviors are projected to all the other 8 behaviors, so for each mapping, there are $9 \times 8 = 72$ projections. Fig. 7.13 shows the 5 projections where the accuracy delta is minimum among all $16 \times 72 = 1152$ projections using β-VED and corresponding single source β-VAE projections and KEMA projections. Generally, mappings within the same modality (e.g. *haptic* to *haptic*, *vision* to *vision*) achieve higher accuracy than mappings between different modalities. This

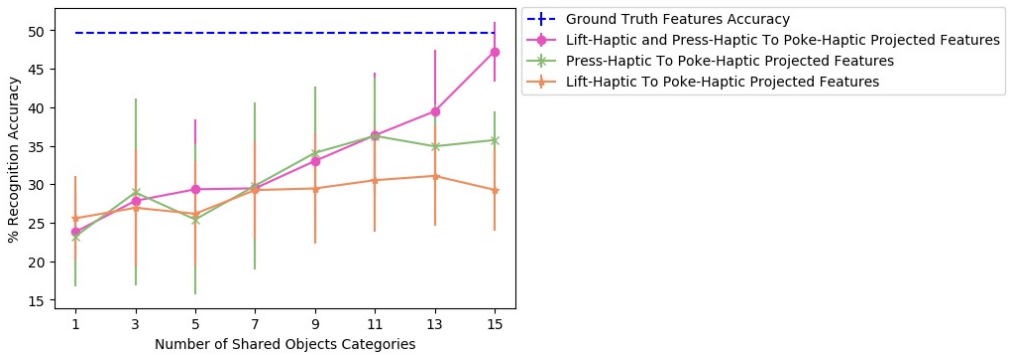


Figure 7.16: Accuracy achieved by the projected features of the target robot for different number of shared objects for β-VAE *lift-haptic* and *press-haptic* to *poke-haptic* projection, β-VED *lift-haptic* to *poke-haptic* projection and β-VED *press-haptic* to *poke-haptic* projection.

indicates that knowing what an object feels like when performing a behavior can help to predict what it would feel like better than what it would sound like or look like given another behavior. Similar to cross-perception projection results, the single source β-VAE achieves similar recognition rates as β-VED. For all the projections shown in Fig. 7.13, the p-value is less than 0.05 when KEMA is compared with β-VED and β-VAE indicating that encoder-decode methods perform significantly better than KEMA.

The β-VAE architecture requires the target robot's features as input as well as output. Since we assume that the target robot did not explore objects from the 5 novel categories, we cannot provide its features as input. Therefore, while training with the 15 categories we compared feeding zero as target robot input and feeding actual target robot's features. We found that the performance is better when we feed in zero (shown in Fig. 7.17). It may be due to the different training and test conditions that causes feeding actual features as target robot's input to perform poor as compared to feeding it zero. Thus, while training as well as testing we feed in zero as input for the target robot and the β-VAE learns to generate the target robot's features. For the first four projections shown in Fig. 7.17, the p-value is less than 0.05 when β-VAE trained using zero as features is compared with β-VAE trained using actual features.

For 2 sources β-VAE, we evaluated *haptic* and *haptic* to *haptic* mapping

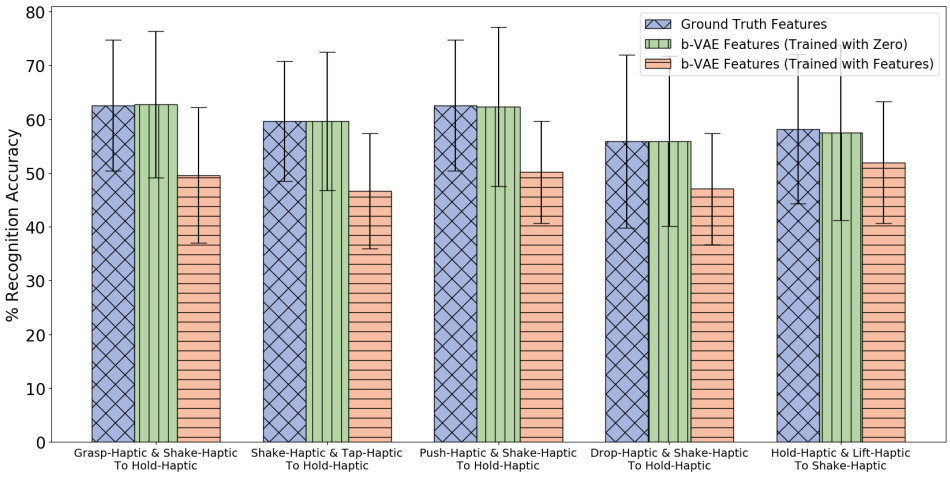


Figure 7.17: Two sources β-VAE cross-behavior projections trained with zeros for target robot where the Accuracy Delta is minimum and corresponding β-VAE projections trained with target robot's features.

because *haptic* to *haptic* is the best performing mapping for the single source robot scenario. Results in Fig. 7.15 indicate by knowing how an object feels like when performing two different behaviors provides a better prediction of how it would feel like when a third behavior is performed. In Fig. 7.15, for the first projection the p-value is less than 0.05 when β-VAE (2 sources) is compared with the better performing source robot among the two corresponding source robots using β-VED method.

Accuracy Delta Results Comparatively, mappings with target modality as *haptic* achieve smallest accuracy delta. The accuracy delta for β-VED of all the four possible mappings with target modality as *haptic* are shown in Fig. 7.18. This result indicates that it is easier to predict what an object would feel like when performing a behavior by knowing what it looks like or what it sounds like when performing another behavior. In addition, when both robots perform behaviors that capture similar object properties, the generated features are more realistic. For example, holding an object provides a good idea about how it would feel like to lift that object as indicated by smaller accuracy delta. Generating *hold-audio* features from most of the source robot's features is relatively easier possibly because holding an object

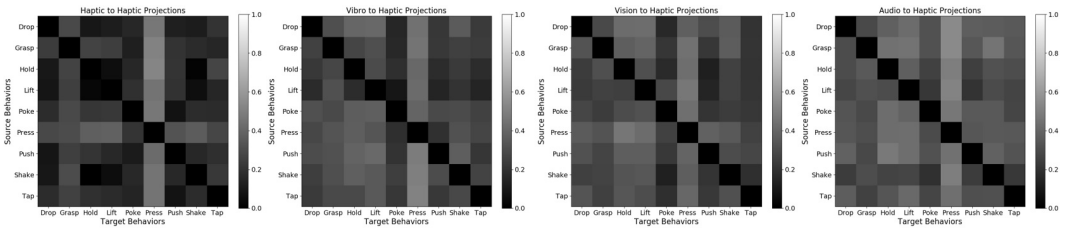


Figure 7.18: Accuracy Delta for 4 mappings using β-VED: *haptic* to *haptic*, *vibro* to *haptic*, *vision* to *haptic*, *audio* to *haptic*. Darker color means lower Accuracy Delta (better) and lighter color means higher Accuracy Delta (worse).

would not produce much sound. However, when the target modality is *vibro*, the accuracy delta is relatively higher, indicating that it is hardest to predict what an object's surface feels like when performing a behavior by knowing what it sounds like or what it looks like when performing another behavior. For example, *grasp-audio* to *push-vibro* and *drop-vibro* to *push-vibro* are the two projections where the accuracy delta is the highest.

There are 36 sensorimotor contexts (9 behaviors x 4 modalities). To find the combination of source and target contexts that is good for knowledge transfer, we computed the accuracy delta matrix, which has an average of accuracy delta values for each pair of contexts. For example, for the projection *lift-haptic* to *hold-haptic* the accuracy delta is 3% and *hold-haptic* to *lift-haptic* the accuracy delta is 5.5%, so the average accuracy delta of this pair of context is 4.2%. The size of the accuracy delta matrix is 36 x 36 and the accuracy delta value of identical contexts is 0. Fig. 7.19 shows a two dimensional ISOMAP [TDSL00] embedding of the accuracy delta matrix. Each dot in the plot corresponds to a context and the distance between a pair of context indicate the efficiency of the transfer (i.e. a pair that is closer to each other is better for knowledge transfer than a pair that is farther). Contexts with the same modality appear closer to each other suggesting that projections within the same modality comparatively perform better. Some of the most efficient pair of behaviors are *hold* and *lift*, *shake* and *hold*, and *drop* and *lift*. This shows that behaviors that capture similar object properties are better for knowledge transfer as each of these pairs of behavior require the robot to keep the object between its

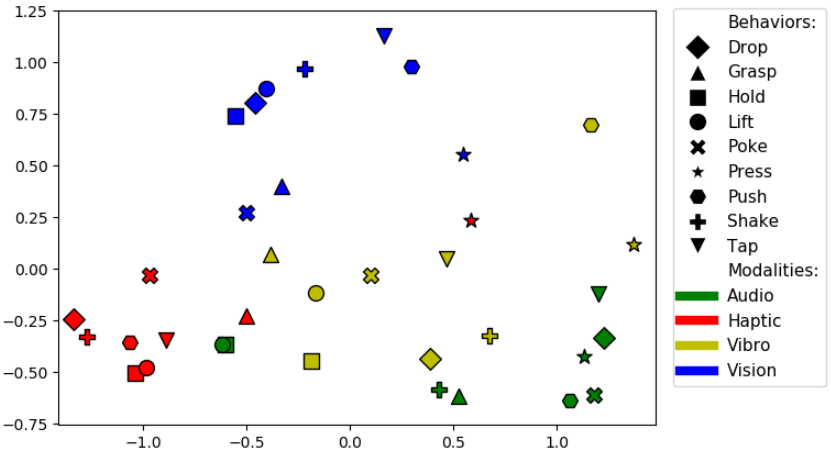


Figure 7.19: Two dimensional ISOMAP embedding of the accuracy delta matrix. Each point represents a sensorimotor context (i.e., a combination of a behavior and sensory modality). Points close in this space represent contexts between which information can be transferred effectively.

grippers for some moment and capture the force felt and images observed in a similar manner by performing both behaviors.

A surprising result is that the *hold-audio* and *lift-audio* contexts are clustered closely with the *haptic* contexts, far away from other *audio* contexts. Upon closer examination, the volume of the sounds produced by the robot's motors when holding or lifting an object was correlated with the object's weight, and thus, the *audio* data served as a proxy haptic sensor for those two behaviors. The results can also be used to detect redundant behaviors – e.g., the *hold* and *lift* behaviors are close to each other in the *haptic*, *audio*, and *vision* modalities, suggesting that they provide essentially the same information. It is important to note that these findings are likely specific to the particular robot, behaviors, and sensory modalities used in this dataset. We expect that the relationships between such sensorimotor contexts will vary depending on the robot and its means of perceiving and interacting with objects in its domain.

Object Selection for Calibration In many situations it is possible that the source and the target robots have limited time to build the mapping function for knowledge transfer. Therefore, it is important to efficiently select the calibration set

of objects explored by both robots to maximize the quality of the learned mapping in a limited time. Here we propose one such procedure.

Let $\mathcal{D}^{c_s}_{source}$ be the dataset of observed features by the source robot in context c_s. These include features with objects from all categories $\mathcal{Y}$. The goal is to select a set of N objects $\mathcal{O}_{calibration}$ with category labels in $\mathcal{Y}_{shared}$ which can then be explored by the target robot in some context c_t in order to learn the source to target mapping function.

1. Cluster the data points in $\mathcal{D}^{c_s}_{source}$ into J clusters
2. For each cluster v_j, compute a weight w_j according to:

$$w_j = \frac{\text{\# of } v_j \text{ data points with labels in } \mathcal{Y}_{target}}{\text{Total \# of data points in cluster } v_j}$$

3. Sample a cluster v_j with probability proportional to its weight, and then uniformly sample an object with label in $\mathcal{Y}_{shared}$ for which a data point falls into v_j in the clustering. Repeat N times (without replacement).

We tested this procedure with K-means [Llo82] to cluster 500 data-points (100 objects x 5 trials with each object) of the source robot into $J = 100$ clusters, and select objects from clusters that capture similar object properties that are more useful for calibration. We limited the size of $\mathcal{O}_{calibration}$ to $N = 5$, substantially less than in results reported so far.

Fig. 7.20 compares the method to two naive baselines: 1) randomly selecting a category in $\mathcal{Y}_{shared}$ and then using data with all 5 objects in that category; and 2) randomly sampling 5 objects with labels $\mathcal{Y}_{shared}$. As demonstrated, selecting 5 objects using the clustering method achieves higher accuracy than randomly selecting 5 objects or a category. This means that the clustering method selects objects that are similar to the 5 target categories, and can be useful when there is a budget of the number of objects both robots are allowed to interact with.

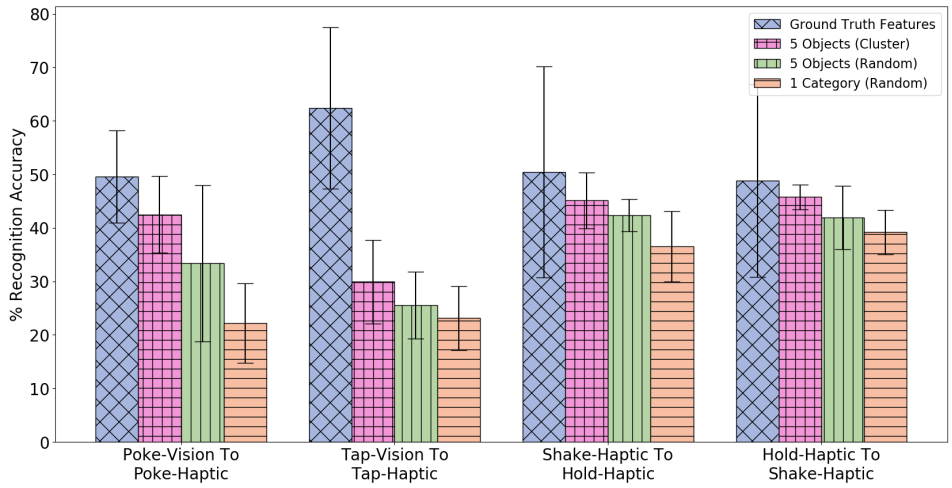


Figure 7.20: Comparison of three different methods of selecting 5 objects for training β-VED. Note that each method selects 25 data-points for training β-VED.

7.4.6 Validation on a Second Dataset

We validated our knowledge transfer framework on another dataset, which is described below along with the evaluation methodology and experimental results.

7.4.6.1 Dataset Description

We used another publicly available dataset collected by [SKSS16], in which a Kinova MICO arm with 6-DOF explored 32 objects using 8 behaviors: *grasp*, *lift*, *hold*, *look*, *lower*, *drop*, *push* and *press*. During the execution of each action (other than *look*) the robot recorded the sensory perceptions from the haptic and the auditory sensory modalities. The haptic signals were recorded for the robot's 6 joints at 15 Hz while the auditory signals was represented as the Discrete Fourier Transform computed with 65 frequency bins. Before grasping the object, the *look* behavior was performed, which produced three different types of visual sensory modalities: 1) an RGB color histogram using 8 bins per channel; 2) Fast point feature histogram (fpfh) shape features and 3) deep visual features produced by feeding the image to the 16-layer VGG network. For additional details on the visual feature extraction pipelines, please consult [TSS$^+$16]. Each behavior was executed 5 times on each of the 32 objects, resulting in 1,280 interactions (8 behaviors x 5 trials x 32 objects). For

additional details regarding the dataset, readers can refer to [SKSS16].

7.4.6.2 Evaluation and Results

The evaluation procedure for this dataset was the same as that for the previous dataset except that instead of recognizing object categories, the robot had to recognize specific objects as the objects in this dataset did not belong to any object categories. We assume that the source robot interacts with all 32 objects, while the target robot interacts with only 24 randomly selected objects. The objects experienced by both robots are used to train the knowledge transfer model and the trained knowledge transfer model is used to generate "reconstructed" sensory signals of the objects that the target robot never interacted with. To train the object recognition model, we again consider three possible training cases previously described with a difference that here we performed 5-fold trial-based cross-validation, where the training phase consisted of 4 trials from each of the object that the target robot never interacted with and the test phase consisted of the remaining trial. Since the robot interacted with each object 5 times, there were 32 (8 objects x 4 trials) examples in the training set, and 8 (8 objects x 1 trials) examples in the test set. This process was repeated 5 times, such that each trial was included in the training set 4 times and once in the test set. The entire procedure of training the knowledge transfer model and object recognition model is repeated 10 times to get an accuracy estimate. Note that the hyperparameters and the structure of the network were kept identical to those that were used for the previous dataset without any additional tuning. The results of cross-perception and cross-behavioral sensorimotor transfer are discussed as follows.

7.4.6.3 Illustrative Example

Consider a cross-behavioral sensorimotor transfer where the source robot uses the *lift* behavior while the target robot uses the *lower* behavior. Projecting *haptic* features from *lift* to *lower*, allows the target robot to achieve a recognition accuracy of 68% compared with 52.2% when using ground truth features (shown in Fig. 7.23). In

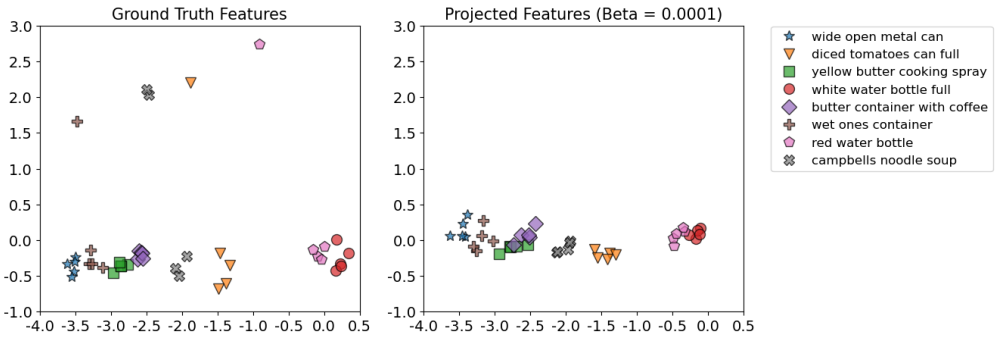


Figure 7.21: 2D visualizations using Principal Component Analysis of the target robot's *lower-haptic* ground truth features and β-VED projected features' (from *lift-haptic*) for the dataset in [SKSS16].

other words, the target robot's object recognition model performs better than if it were trained on real data.

To visualize the *lift-haptic* to *lower-haptic* projection, we reduced the dimensionality of the generated and the ground truth features of the 5 objects the target robot never interacted with into 2D space by PCA (shown in Fig. 7.21). Fig. 7.21 shows the clusters of both ground truth and projected features using β-VED. The clusters of projected features not only look very similar to the ground truth features, but also have less variance, which may account for the higher recognition rate when using reconstructed features.

7.4.6.4 Accuracy Results of Object Recognition

Cross-Perception Sensorimotor Transfer Since there are 2 modalities (*audio* and *haptic*), if a sensor fails, there is 1 possible mapping from the available sensor to the failed sensor, so there are $2 \times 1 = 2$ possible mappings (e.g. *audio* to *haptic* and *haptic* to *audio*). There are 7 interactive behaviors, so there are $2 \times 7 = 14$ projections (e.g. *hold-haptic* to *hold-audio* and *lower-audio* to *lower-haptic*, etc.). There are also 3 vision based modalities (*color*, *shape* and *vgg*) only for *look* behavior, so there $3 \times 2 \times 1 = 6$ more projections (e.g. *look-color* to *look-shape* and *look-vgg* to *look-color*, etc.). Thus, in total there are 20 cross-perception projections. Fig. 7.22 shows the 5 β-VED cross-perception projections with the least accuracy delta and

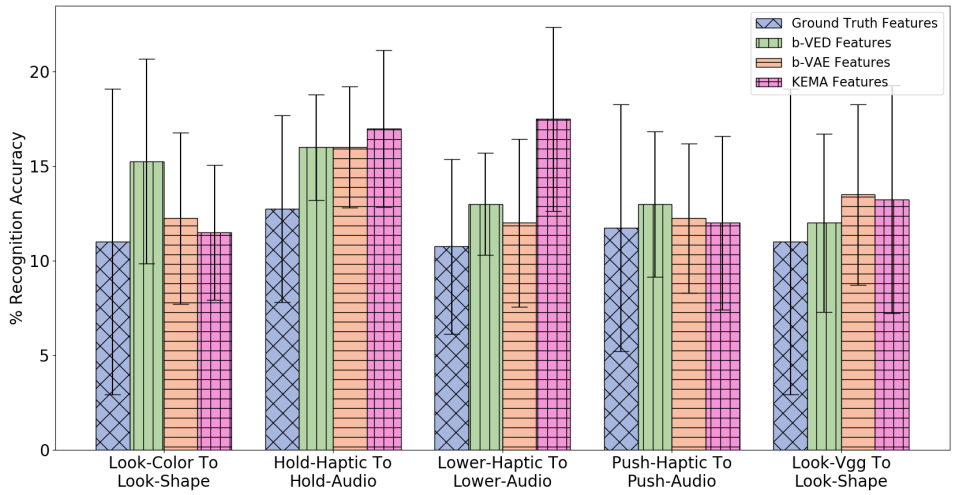


Figure 7.22: β-VED cross-perception projections where the Accuracy Delta is minimum and corresponding β-VAE projections and KEMA projections for the dataset in [SKSS16].

corresponding single source β-VAE projections and KEMA projections. Note that the reconstructed features of these 5 projections achieve higher accuracy than the ground truth features, however there are projections such as *look-shape* to *look-vgg* and *hold-audio* to *hold-haptic*, where ground truth features achieve higher accuracy. Recovering *audio* features from *haptic* was the easiest task, indicating that knowing how forces felt when performing a behavior can inform how the object would sound when performing that behavior.

Cross-Behavioral Sensorimotor Transfer Since there are 2 sensory modalities (*audio* and *haptic*), there are $2 \times 2 = 4$ possible mappings from the source to the target robot (e.g. *audio* to *haptic* and *haptic* to *audio*, etc.). Each of the 7 interactive behaviors are projected to each of the other 6 behaviors, so for each mapping, there are $7 \times 6 = 42$ projections (e.g. *lift-haptic* to *lower-haptic* and *hold-audio* to *lower-haptic*, etc.). Thus, there are $4 \times 42 = 168$ projections without using vision modalities. Since there are also 3 visual modalities (*color*, *shape* and *vgg*) only for *look* behavior, we projected visual modalities to non-visual modalities $3 \times 2 = 6$ mappings, and non-visual modalities to vision modalities $2 \times 3 = 6$ mappings for *look* behavior to other behaviors $1 \times 7 = 7$ projections and other behaviors to *look*

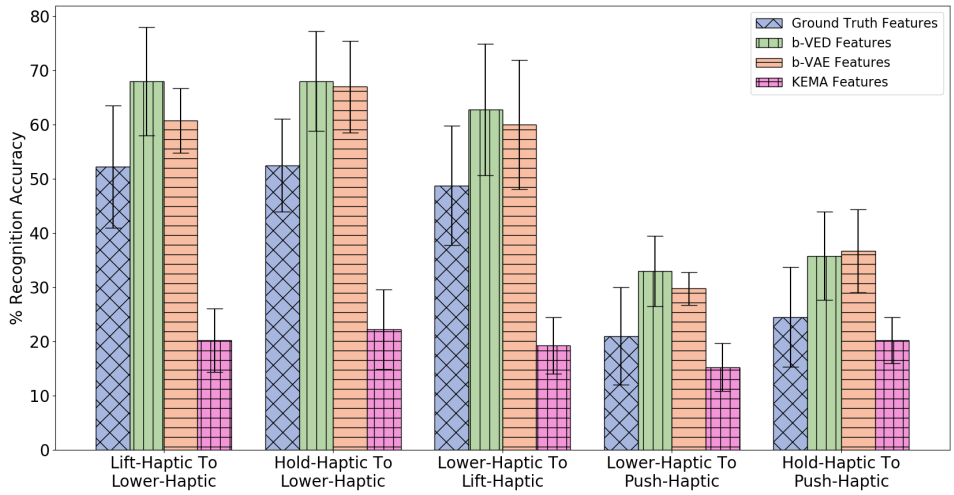


Figure 7.23: β-VED cross-behavior projections where the Accuracy Delta is minimum and corresponding β-VAE projections and KEMA projections for the dataset in [SKSS16].

behavior $7 \times 1 = 7$ projections. Thus, there are $6 \times 7 + 6 \times 7 = 84$ projections using vision modalities, making $168 + 84 = 252$ total cross-behavioral projections. Fig. 7.23 shows the 5 β-VED cross-behavioral projections where the accuracy delta is minimum and corresponding single source β-VAE projections and KEMA projections. While the reconstructed features of these 5 projections achieve higher accuracy than the ground truth features, there are projections such as *push-haptic* to *look-vgg* and *look-shape* to *hold-haptic*, where ground truth features achieve higher accuracy. Similar to previous results, mappings within the same modality (e.g. *haptic* to *haptic*) achieve higher accuracy than mappings between different modalities. One interesting similarity is *haptic* to *haptic* which is the best performing mapping for the previous dataset and *haptic* to *haptic* is the best performing mapping for this dataset. Moreover, the best performing combination of the source and target behaviors are also similar. For example, in the previous dataset *lift-haptic* to *hold-haptic* projection generated very realistic features and in this dataset *lift-haptic* to *lower-haptic* projection has a very low accuracy delta. This shows that the source and target behavior combination that generates realistic features can be applied to different robots.

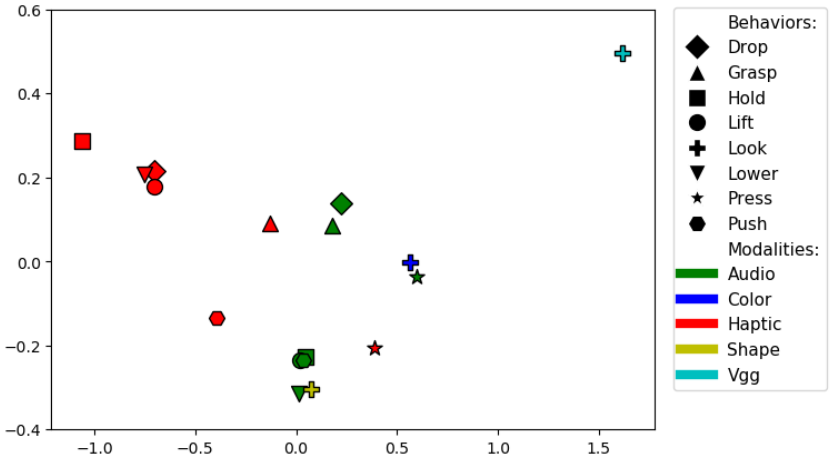


Figure 7.24: Two dimensional ISOMAP embedding of the accuracy delta matrix for the dataset in [SKSS16]. Each point represents a sensorimotor context (i.e., a combination of a behavior and sensory modality). Points close in this space represent contexts between which information can be transferred effectively.

7.4.6.5 Accuracy Delta Results

There are 17 sensorimotor contexts (7 behaviors x 2 non-visual modalities + 1 behavior x 3 visual modalities). To visualize the combination of source and target contexts that are good for knowledge transfer, we plotted the two dimensional ISOMAP [TDSL00] embedding of the accuracy delta matrix (shown in Fig. 7.24) as we did for the previous dataset. Some of the most efficient pairs of behaviors are *lift* and *lower* and *grasp* and *drop*. Similar to previous results, contexts with the same modality appear closer to each other indicating that projections within the same modality perform better than projections within different modalities. Moreover, pair of behaviors such as *lift* and *lower* that capture similar object properties are better for knowledge transfer similar to previous dataset. Some exceptions include the *look-shape* context which lies close to several of the contexts that use the *audio* modality. The *press-haptic* context lies slightly outside the remaining *haptic* contexts as unlike behaviors such as *lift* and *lower*, the *press* action does not give the robot information about the object's mass, but rather, it captures its compliance.

7.5 Summary

Behavior-grounded sensory object knowledge is specific to each robot's embodiment, sensors, and actions which makes it difficult to transfer multisensory representations from one robot to another. We proposed and evaluated a framework for knowledge transfer that uses variational auto-encoder and encoder-decoder networks to project sensory feedback from one robot to another robot across different behaviors and modalities. The framework enables a target robot to use knowledge from a source robot to classify objects into categories it has never interacted with before. In addition, using the proposed knowledge transfer method the target robot can recover the features of a failed sensor from the available sensors. In this way, the target robot, instead of learning a classifier from scratch, can start immediately with a classifier that performs nearly as good as if the target robot learned by collecting its own labeled training set through exploration. We also proposed a method to select a set of objects that would be better to transfer knowledge in a time constrained situation where the robots cannot interact with a large number of objects to train the knowledge transfer model. Moreover, we successfully validated the proposed knowledge transfer framework on another dataset without any additional hyperparameter tuning. These results address some of the major challenges in the deployment of interaction based multisensory models, namely that they require a large amount of interaction data to train and cannot be directly transferred across robots.

There are several closely-related research problems that can be addressed in the future work. First, a limitation of the our dataset is that the sensory features are dependent on the robot's environment, so the transferred features would not apply to the robot in a different environment. For instance, a pencil box would produce different auditory and visual features when dropped on a wooden table than when dropped on a soft cushion. Thus, there is a need to develop a framework to transfer knowledge that can generalize across different environments. Moreover, the dataset used in our experiments is relatively small (for each object category there are only 25 examples), and thus, not large enough to answer questions like "how much data is

required to reach the optimal performance?" Thus, in future work we would collect a relatively larger dataset that can answer this question.

Another limitation of our experiment is that the dataset we used contains only one robot, and thus we considered the case where the source and target robots are morphologically identical but differ in terms of behaviors and sensory modalities. In future work, we plan to evaluate our framework on robots that not only perform different behaviors, but also have different embodiment and feature representations. In addition, the run-time complexity of the β-VAE model we presented increases linearly with the increase in the number of source robots used. Having a model that can scale with the number of robots without increasing run-time complexity could improve the proposed method. A model that can incrementally improve performance by learning from new data-points acquired by one of the robots is also a promising avenue for future exploration. Finally, in our experiments, we addressed a category recognition task. In future work, we plan to extend the framework to handle sensorimotor knowledge transfer for other tasks as well, such as manipulating objects, grounding language, etc.

Chapter 8

Transferring Implicit Knowledge of Non-Visual Object Properties Across Heterogeneous Robot Morphologies*

8.1 Introduction

Humans learn about object properties by physically interacting with objects and perceiving multiple sensory signals, including vision, audio, and touch [TVCO04a, LK87, WWCM07, EB04, Gib88, CHPS21]. Interactions based on non-visual modalities such as audio and touch are essential, because vision alone is insufficient for detecting intrinsic object properties [McC89]: e.g., detecting whether an opaque bottle is full of liquid or empty. Recent works show that learning implicit knowledge of non-visual object properties leads to robots' improved downstream performance, in material classification [EXS+20], liquid property estimation [HGY22], object categorization [TS19], and human-robot dialogue interaction [TPS+17].

A robot may learn about object properties by performing exploratory interactions on objects and analyzing the effects via a diverse set of sensors [MFHH22,

*This chapter is based on the following paper: Gyan Tatiya, Jonathan Francis, and Jivko Sinapov, "Transferring Implicit Knowledge of Non-Visual Object Properties Across Heterogeneous Robot Morphologies", *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023. [TFS23]

WCH+21, LLC22]. The immediate issue is that this process is time-consuming, as it must be repeated for each robot. A natural desire may be to *transfer* representation of the object properties to a new robot to enable it to learn faster and complete its downstream tasks more efficiently. However, if the new robot has different interaction capabilities (e.g., different sensor models, or a different physical embodiment or *morphology*), the implicit knowledge gained by the previous robot is not directly transferable to the new one. Indeed, a robot's machine learning model for the interactive perception tasks cannot be naturally applied to another robot because these models are specific to each robot's embodiment, sensors, and environment [FKL+22]. While there is a great need to transfer implicit knowledge of object properties across heterogeneous robot morphologies, obtaining a general-purpose representation to facilitate rapid learning has remained challenging.

To address this challenge, we propose a framework that leverages learned projection functions to transfer implicit knowledge of non-visual object properties from a more-experienced source robot to a newly-deployed target robot. Specifically, we consider the general encoder-decoder network (EDN) model class [BKC17] and the kernel manifold alignment (KEMA) method [TSES20, LLL+18, WM11] as projection functions for learning object property-based and object identity-based correspondences. To test our framework, we collected a dataset of two robots, *Baxter* and *UR5*, that performed eight behaviors on 95 objects. We evaluate our framework on two tasks: object-property and object-identity recognition tasks. The results of our experiments show that KEMA learned using object identity-based correspondence consistently outperforms EDN in both tasks indicating transferring knowledge from robots to a shared latent space boosts the performance of the target robot. Furthermore, we propose a data augmentation technique independent of the learning task and show that using our data augmentation technique improves the models' generalization and prevents overfitting.

This chapter investigates how robots can transfer implicit knowledge of non-visual object properties across diverse robot embodiments, utilizing the Encoder-Decoder Network (EDN) from Chapter 5 and the Kernel Manifold Alignment (KEMA)

method from Chapter 6. The dataset, collected by two heterogeneous robots (Chapter 3), serves for evaluation. Unlike Chapter 5 and Chapter 7 that focused on building object-identity correspondences to learn projection functions, this chapter learns projection functions based on building object-property correspondences rather than object-identity correspondences. Evaluation includes object-property and object-identity recognition tasks unlike object category recognition task in Chapters 5 and 7. This chapter builds upon the methodologies introduced in Chapters 5 and 6, extending the application of knowledge transfer to object-property and object-identity recognition tasks.

8.2 Related Work

8.2.1 Interactive object perception

Studies in psychology and cognitive science show that humans manipulate objects in multiple stages to extract information about their properties, such as texture, stiffness, temperature, and weight [KL92, LK93, DFBP02]. In addition, the human brain leverages a multisensory representation when recognizing object properties, enabling flexible generalizability to unknown contexts [LCS07, LS14]. Recent advances in intelligent robotics consider integrating multisensory information acquired by object exploration [BHS+17, TS19, PGGG+20, SLZ+20, NGTJJ22, LKS+20, WWW+22], where one challenge is that the implicit knowledge acquired by one robot through interactive perception cannot be directly transferred to another robot: the unique nature of the robot's embodiment drastically affects the sensed data distribution and resultant model that each robot learns. Whereas the focus of prior work has been limited to learning from scratch for each robot [SSS+14a, FLN+19, THCHS19], this is prohibitively expensive at scale, e.g., for a fleet of heterogeneous robots. We propose a framework for transferring implicit knowledge about object properties from a source robot to a target robot.

8.2.2 Transferring knowledge of object properties

Recent work demonstrates that implicit knowledge from the interactive object perception can be transferred across sensor models and robots [FLN+19, THCHS19, TS19, TSES20, SSS+14a, THHS20]. In [SSS+14a], a robot performed interactive object perception to improve object category recognition. As implicit knowledge transfer was not the focus of that work, experiments were conducted on only a single robot. Moreover, whereas object properties may sometimes be the same for objects in different categories (e.g., bottles and cups can have similar colors, contents, and weights), their method encouraged unconstrained feature similarity based on object category alone, compromising prospects for transferring the features across robots or tasks. Our cross-robot transfer approach jointly learns to distinguish between different categories while leveraging learned similarities across properties. In [THCHS19], authors consider object categorization under a transfer learning paradigm, wherein an encoder-decoder network was used to generate a "target" robot's features from a "source" robot's learned representation. The authors use only a single robot in their experiments; however, so inherent challenges introduced by different robot morphologies remain to be studied. The approach in [TSES20] was used to project features from 3 robots with different embodiments to a shared latent space for object-identity recognition. However, their experiments consisted of only simulated robots that recorded only one sensor modality (effort) during interaction with objects that varied in only one dimension (weight). To address these shortcomings, we collect a multisensory dataset using two real robots with different morphologies that explore 95 objects that vary by color, weight, and contents. We develop a multi-stage projection method for implicit knowledge transfer across two heterogeneous robots, and we evaluate our approach on object-property recognition and object-identity recognition.

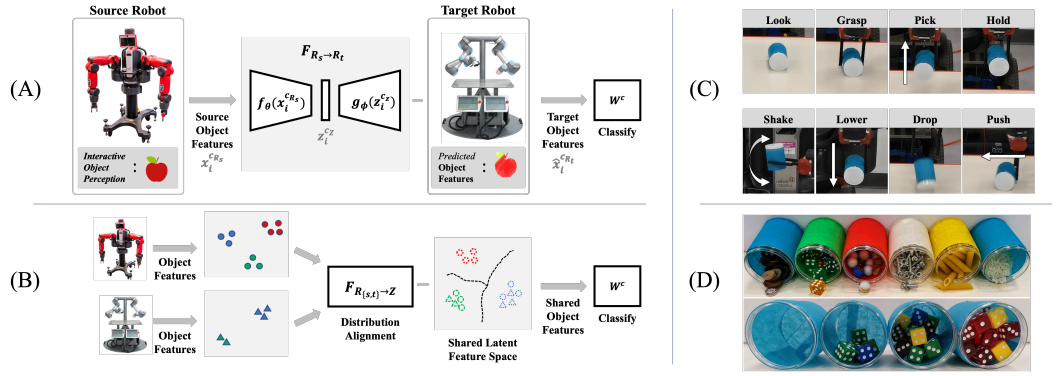


Figure 8.1: (A) Shows projection from *Baxter* to *UR5* using Encoder-Decoder Network (EDN). (B) Shows projection from *Baxter* and *UR5* to a shared latent space using Kernel Manifold Alignment (KEMA). (C) The 8 exploratory behaviors used to learn about the objects. (D) The 95 objects used in this study vary in: **(top)** colors (*blue*, *green*, *red*, *white*, and *yellow*), contents (*wooden buttons*, *plastic dices*, *glass marbles*, *nuts & bolts*, *pasta*, and *rice*), and **(bottom)** weights (*empty*, *50g*, *100g*, and *150g*).

8.2.3 Interactive object perception datasets

Compared to existing object interaction datasets [SSS+14a, TSES20, GSC+22], ours offers additional value for research needs. In [SSS+14a], the dataset only contained a single robot, whereas we collected our dataset using two robots with different morphologies. In [TSES20], simulated robots were used that collected only effort signals during object interaction. In contrast, we used real-world robots and collected multiple sensory signals, including vision, audio, and haptic. In [GSC+22], the audio and tactile signals correspond to impact or touch behavior performed on 3D virtualized objects. However, we collected the visual and non-visual sensory modalities while the robots performed several exploratory behaviors (e.g., grasp, shake) on 95 real-world objects that vary in multiple dimensions (color, weight, and content). To the best of our knowledge, our dataset contains the largest number of objects, with the most dimensions of distinction ever explored by multiple real robots for transferring implicit knowledge.

8.3 Learning Methodology

8.3.1 Notation and Problem Formulation

Consider two robots with different morphologies, represented as source and target robots, or R_s and R_t respectively. For a given robot R, let $\mathcal{B}_R$ be the set of exploratory behaviors (e.g., *grasp*, *lift*) and let $\mathcal{M}_R$ be the set of non-visual sensory modalities (e.g., *audio*, *force*). Let $\mathcal{C}_R$ be the set of sensorimotor contexts, including each possible combination of a behavior in $\mathcal{B}_R$ and a sensory modality in $\mathcal{M}_R$ (e.g., *grasp-audio*, *lift-force*). For an exploration trial, the robot R performs exploratory behaviors $\mathcal{B}_R$ on a specific object and records a sensory signal for each modality in $\mathcal{M}_R$. There are n_R such exploration trials on each object. For the i^{th} exploration trial, robot R's observation feature is $x_i^{c_R} \in \mathbb{R}^{D_{c_R}}$, where $i \in \{1, ..., n_R\}$, $c_R \in \mathcal{C}_R$, and D_{c_R} is the dimension of robot R's feature space under context c_R.

Let $\mathcal{O}$ be the set of objects that vary in non-visual properties (e.g., *weight*, *sound*). We assume that the source robot has explored each object n_{R_s} times, whereas the target robot has comparatively less experience. More specifically, either the target robot has only explored a subset $\mathcal{O}_t \subset \mathcal{O}$ or explored an object for less trials than n_{R_s} (i.e., $n_{R_t} < n_{R_s}$). Our goal is to learn a projection function to transfer knowledge gained through object interaction, from the more-experienced *source* robot to the less-experienced *target* robot. We learn the projection function using the common objects experienced by both robots and transfer knowledge about the source robot's additional experience by using the learned projection function. This knowledge transfer will help the target robot learn about object properties faster, with fewer object interactions, and predict the properties of novel objects.

We consider learning two projection functions. First, the projection function $F_{R_s \to R_t}$, that projects the observation features from the source robot's feature space to the target robot's feature space. More specifically, $F_{R_s \to R_t} : x_i^{c_{R_s}} \to \hat{x_i}^{c_{R_t}}$, where $\hat{x_i}^{c_{R_t}}$ is the projected features in the target robot's feature space. Second, the projection function $F_{R \to \mathcal{Z}}$, that projects the observation features from each robot's feature space to a shared latent feature space. More specifically, $F_{R_s \to \mathcal{Z}} : x_i^{c_{R_s}} \to z_i^{c_{\mathcal{Z}}}$

and $F_{R_t \to \mathcal{Z}} : x_i^{c_{R_t}} \to z_i^{c_\mathcal{Z}}$, where $z_i^{c_\mathcal{Z}} \in \mathbb{R}^{D_\mathcal{Z}}$ and represents the shared latent features of size $D_\mathcal{Z}$. In the first mapping, we train the target robot in its own feature space; for the second mapping, we train the target robot in the shared latent space.

We also consider two ways to build correspondences between the source and the target robots, for learning the projection functions. First, object-identity correspondence, in which the source-target pair corresponds to the same object identity. It is applicable when both robots have access to the same objects. Second, object-property correspondence, in which the source-target pair corresponds to the same object property. It is applicable when both robots operate in different environments and do not have access to identical objects but have access to objects with the same properties (e.g., red and blue bowls containing rice).

8.3.2 Projection to Target Feature Space

We propose using an Encoder-Decoder Network (EDN) [THCHS19] to train the projection function $F_{R_s \to R_t}$, mapping observation features from the source robot's feature space to the target robot's feature space (Fig. 8.1A). First, encoder f_θ transforms the observation feature of the source robot $x_i^{c_{R_s}}$ into a fixed-size lower-dimensional vector $z_i^{c_\mathcal{Z}} \in \mathbb{R}^{D_z}$ of size D_z. Then, decoder g_ϕ uses this code vector $z_i^{c_\mathcal{Z}}$ to generate the predicted observation feature of the target robot $\hat{x_i}^{c_{R_t}}$. We denote this overall non-linear mapping as $F_{R_s \to R_t}$: $\hat{x_i}^{c_{R_t}} = g_\phi(f_\theta(x_i^{c_{R_s}}))$, where θ and ϕ are network parameter weights of encoder and decoder, respectively. For training the EDN, we use a dataset of source-target feature pairs $\{x_i^{c_{R_s}}, x_i^{c_{R_t}}\}_{i=1}^N$, with N training samples. We optimize EDN parameters by minimizing root mean-squared error (RMSE) between real features observed by target robot ${x_i}^{c_{R_t}}$ and "generated" target features $\hat{x_i}^{c_{R_t}}$ obtained by applying the projection to the corresponding source features: $\theta^\star, \phi^\star = \underset{\theta,\phi}{\arg\min} \sqrt{\frac{1}{N}\sum_{i=1}^N (x_i^{c_{R_t}} - \hat{x}_i^{c_{R_t}})^2}$.

Given a trained EDN, we generate the target robot's feature to transfer knowledge about the source robot's additional experience; then, using a standard multi-class classifier, we can train the target robot to recognize object properties with the "generated" features.

8.3.3 Projection to Shared Latent Feature Space

The projection $F_{R\to\mathcal{Z}}$ can be achieved through distribution alignment—organizing observation features from each robot's feature space within a shared representation (Fig. 8.1B). We illustrate this mapping via Kernel Manifold Alignment (KEMA) [TSES20], which constructs a set of domain-specific projection functions for each robot $F_R = [F_{R_s}, F_{R_t}]^T$, such that the examples of the same object property would locate closer while examples of different object properties would locate distantly. To compute the data projection matrix F_R, we minimize the cost related to the projection functions being too dissimilar: $\{F_{R_s}, F_{R_t}\} = \underset{F_{R_s},F_{R_t}}{\arg\min}(C(F_{R_s}, F_{R_t}))$. Here, $C(\cdot) = \frac{1}{\texttt{DIS}}(\mu * \texttt{GEO} + (1-\mu) * \texttt{SIM})$, where the geometry of a domain, class similarity, and class dissimilarity are represented as `GEO`, `SIM`, and `DIS`, respectively. `GEO` is minimized to preserve the local geometry of each domain by penalizing projections in the input domain that are far from each other. `SIM` is minimized to encourage examples with the same object property to be located close to each other in the latent space by penalizing projections of the same object property mapped far from each other. `DIS` is maximized to encourage examples with different object properties to be located far apart in the latent space by penalizing projections of the different object properties that are close to each other. The parameter $\mu \in [0, 1]$ regulates the contribution of the `GEO` and the `SIM` terms. For more details on KEMA, please see [TCV16]. Data in the latent feature space are comparable and can be used to train a standard multi-class classifier for different robots. The target robot can use this classifier to recognize properties of objects it has never interacted with.

8.3.4 Model Implementation and Training

Specific EDN architectures (e.g., transformers, dense convolutions, etc.) may be chosen according to the form of the data observations; in our experiments, we used an architecture that consists of three fully-connected layers for both encoder and decoder, with 1000, 500, 250 units, activation via Exponential Linear Units (ELU), and a 125-dimensional latent code vector. We use Adam [KB15] with a learning

rate of 10^{-4} to compute gradients according to RMSE, over 1000 epochs. We used Radial Basis Function (RBF) for KEMA's kernel function, with $\mu = 0.5$. We train the target robot's recognition model via a multi-class SVM with the RBF kernel. For the EDN approach, this recognition model is trained using the "generated" features from the source robot and the real features of the target robot used to train the EDN; for the KEMA approach, this recognition model is trained using the shared latent features corresponding to both robots' datapoints used to learn the KEMA projection function.

8.4 Evaluation

8.4.1 Experimental Platform and Feature Extraction

8.4.1.1 Robots and Sensors

We collected our dataset using two robots: *Baxter* [bax] and *UR5* [ur5] (Fig. 8.1A). *Baxter* has dual 7-degree-of-freedom (DOF) arms and a 2-finger gripper. We used the left *Baxter* arm for the data collection. *UR5* has 6-DOF and 2-finger Robotiq 85 gripper. *Baxter* had a PrimeSense camera mounted on its head, which captures 640×480 images, and an Audio-Technica PRO 44 microphone placed on its workstation. *Baxter* hand camera captures 480×300 images. *UR5* had an Orbbec Astra S 3D Camera mounted on its frame, which captures 640×480 images, and a Seeed Studio ReSpeaker microphone placed on its workstation. We recorded data from 14 and 11 sensor modalities for *Baxter* and *UR5*, respectively. For more dataset details, such as sampling rate, please see: `https://github.com/gtatiya/Implicit-Knowledge-Transfer`.

8.4.1.2 Exploratory Behaviors and Objects

Both robots perform 8 behaviors: *look*, *grasp*, *pick*, *hold*, *shake*, *lower*, *drop*, and *push* (Fig. 8.1C). We chose these diverse behaviors because they can capture various object properties. *Look* is a non-interactive behavior in which robots record visual

modalities (*RGB*, *Depth*, and *Point-Cloud*) from their head camera. All other behaviors are interactive, encoded as robot joint-angle trajectories. For all behaviors, *Point-Cloud* was recorded for the first 5 frames. Both robots explore 95 objects (cylindrical containers) that vary in 5 colors (*blue*, *green*, *red*, *white*, and *yellow*), 6 contents (*wooden buttons*, *plastic dices*, *glass marbles*, *nuts & bolts*, *pasta*, and *rice*), and 4 weights (*empty*, *50g*, *100g*, and *150g*) shown in Fig. 8.1D. There are 90 objects with contents (5 colors x 3 weights x 6 contents) and 5 objects without any content that only vary by 5 colors.

8.4.1.3 Data Collection

While recording sensor data, robots perform all 8 behaviors in a sequence on the 95 objects, in round-robin fashion, to minimize any transient noise effects after a single trial on an object. Both robots perform 5 such trials on each object, resulting in 7,600 interactions, overall.

8.4.1.4 Feature Extraction

We used all interactive * behaviors in our experiments (i.e., all behaviors listed above except *look*). We used audio, effort at the robot's joints, and force at the robot's end-effector in our experiments, as they play crucial roles in the human somatosensory system for recognizing non-visual object properties. For audio, we used librosa [MRL+15] to generate mel-scaled spectrograms of the audio wave files recorded by robots with FFT window length of 1024, hop length of 512, and 60 mel-bands. Then, a spectro-temporal histogram was computed by discretizing both time and frequency into 10 equally-spaced bins, where each bin consists of the mean of values in that bin. Effort and force data were discretized into 10 equally-spaced temporal bins for joints and axes, respectively. Thus, audio and force data are represented as 100 and 30 dimensional feature vectors, respectively. For *Baxter* and *UR5*, *effort* data is represented as 70 and 60 dimensional feature vectors, respectively. Fig. 8.2

*Experiments with *look* histogram features were performed, but no improvements were observed, indicating that vision alone is insufficient.

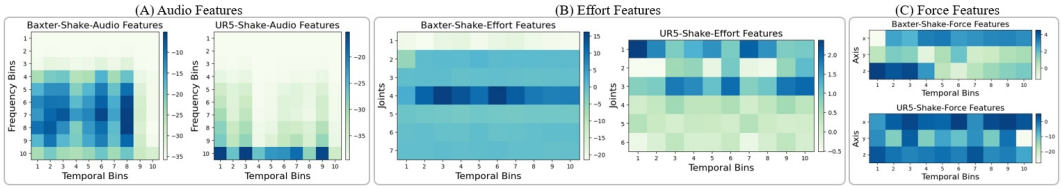


Figure 8.2: Examples of (A) *audio*, (B) *effort* and (C) *force* features when *Baxter* and *UR5* perform *shake* on a *blue-marbles-150g* object.

visualizes both robots' *audio*, *effort*, and *force* features when they perform *shake* behavior on a *blue-marbles-150g* object.

8.4.1.5 Data Augmentation

To improve model generalization, we increase the number of object trials through data augmentation: we compute each bin's mean and standard deviation in the discretized representation of all object trials and sample $k = 5$ additional trials of each object. [†] The rationale behind augmenting data by constraining on trials is to generate realistic data that is less likely to be impossible to produce in the real-world. Furthermore, this data augmentation technique is independent of the downstream task and can be applied for both object-property and object-identity recognition.

8.4.2 Evaluation

We evaluated performance of the projection methods: 1) EDN projects source robot features to a target robot's feature space, and 2) KEMA projects individual robot features to a shared feature space. To learn both projections, we evaluate two ways to build correspondence between source-target data pairs: 1) object identity-based pairs, wherein both source and target robots interact with the same object identity (e.g., *baxter-buttons-50g* and *ur5-buttons-50g*); and 2) object property-based pairs, wherein source and target robots interact with objects that share a property (e.g., *baxter-buttons-50g* and *ur5-dices-50g*, wherein weight is same and contents are different). In both correspondence types, we use the same behavior and modality for both source and target robots. We consider 2 tasks: object property-recognition and

[†]Experiments with $k = 10$ showed little additional improvement.

object identity-recognition. In property-recognition, the target robot must recognize content and weight of the object it interacts with; there are 7 content classes and 4 weight classes, including an empty class. In object identity recognition, the target robot must recognize the specific object identity.

8.4.2.1 Object Property Recognition Task

As a baseline condition, we train the target robot using data in its own feature space. For the transfer condition, we train the target robot using features obtained by applying the projections. In training each projection, we use all 95 objects for the source robot and increment the number of objects the target robot interacts with, from 4 (for weight-recognition) and 7 (content-recognition), to 76 objects (80% of objects). The remaining 19 objects (20% objects) are held-out for testing target robot performance. We randomly-sampled 76 objects for incremental training and used remaining 19 for testing; we repeated this process 10 times, in both conditions. For best target robot performance in the baseline condition, we train using all 95 objects and evaluate on test objects in each fold. In all cases, we use all 5 trials of each object.

8.4.2.2 Object Identity Recognition Task

The baseline and transfer conditions of the object identity recognition task are the same as in the property recognition task. We evaluated the target robot's performance to recognize 12 randomly-sampled objects from the 95 objects. When training each projection method, we used all 5 trials of each object for the source robot and increment the number of trial per object from 1 to 4 (80% trials) for the target robot. The remaining 1 trial (20% trials) of each object is held-out for testing the target robot's performance. For both conditions, we performed 5-fold cross-validation such that each trial of all 12 objects is included in the test set, once. For best target robot performance in the baseline condition, we train using all 5 trials of all 12 objects and evaluate on the test trial of each object for each fold. The process of selecting 12 objects, and performing 5-fold cross-validation for both

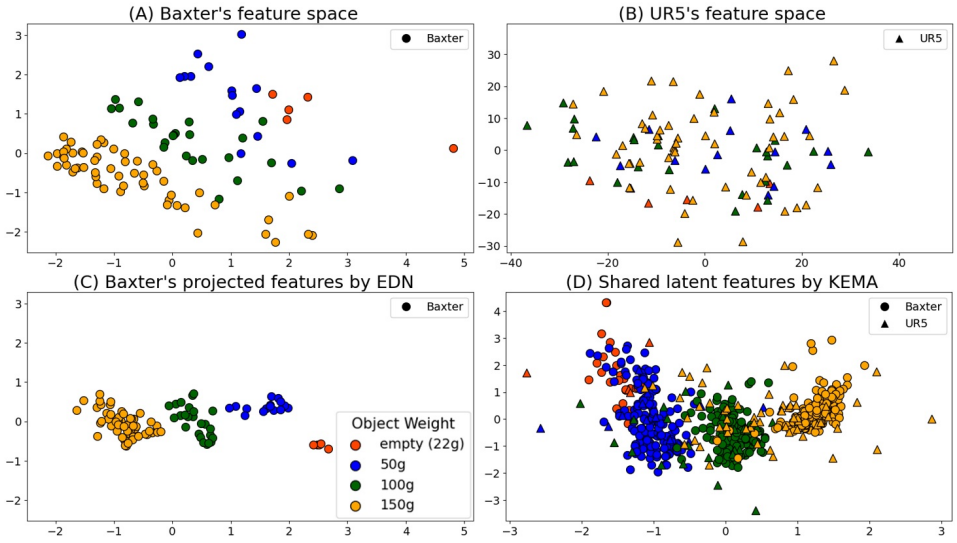


Figure 8.3: Original sensory features of (A) *Baxter* and (B) *UR5* for *pick-force* performed on 20 objects in 2D space, and (C) the projected features from *UR5-pick-force* to *Baxter-pick-force* projection using EDN, and (D) first 2 dimensions of corresponding features in the shared latent feature space generated using KEMA.

conditions is repeated 10 times to compute performance statistics.

8.4.2.3 Evaluation Metrics

We used two metrics to evaluate the target robot's recognition performance. First, we consider accuracy $A = \frac{\text{correct predictions}}{\text{total predictions}}\%$; the second metric is the accuracy delta (ΔA), which measures the drop in accuracy due to using projected features (obtained by interacting with *fewer* objects) versus using the target robot's own features (obtained by interacting with *all* objects). We compute mean accuracy delta of the least m number of object interactions in our experiments, defined as: $\text{m}\Delta A = \frac{1}{m}\sum_{j=1}^{m}(A_{all} - A^{j}_{projected})\%$,

where A_{all} is the accuracy obtained using 100% of the target robot's data, $A_{projected}$ is the accuracy obtained using projected features, and $m = 10$ for object property recognition, and $m = 4$ for object identity recognition. For both metrics, we use recognition accuracy computed as a weighted combination of all the behaviors and modalities used, based on their performance on the training data.

8.5 Results

8.5.1 Illustrative Example.

Consider the case where a source robot (*Baxter*) and a target robot (*UR5*) perform *pick* behavior and record force signal. *Baxter* interacts with all 95 objects, and *UR5* interacts with only 20 objects; both robots perform 5 trials on each object. We use Principal Component Analysis (PCA) to visualize the robots' feature spaces (Fig. 8.3A and 8.3B) and plot object weights with different colors. In Fig. 8.3A we only plot *Baxter's* features of the common 20 objects, for comparison to original and projected features shown in Fig. 8.3C.

We project *UR5-pick-force* to *Baxter-pick-force*, via EDN with object identity-based correspondences, and visualize with PCA in Fig. 8.3C. Compared to *Baxter*'s space (Fig. 8.3A), projected features are more tightly clustered for different weights. We also generate the shared latent features using KEMA with object identity-based correspondences. We plot first 2 dimensions of latent features in Fig. 8.3D: data collected by both robots of 4 different weights are clustered together, indicating both robots' data distribution is aligned efficiently.

Consider another case where *UR5* interacts with one object of each weight 5 times and learns to recognize the object's weight using 20 examples (4 weights $\times$ 5 trials). The mean accuracy computed over 10 folds using these 20 examples is 22.31 ± 8.05. This learning process is the same as in our baseline condition, where the robot learns using its own features. Now, we additionally use the 5 trials with data augmentation and train *UR5* to recognize the object's weight using 40 examples: 4 weights $\times$ (5 real trials + 5 augmented). The mean accuracy computed over 10 folds using these real and augmented data is 28.21 ± 6.09; the increased accuracy shows that using data augmentation improves recognition performance. Since, we consistently observed improvements from augmentation, we only report the performance of our baseline and transfer conditions using augmentation.

8.5.2 Object Property Recognition Results.

For the object property-recognition task, we evaluated both projection methods by building correspondences based on object-identity and object-property. For EDN, we built object-identity correspondences by mapping each source robot's object trial to all the target robot's trials of that object. We built object-property correspondences by mapping each source robot's object with a property of the recognition task to all the target robot's objects with that property. For example, for the weight recognition task, a *50g* object interacted by the source robot will be mapped to all the *50g* objects interacted by the target robot. For KEMA, we build the manifold alignment using all 95 objects of the source robot and incrementally vary the number of objects the target robot interacts with, for both object-identity and object-property correspondences.

Fig. 8.4 shows results of EDN and KEMA on the weight- and content-recognition tasks, where *Baxter* is the source robot and *UR5* is the target robot: all transfer conditions for both approaches perform better than the baseline condition when the target robot interacts with fewer objects. As the target robot interacts with more objects, KEMA still performs better than baseline condition, and EDN performs comparable to baseline condition. Overall, results indicate that the proposed knowledge transfer methods can boost target robot performance, notably when it has limited time to learn tasks and cannot interact with many objects. We also experimented with *UR5* as the source robot and *Baxter* as the target, and observed a similar performance boost with transfer.

Table 8.1 shows mean accuracy delta (mΔA) results of both methods and both correspondence types. Lower mΔA means better performance, i.e., closer to the case where the target robot is trained using its own data with all objects. For KEMA, object-identity correspondence yields better performance; for EDN, both correspondences perform comparably. These findings indicate that object-identity correspondence builds better alignment for projecting features into the shared latent space than object property correspondence, though the correspondences yield com-

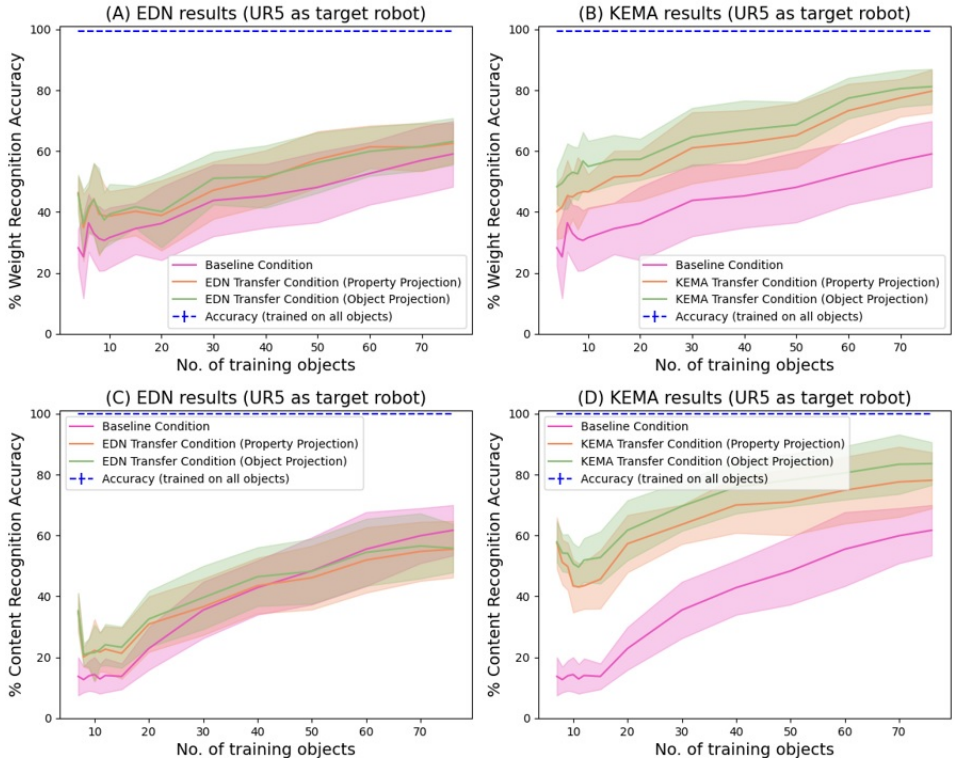


Figure 8.4: Accuracy results of the baseline and transfer conditions, EDN (**left**) and KEMA (**right**), on the weight (**top**) and content (**bottom**) recognition tasks, for *Baxter* (source) and *UR5* (target).

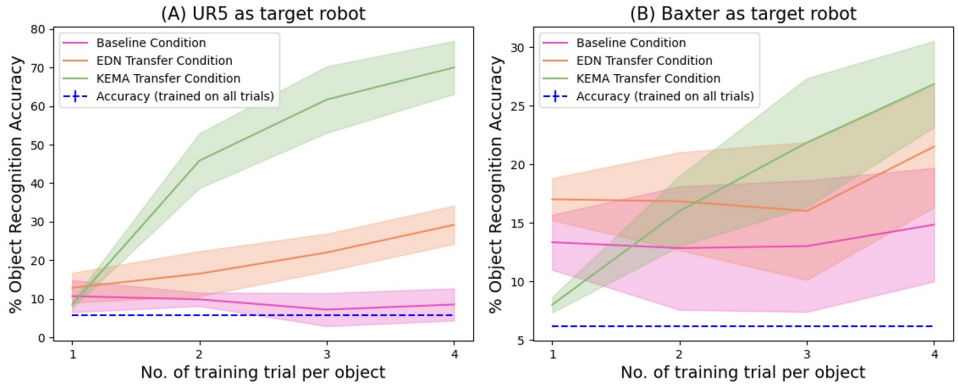


Figure 8.5: Accuracy results of the baseline and transfer conditions on the object identity recognition tasks.

parable performance for projecting features into the target feature space. KEMA outperforms EDN in all cases, showing KEMA as more efficient in transferring implicit object property knowledge across robots.

8.5.3 Object Identity Recognition Results.

For the object identity recognition task, we evaluated EDN and KEMA approaches by building correspondences based on object-identity. We build the object-identity correspondences in the same manner as for the object property recognition task.

Table 8.1: Mean accuracy delta (mΔA) results of EDN and KEMA for object identity-based and property-based correspondences.

	UR5		***Baxter***	
Method (correspondence)	Weight	Content	Weight	Content
EDN (object-identity)	57.72	71.26	31.91	32.88
EDN (object-property)	58.57	72.54	30.49	32.78
KEMA (object-identity)	**44.88**	**42.17**	**28.84**	**19.25**
KEMA (object-property)	51.88	47.53	34.88	22.60

Table 8.2: Mean accuracy delta (mΔA) results of EDN and KEMA on the object identity recognition tasks.

Method (correspondence)	***UR5***	***Baxter***
EDN (object-identity)	-14.29	-11.67
KEMA (object-identity)	**-40.67**	**-12.00**

We emphasize that identifying specific objects in our dataset is a challenging task. For example, if two objects have the same weight but different contents, it would be very crucial to listen to the audio signal produced while performing behaviors, as the force signal would not be helpful to distinguish those objects. Thus, we used 12 randomly-sampled objects with unique weight and content for the object identity recognition task.

Fig. 8.5 shows the accuracy results, and Table 8.2 shows the mean accuracy delta results of both approaches on object-identity correspondence. Overall, both approaches perform better than the baseline condition, and KEMA performs significantly better than EDN. These results indicate that features in the shared latent space contain more helpful information for identifying specific objects than

the target feature space. Negative values in Table 8.2 show that using projected features for training the target robot leads to better performance than using 100% of the target robot's own features. These results indicate that using projected features from the source robot helps the target robot to learn a recognition model that generalizes better for object identity recognition. In addition, our baseline condition also performs better than using 100% of the target robot's own features, indicating that the data augmentation technique we applied improves the generalization of the recognition models.

8.6 Summary

For a robot to learn about implicit object properties, it must perform object exploration while processing various non-visual modalities. This process is costly across multiple robots as object feature representations are unique to a robot's morphology. We proposed a framework for transferring implicit object property knowledge across heterogeneous robots and evaluated two projection methods, on two interactive perception tasks; results showed that learning on a target robot is accelerated through transfer from source robot, even if it explores fewer objects. Although our framework expedites learning on the less experienced target robot, there are some limitations. We encoded different behaviors in robots for object exploration.

In future work, we plan to enable robots to learn behaviors to extract different object properties, autonomously. Moreover, we assumed that both source and target robots explored objects using the same sensorimotor context; thus, we used this same context while learning the projections. We plan to select sensorimotor contexts for learning projections more efficiently. Furthermore, we plan to automate the selection of objects to be explored, to learn the projection faster. Finally, we envision a scenario where more than two robots explore objects with additional properties, e.g., shape, size, material, and stiffness.

Chapter 9

Cross-Tool and Cross-Behavior Perceptual Knowledge Transfer for Grounded Object Recognition*

9.1 Introduction

Humans employ specialized tools to acquire knowledge about objects' properties and develop a comprehensive understanding of their physical characteristics, such as size, shape, texture, weight, and durability. For example, kitchen utensils (e.g., knives, spoons) can be employed to examine the properties of food, including its texture and consistency. Robots are expected to operate effectively in human environments; thus, the ability to estimate the physical properties of objects has become an essential component of robotics research. Recent studies demonstrated robots can effectively use tools to interact with objects and learn about various properties, including material composition, shape, hardness, elasticity, brittleness, and adhesiveness [SLZ+21, GS14, LKS15, HWL+20, BLSS19, GBKS19, TSL+21, KR23, ZAS+23, LBDC23].

*__This chapter is based on the following paper:__ Gyan Tatiya and Jonathan Francis and Jivko Sinapov, "Cross-Tool and Cross-Behavior Perceptual Knowledge Transfer for Grounded Object Recognition", *Under review for IEEE International Conference on Robotics and Automation (ICRA)*, 2024. [TFS24]

Robots can use tools to execute actions on objects and observe their effects via various sensors, including visual, audio, and haptics, to acquire knowledge of objects' properties. Non-visual modalities, such as audio and haptics, are essential, as vision alone cannot provide information about an object's intrinsic properties, including its weight, temperature, or hardness. One of the challenges when representing non-visual modalities is that data collection requires significant time for this interactive object exploration, which may delay downstream tasks [MFHH22, LKS+20, PGGG+20, LBDC21, WWW+22, WCH+21, LLC22]. A logical solution for efficient learning would be to transfer object property representation to a new robot. However, if the new robot possesses different interaction capabilities, such as new behaviors or tools, the implicit knowledge obtained by the previous robot cannot be directly transferred to the new one [FKL+22]. A robot's multisensory model for interactive perception tasks is unique to its sensors, behaviors, and tools. Therefore, transferring knowledge of non-visual object properties across different sensorimotor contexts is challenging, and each robot must learn its task-specific sensory models from scratch.

To overcome this challenge of transferring implicit knowledge of non-visual object properties, we propose a framework leveraging triplet loss as our primary method to share tool-mediated behavioral knowledge across sensorimotor contexts, i.e., a tool-behavior pair. Our method aims to learn a shared latent feature space by utilizing the implicit knowledge of the source robot with more experience and transferring it to the target robot with less experience. The target robot can use the learned feature space to learn to recognize novel objects it has not previously interacted with, given the source robot has explored them. To evaluate our method, we collected a dataset using a UR5 robot that used 6 tools to perform 5 behaviors on 15 granular objects. We tested our method on two tasks: cross-tool transfer and cross-behavioral transfer. Our results demonstrate the less-experienced target robot can bootstrap its object property learning by leveraging the source robot's experience. Our method enables the target robot to recognize novel granular objects it has not interacted with before test time, thus improving its learning process'

efficiency and accuracy.

In the context of the broader dissertation, this chapter introduces a framework for transferring implicit knowledge of non-visual object properties, specifically focusing on tool-mediated behavioral knowledge, across diverse sensorimotor contexts. Leveraging triplet loss as the primary method for contrastive training, our approach aims to *Transfer using Projection to Shared Latent Feature Space*, enabling knowledge transfer from a source robot with more experience to a target robot with less experience. The evaluation utilizes a dataset collected by the UR5 robot, as described in Chapter 3. Unlike previous chapters that focused on direct object exploration, this chapter investigates perceptual knowledge transfer for tool-mediated object exploration tasks. Additionally, we employ the Kernel Manifold Alignment (KEMA)-based method proposed in Chapter 6 as a baseline for comparison. The evaluation encompasses two tasks: cross-tool transfer and cross-behavioral transfer. By demonstrating the less-experienced target robot's ability to bootstrap its object property learning using the source robot's experience, this chapter contributes to enhancing the efficiency and accuracy of the learning process in robotic systems.

9.2 Related Work

Psychological research demonstrated that children begin to comprehend how objects can be used as tools to develop intuition about the physical world at an early age [BG11], often using utensils like spoons and forks to investigate food characteristics, such as texture. Employing tools indicates intelligent adaptability: it necessitates an understanding of the properties of the tool and the object being acted upon [CD89]. By using tools to explore objects, infants can modify the properties of the object being acted upon, enabling them to learn by observing the effects of their actions [Loc00].

Robotics research demonstrated robots can likewise use tools to explore objects and learn about their physical properties. Gemici *et al.* [GS14] developed a method to manipulate deformable food items (e.g., bread, tofu) using kitchen tools

(e.g., knife, spatula), and infer their physical properties (e.g., elasticity, adhesiveness, and hardness); their PR2 robot executed cutting and splitting actions on food items and used haptic data (e.g., force and tactile) to learn about food properties by monitoring changes in the food due to actions. Sawhney *et al.* [SLZ+21] used multimodal data (e.g., audio, force) to classify food materials by interacting with them using tools. Sundaresan *et al.* [SBS22] deployed a multimodal policy on a Franka robot that leveraged visual and haptic observations during interaction with deformable food items to plan skewering motions rapidly and reactively. One challenge faced by these approaches is that implicit knowledge gained by a robot via object interaction cannot be directly used by another robot, as each robot's unique sensorimotor context significantly affects the sensed data distribution and the resultant model that each robot learns. These works focused on learning from scratch, for each robot's sensorimotor context, which is expensive at scale for robots operating under heterogeneous contexts. We propose a framework for transferring implicit knowledge acquired during object exploration using tools, from a source robot to a target robot, which *differ* in their sensorimotor contexts.

Recent studies transfer implicit knowledge across sensorimotor contexts in interactive object perception, yet they did not use tools and were limited to rigid objects [THCHS19, TSES20, TFS23]. In [THCHS19], an encoder-decoder network was used to generate a "target" robot's features from a "source" robot's learned representation for object categorization. This study only considered exploring rigid objects without tools, however, hence challenges associated with granular objects explored with tools remained unaddressed. In [TFS23], a distribution alignment-based approach was used to project features from two heterogeneous robots with different embodiments into a shared latent space for non-visual object property recognition. Whereas they demonstrated a shared latent space to be more effective for transfer, compared to learning projection functions to generate target context features, this study was also limited to exploring rigid objects without tools. Moreover, they assumed heterogeneous robots had access to the same behavior in their sensorimotor context and thus learned the shared latent space for the same behavior across dif-

ferent robots. To overcome these limitations, we collected a multisensory dataset using a UR5 robotic arm that performed 4,500 interactions to explore 15 granular food-like materials using 6 tools and 5 behaviors, developed a projection method for implicit knowledge transfer across two heterogeneous sensorimotor contexts, and evaluated our approach on cross-tool and cross-behavioral transfer tasks.

9.3 Learning Methodology

9.3.1 Notation and Problem Formulation

Consider two robots, source and target, that explore a set of granular food-like objects $\mathcal{O}$ (e.g., *salt*, *wheat*), kept in containers, by using a set of tools $\mathcal{T}$ (e.g., *spoon*, *fork*) and performing a set of exploratory behaviors $\mathcal{B}$ (e.g., *stirring*, *twist*), while recording a set of non-visual sensory modalities $\mathcal{M}$ (e.g., *audio*, *effort*). Let the robots use each tool to perform each behavior n times on each object. Let $\mathcal{C}$ be the set of exploratory contexts, including each possible combination of a tool in $\mathcal{T}$, a behavior in $\mathcal{B}$, and a sensory modality in $\mathcal{M}$, e.g., *spoon-stirring-audio*, *fork-twist-effort*. For the i^{th} exploratory trial, the robot's observation feature is $x_i^c \in \mathbb{R}^{D_c}$, where $i \in \{1, ..., n\}$, $c \in \mathcal{C}$, and D_c is the dimension of the robot's feature space under context c.

Let $c_s, c_t \in \mathcal{C}$ be the sensorimotor contexts of the source and target robots, respectively, which differ either by tool or behavior, e.g., for different tools, *spoon-stirring* as c_s and *fork-stirring* as c_t, and for different behaviors, *spoon-stirring* as c_s and *spoon-twist* as c_t; the sensory modality remains the same for both c_s and c_t contexts. Consider the case where the source robot explored all objects in $\mathcal{O}$ under context c_s; however the target robot under context c_t only explored a subset of the objects $\mathcal{O}_{shared} \subset \mathcal{O}$, and needs to learn an object recognition model for the remaining set of novel objects $\mathcal{O}_{novel} \subset \mathcal{O}$, with $\mathcal{O}_{shared} \bigcap \mathcal{O}_{novel} = \emptyset$. Our goal is to learn a projection function using $\mathcal{O}_{shared}$, to transfer knowledge about novel objects $\mathcal{O}_{novel}$ from the more-experienced source robot to the less-experienced target robot. This knowledge transfer will help the target robot to learn about novel

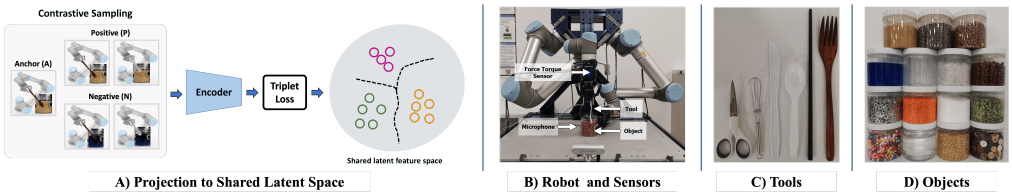


Figure 9.1: (A) Projection from source and target feature spaces into a shared latent space using Triplet Loss. (B) Experimental platform and sensors of the *UR5* robot. (C) The 6 tools used in this study: *metal-scissor*, *metal-whisk*, *plastic-knife*, *plastic-spoon*, *wooden-chopstick*, and *wooden-fork* (left to right). (D) The 15 objects used in this study (row-wise, left to right): *cane-sugar*, *chia-seed*, *chickpea*, *detergent*, *empty*, *glass-bead*, *kidney-bean*, *metal-nut-bolt*, *plastic-bead*, *salt*, *split-green-pea*, *styrofoam-bead*, *water*, *wheat*, and *wooden-button*.

objects without prior interaction with them.

For transferring object knowledge, we consider a projection function $F_{c\to\mathcal{Z}}$, that projects the observation features from source and target contexts' feature spaces to a shared latent feature space, such that the robots can be trained to recognize objects in that latent space, as opposed to each robot's own feature space. More specifically, $F_{c_s\to\mathcal{Z}} : x_i^{c_s} \to z_i^{c_\mathcal{Z}}$ and $F_{c_t\to\mathcal{Z}} : x_i^{c_t} \to z_i^{c_\mathcal{Z}}$, where $z_i^{c_\mathcal{Z}} \in \mathbb{R}^{D_\mathcal{Z}}$ and represents the shared latent features of size $D_\mathcal{Z}$. This will enable the robots to use the observation features collected under both contexts to learn an object recognition model and perform better than a model trained only using a specific context's observation features. Learning a shared latent feature space would enable the target robot to recognize novel objects, given the source robot has explored those objects.

9.3.2 Knowledge Transfer Model

To learn the projection function $F_{c\to\mathcal{Z}}$, we employ Triplet Loss (TL) [BRPM16], which guides our neural projection to map sensory data from both source and target contexts (c_s, c_t) into a common latent space (Fig. 9.1A). The essence of triplet loss is to ensure that embeddings of examples belonging to the same object class are closer in the latent space than those of dissimilar examples from different object classes:

$$\mathcal{L}(A, P, N) = \min(0, d(A, P) - d(A, N) + \alpha), \tag{9.1}$$

for anchor example A, positive example P, negative example N, and a margin hyperparameter α that defines a minimum difference that must be maintained between the distance from the anchor to the positive sample and the distance from the anchor to the negative sample (we set $\alpha = 1$). The function $d(x, y)$ calculates the distance between examples x and y using the Euclidean distance formula, $d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$, where x and y are the two examples being compared, and n is the dimensionality.

To train our projection function using triplet loss, we construct a dataset of triplets (A, P, N) as follows: For each object $o \in \mathcal{O}$, we designate the anchor (A) as data from the source context (c_s) for object o. The positive (P) is either from the same source context (c_s) but a different trial or from the target context (c_t) for the same object. The negative (N) is either from the same source context (c_s) or from the target context (c_t) for a different object. We randomly sample a single example for both positive and negative cases when multiple examples are available from source and target contexts. This triplet dataset is created using all trials of objects, and we optimize the triplet loss function over it. By doing so, our network learns to map sensory data from both source and target contexts into a shared latent space $(\mathcal{Z})$. In this latent space, objects of the same class are brought closer together than objects of different classes. Consequently, when the target robot encounters novel objects $(\mathcal{O}_{shared})$, it can effectively recognize them by comparing their embeddings in the shared latent space $(\mathcal{Z})$, even if it has not directly interacted with them during exploration. This process ensures that the robot builds a robust representation of objects, capable of generalizing across diverse contexts and effectively recognizing novel objects.

9.3.3 Model Implementation

The knowledge transfer model is constructed as a Multi-Layer Perceptron (MLP) comprising three hidden layers with 1000, 500, and 250 units, employing the Rectified Linear Unit (ReLU) activation function. This model projects sensory data into a shared latent vector of dimension $D_{\mathcal{Z}} = 125$. To enable the target robot to

recognize novel objects, we use shared latent features corresponding to the novel objects in the target context projected by the source context. These shared latent features are comparable and can be employed to train a standard multi-class classifier across different contexts. We train an MLP model with a single hidden layer of 100 units for the recognition task, allowing the target robot to discern objects it has not directly encountered. The knowledge transfer and classification models are updated for 500 training epochs, leveraging the Adam optimization algorithm [KB15] with a learning rate set at 10^{-4}. We used PyTorch [PGM+19] for model implementation.

9.4 Evaluation Design

9.4.1 Experimental Platform and Feature Extraction

9.4.1.1 Robot and Sensors

We collected a dataset using the *UR5* robot with a 6-DOF and a 2-finger Robotiq 85 gripper (shown in Fig. 9.1B). The *UR5* had a Seeed Studio ReSpeaker microphone placed on its workstation, and a force sensor measuring effort at each joint, and a force-torque sensor at the end-effector. We recorded audio data at a sampling rate of 16 kHz, effort data at 135 Hz, and force data at 125 Hz.

9.4.1.2 Tools, Exploratory Behaviors and Objects

The robot used 6 tools *metal-scissor*, *metal-whisk*, *plastic-knife*, *plastic-spoon*, *wooden-chopstick*, and *wooden-fork* (Fig. 9.1C) to perform 5 interactive behaviors: *stirring-slow*, *stirring-fast*, *stirring-twist*, *whisk*, and *poke* (Fig. 9.2). We chose these specific tools and behaviors because they capture different aspects of objects' properties. The interactive behaviors are encoded as robot joint-angle trajectories. The robot explored 15 objects: *cane-sugar*, *chia-seed*, *chickpea*, *detergent*, *empty*, *glass-bead*, *kidney-bean*, *metal-nut-bolt*, *plastic-bead*, *salt*, *split-green-pea*, *styrofoam-bead*, *water*, *wheat*, and *wooden-button* (Fig. 9.1D) kept in cylindrical containers.

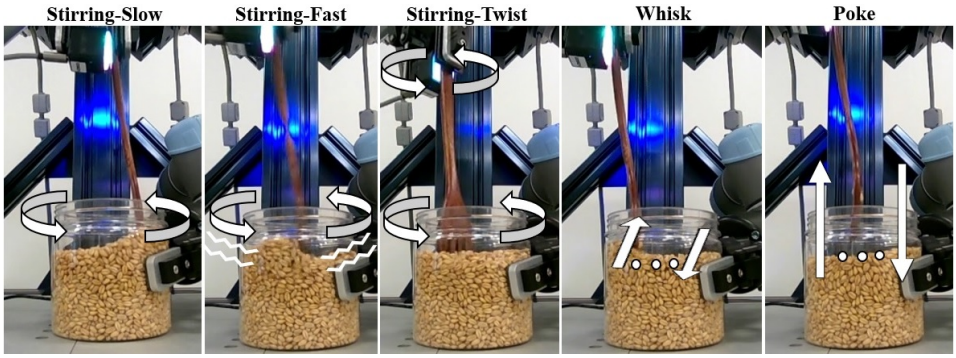


Figure 9.2: The 5 behaviors used to explore objects: *stirring-slow*, *stirring-fast*, *stirring-twist*, *whisk*, and *poke* (left to right).

9.4.1.3 Data Collection

While recording sensory data, the robot performed all 5 behaviors in a sequence on an object using a tool. Once an object was explored using a tool, the same object was not explored again until all the objects were explored using that tool to limit any transient noise effects after a trial on an object. We used another *UR5* arm only to hold the containers (Fig. 9.1B). The robot performed 10 trials on each object using a tool, resulting in 4,500 interactions (6 tools x 5 behaviors x 15 objects x 10 trials). Datasets download link, source code, and complete results are available on the GitHub page of the study [*]. Please reference our dataset available for download.

9.4.1.4 Feature Extraction and Data Augmentation

We used the 6 tools and 5 interactive behaviors listed above to conduct our experiments. We used 3 non-visual modalities (i.e., audio, effort, and force) because they are essential for the human somatosensory perception of object properties. The feature extraction parameters and data augmentation routines were adopted from [TFS23]. To represent audio data, first, we used librosa [MRL+15] to generate mel-scaled spectrograms of audio wave files recorded by robots with FFT window length 1024, hop length 512, and 60 mel-bands. Secondly, a spectro-temporal histogram was computed by discretizing both time and frequencies into 10 equally-spaced bins, where each bin consisted of mean of values in that bin. Similarly, we discretized time into 10 equally-spaced bins for effort and force data, for 6 joints and 3 axes,

[*] https://github.com/gtatiya/Tool-Knowledge-Transfer

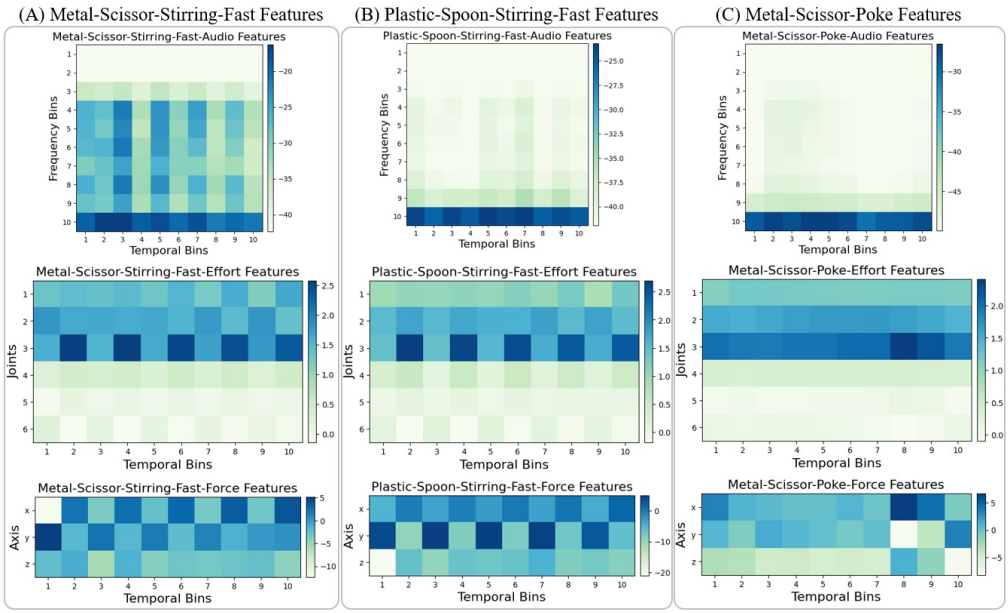


Figure 9.3: Examples of *audio*, *effort*, and *force* features (top to bottom) when *UR5* uses *metal-scissor* tool to perform *stirring-fast* (A) and *poke* (C) behaviors, and uses *plastic-spoon* tool to perform *stirring-fast* behavior (B) on a *metal-nut-bolt* object. Please note the difference in the features when only the tools are different ((A) and (B)) and when only the behaviors are different ((A) and (C)).

respectively. Thus, audio, effort, and force data are represented as 100, 60, and 30 dimensional feature vectors, respectively. Fig. 9.3 visualizes the robot's *audio*, *effort*, and *force* features when it uses different tools to perform different behaviors on an object. To augment data, we computed the mean and standard deviation of each bin in the discretized representation of all trials of an object, and sampled 10 additional trials of each object. These augmented data were used to train all methods.

9.4.2 Evaluation

9.4.2.1 Transfer and Baseline Conditions

For the transfer condition, we assume the source robot interacts with all 15 objects in $\mathcal{O}$ under the source context, but the target robot interacts with only 10 randomly selected objects (66.67% of objects) in $\mathcal{O}_{shared}$ under the target context. The 10 shared objects under both contexts are used to train the knowledge transfer model that projects the sensory signal of both contexts into a shared latent space. Sub-

sequently, an object classifier is trained using the projected data from the source context corresponding to the 5 objects (33.33% of objects) in $\mathcal{O}_{novel}$ that are novel to the target context. We used the latent features corresponding to the 5 novel objects under the target context generated by the trained knowledge transfer model to test the object classifier. We used two baseline conditions. For baseline 1, the target robot is trained to recognize objects using its own data collected during object interactions under the target context. This baseline would show the target robot's performance if it actually explored all the objects under the target context during the training phase. Baseline 2 is similar to baseline 1, except the target robot is trained under the source context. The target robot's own data observed under the target context are used to test both baseline conditions. Baseline 2 is zero-shot classification, as the target robot is trained under the source context and tested under the target context. In each condition, the classifiers were trained on randomly sampled 8 trials (80% of trials) from each of the 5 novel objects and tested on the held-out 2 trials (20% of trials). The process of randomly selecting 10 objects in $\mathcal{O}_{shared}$ to train knowledge transfer mode, training and testing the object classifiers on 5 novel objects for the transfer and baseline conditions, was repeated 10 times to compute performance statistics. We used a dataset with one robot; hence, we assumed the source and target robots are physically identical, although employing different tools and behaviors during object interaction. Nonetheless, our proposed transfer learning methodology remains pertinent in scenarios where the two robots are not physically identical.

9.4.2.2 Evaluation Metrics

We used two metrics to evaluate the object recognition performance of the target robot on the objects it has not explored. First is accuracy, defined as $A = \frac{\text{correct predictions}}{\text{total predictions}}$ (%). The second metric is accuracy delta ($A\Delta$), which measures the difference in classification accuracy by using the latent features for training instead of the ground-truth features. We define accuracy delta as $A\Delta = A_{truth} - A_{latent}$, where A_{truth} and A_{latent} are the accuracies obtained when using ground-truth and

latent features, respectively. A smaller accuracy delta indicates it is easy for the target robot to learn about the novel objects using the knowledge transferred by the source robot. To report both metrics' results, we use the recognition accuracy computed by performing a weighted combination of each modality used based on their performance on the training data.

9.4.2.3 Transfer Tasks

We consider two tasks: cross-tool sensorimotor transfer and cross-behavioral sensorimotor transfer. In cross-tool sensorimotor transfer, the source and target robots' contexts differ only by tools (e.g., *scissor-stirring* as the source context and *spoon-stirring* as the target context). In cross-behavioral sensorimotor transfer, the source and target robots' contexts differ only by behaviors (e.g., *scissor-stirring* as the source context and *scissor-poke* as the target context). In both tasks, we align the same modality for both source and target robots into the shared latent space.

9.4.2.4 Baseline Transfer Method

We use Kernel Manifold Alignment (KEMA) [TCV16] as our baseline. KEMA is a distribution alignment method to align observation features from various contexts and represent them within a shared latent space. KEMA constructs domain-specific projection functions, which project data from both source and target contexts into a shared latent space. This projection ensures that examples of the same object class are closely grouped while those from different classes are separated. KEMA has demonstrated effectiveness in various domains, including visual object recognition [TCV16], facial expression recognition [TCV16], and human action recognition [LLL+18]. In robotics, KEMA has been successfully employed to align haptic data [TSES20] and audio data [TFS23] across heterogeneous robots.

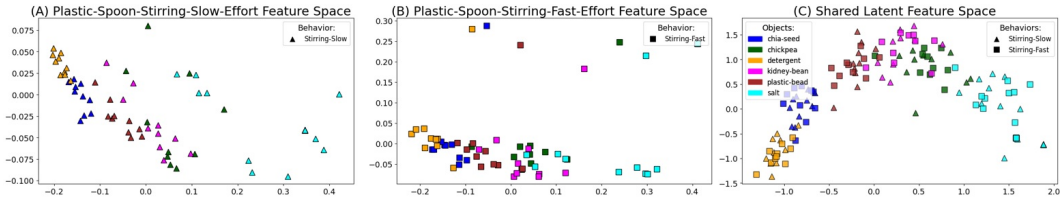


Figure 9.4: Original sensory features of (A) *plastic-spoon-stirring-slow* and (B) *plastic-spoon-stirring-fast* for *effort* performed on 6 objects in 2D space, and first 2 dimensions of corresponding features in the shared latent feature space (C).

9.5 Results

9.5.1 Illustrative Example

Consider the case where *UR5* uses the *plastic-spoon* to perform the *stirring-slow* behavior as the source context and the *stirring-fast* behavior with the same tool as the target context, on 15 objects, 10 times, while recording *effort* signals, which our knowledge transfer model uses to generate the shared latent features. Fig. 9.4 visualizes the original and latent features of 6 objects in 2D space. To visualize the original sensory signal of source and target contexts, we reduced their dimension to 2 by Principal Component Analysis (PCA) and plotted in Fig. 9.4A and 9.4B. Similarity, we plot the features in 2D for visualization (Fig. 9.4C): datapoints collected in both contexts are clustered together in the shared latent space, indicating both feature spaces are aligned effectively.

Consider another case where *UR5* learns to recognize the 15 objects using each tool and each behavior. To achieve this, we train a classifier for each tool and behavior pair using 8 trials of each object and test it on the held-out 2 trials. We perform 5-fold cross-validation such that each trial of the 15 objects is included in the test set once and compute the mean accuracy of all folds. Table 9.1 shows the recognition accuracy computed by performing a weighted combination of all 3 non-visual modalities used based on their performance on the training data. In the table, the bottom row shows the accuracy computed by performing a weighted combination of all the behaviors. We report these recognition accuracies to illustrate how each tool and behavior pair performs.

Table 9.1: Accuracy percentage (%) achieved by *UR5* using each tool and behavior pair to recognize 15 objects (↑).

	Tools					
Behaviors	***Metal-Scissor***	***Metal-Whisk***	***Plastic-Knife***	***Plastic-Spoon***	***Wooden-Chopstick***	***Wooden-Fork***
Stirring-Slow	26.00	33.33	18.67	36.67	16.00	42.00
Stirring-Fast	48.67	35.33	30.00	44.67	18.00	42.00
Stirring-Twist	17.33	26.00	16.00	23.33	15.33	32.67
Whisk	20.00	24.00	27.33	28.00	14.67	39.33
Poke	20.00	15.33	21.33	24.67	17.33	30.00
All behaviors	51.33	50.00	48.00	52.00	39.33	63.33

9.5.2 Accuracy Results of Object Recognition

For cross-tool sensorimotor transfer, each of the 6 tools is projected to all the other 5 tools, for each *behavior*, allowing 150 cross-tool projections (6 tools $\times$ 5 other tools $\times$ 5 behaviors). For cross-behavioral sensorimotor transfer, each of the 5 behaviors is projected to all the other 4 behaviors, for each *tool*, allowing 120 cross-behavioral projections (5 behaviors $\times$ 4 other behaviors $\times$ 6 tools).

Table 9.2 shows the mean accuracy and $A\Delta$ values for all projections, considering both transfer methods (TL and KEMA) and both baseline conditions. Our transfer method (TL) achieves higher accuracy than the baseline condition 1 in 74 and 8 projections across all cross-tool and cross-behavioral projections, respectively. In comparison to KEMA, our method achieves a lower mean $A\Delta$ in both baseline conditions. These results show using latent features transferred by the source robot using our method aids the target robot in learning a recognition model that generalizes better for object recognition under specific projections (discussed in the next section). Notably, a smaller mean $A\Delta$ (including negative mean $A\Delta$) in Table 9.2 indicates it is easy for the target robot to learn a classifier from latent features projected by the source robot and achieve comparable performance as if the target robot actually explored the objects.

We conducted additional experiments to assess the robustness and adaptability of our method. First, we repeated the experiments without data augmentation, to simulate a scenario that can resemble cases with limited data availability (shown

Table 9.2: Mean accuracy (↑) and $A\Delta$ (↓) for transfer and both baseline conditions in cross-tool and cross-behavior transfers. The experiments were conducted using discretized representations, with the inclusion of data augmentation, and an MLP classifier.

	KEMA		***TL (ours)***	
	Cross-Tool	***Cross-Behavior***	***Cross-Tool***	***Cross-Behavior***
Baseline 1 Mean Accuracy	50.6±12.5%	50.7±12.3%	50.6±12.3%	50.3±12.4%
Baseline 2 Mean Accuracy	26.5±6.0%	23.9±4.7%	26.0±5.8%	24.1±4.9%
Transfer Mean Accuracy	22.3±4.3%	22.1±4.9%	49.9±10.6%	33.7±8.6%
Baseline 1 Mean $A\Delta$	28.3±13.8%	28.6±14.6%	**0.7±13.9%**	**16.5±12.9%**
Baseline 2 Mean $A\Delta$	4.2±8.2%	1.7±7.3%	**-23.8±11.5%**	**-9.5±7.9%**

Table 9.3: Mean accuracy (↑) and $A\Delta$ (↓) for transfer and both baseline conditions in cross-tool and cross-behavior transfers. The experiments were conducted using discretized representations, without the inclusion of data augmentation, and an MLP classifier.

	KEMA		***TL (ours)***	
	Cross-Tool	***Cross-Behavior***	***Cross-Tool***	***Cross-Behavior***
Baseline 1 Mean Accuracy	51.6±12.6%	51.6±12.8%	51.5±11.5%	51.0±12.2%
Baseline 2 Mean Accuracy	26.2±5.6%	23.7±4.3%	26.1±5.3%	23.7±4.5%
Transfer Mean Accuracy	22.1±4.1%	20.9±4.3%	49.5±9.8%	35.1±8.2%
Baseline 1 Mean $A\Delta$	29.5±13.9%	30.6±13.5%	**1.9±12.0%**	**15.8±11.6%**
Baseline 2 Mean $A\Delta$	4.1±7.7%	2.7±5.6%	**-23.4±9.7%**	**-11.3±7.1%**

in Table 9.3). In this context, when using KEMA, the mean $A\Delta$ for baseline condition 1 averaged 29.5±13.9% and 30.6±13.5% for all cross-tool and cross-behavioral projections, respectively. In contrast, our method (TL) achieved substantially lower mean $A\Delta$ values of 1.9±12.0% and 15.8±11.6% for the same projections. Remarkably, even without data augmentation, our method consistently outperforms KEMA for both baseline conditions. Furthermore, we explored the impact of using a simple SVM classifier, as an alternative to an MLP, in the same experiments, both with and without data augmentation. Regardless of the classifier used, our method consistently achieved lower mean $A\Delta$ values for baseline conditions 1 and 2 compared to KEMA. These results underscore the robustness of our approach across varying data availability scenarios and with different classification models, demonstrating its ability to learn effective latent features that significantly aid the target robot in recognizing novel objects under diverse conditions.

9.5.2.1 Heterogeneous Feature Representation

The representation of a robot's sensory features can vary based on the chosen feature extraction method. To assess the adaptability of our framework to different feature

Table 9.4: Mean accuracy (↑) and $A\Delta$ (↓) for transfer and both baseline conditions in cross-tool and cross-behavior transfers. The experiments were conducted using learned representations obtained from autoencoders, with the inclusion of data augmentation, and an MLP classifier.

	KEMA		***TL (ours)***	
	Cross-Tool	***Cross-Behavior***	***Cross-Tool***	***Cross-Behavior***
Baseline 1 Mean Accuracy	64.4±16.6%	63.6±17.0%	63.6±16.7%	63.6±16.6%
Baseline 2 Mean Accuracy	19.4±3.5%	20.1±4.2%	19.6±3.6%	20.2±3.9%
Transfer Mean Accuracy	18.5±4.1%	17.5±4.7%	27.7±5.4%	27.1±6.1%
Baseline 1 Mean $A\Delta$	45.8±18.2%	46.1±18.1%	**35.8±15.1%**	**36.4±15.2%**
Baseline 2 Mean $A\Delta$	0.8±5.5%	2.6±5.8%	**-8.1±6.6%**	**-6.9±6.7%**

representations, we conducted additional experiments utilizing learned representations obtained through autoencoders, in contrast to the discretized representation used in previous experiments. In this set of experiments, we fixed the sensory data of each behavior by first calculating the average duration of each behavior. We determined the average time frames for each modality by multiplying this average duration with the modality-specific frame rate. This process is used to compute the fixed-sized raw sensory data by interpolation, ensuring a consistent and uniform representation across modalities for each trial of a behavior. Subsequently, we employed autoencoders to learn feature representations using the fixed-sized raw sensory data as input. The autoencoders were trained to reduce the dimensionality of the input into a low-dimensional code using an encoder and then reconstruct the input using the code as input through a decoder. For the autoencoders, fully connected layers were utilized in both the encoder and decoder. The code vectors obtained were then employed in knowledge transfer experiments, with an MLP as the classifier. The results, presented in Table 9.4, demonstrate the efficacy of the learned representations, with our method consistently achieving lower mean $A\Delta$ values compared to KEMA. This consistency across different feature representations underscores the versatility of our approach and its ability to facilitate knowledge transfer using Triplet Loss under varying representation schemes.

Table 9.5: Mean $A\Delta$ (baseline 1) for each behavior in cross-tool projections and for each tool in cross-behavioral projections (↓).

Cross-Tool		*Cross-Behavior*	
Behaviors	***Mean*** $A\Delta$	**Tools**	**Mean** $A\Delta$
Stirring-Slow	6.6±12.0%	Metal-Scissor	11.1±9.5%
Stirring-Fast	12.2±14.4%	Metal-Whisk	14.1±11.7%
Stirring-Twist	-6.6±12.8%	Plastic-Knife	16.2±8.2%
Whisk	-7.8±11.1%	Plastic-Spoon	25.6±12.5%
Poke	-1.4±5.3%	Wooden-Chopstick	5.6±9.7%
—	—	Wooden-Fork	26.6±11.0%

9.5.3 Accuracy Delta Results of Object Recognition

In cross-tool projections, for each tool as the source tool, we used all the other tools as the target tool, allowing 30 (6 tools × 5 other tools) projections for each behavior. In cross-behavioral projections, for each behavior as the source behavior, we used all the other behaviors as the target behavior, allowing 25 (5 behaviors × 4 other behaviors) projections for each tool. Table 9.5 shows the mean $A\Delta$ (baseline 1) of all projections for each behavior and each tool in cross-tool and cross-behavioral projections, respectively.

For cross-tool projections, the least mean $A\Delta$ is achieved by *whisk*, and *stirring-twist* behaviors. Compared to other behaviors, these behaviors deform the tools less during object interaction and are shorter behaviors. However, longer behaviors deform the tools more with object interaction and achieve higher mean $A\Delta$ (e.g., *poke*, *stirring-slow*, and *stirring-fast*). This shows if the robot needs to use a new tool, the prior experience gained by a shorter behavior that deforms the tools less would be better to be transferred to the target context with the new tool. For cross-behavioral projections, the least mean $A\Delta$ is achieved by *wooden-chopstick*, *metal-scissor*, and *metal-whisk* tools. Compared to other tools, these tools get deformed less while performing behaviors on objects and have pointed ends making limited object contact. However, other tools have wider ends, making them deform more with object interaction (e.g., *plastic-knife*, *plastic-spoon*, and *wooden-fork*). This shows that if the robot needs to perform a new behavior, the prior knowledge of behaviors gained using rigid and pointed tools would be better to be transferred to the target context's new behavior.

9.5.4 Tools and Behaviors Transfer Relationships

To compute the transfer relation between each tool and behavior pair, we consider cross-tool and cross-behavior projections simultaneously. For such projections, for each tool and behavior pair as the source context, we use all the other tool and behavior pairs as the target context. More specifically, we used 30 tool and behavior pairs (6 tools $\times$ 5 behaviors) as the source context and the other 35 pairs as the target context, allowing 870 (30 $\times$ 29) projections. We compute the $A\Delta$ for each projection and represent them in an 870 $\times$ 870 matrix, where the $A\Delta$ of identical contexts is 0. A 2D visualization of PCA embedding of the $A\Delta$ matrix is shown in Fig. 9.5. Each dot in the plot represents a context, and the distance between a pair of contexts indicates how efficient the transfer is between them. The closer the two contexts are, the more efficiently they transfer knowledge.

Contexts with the same or similar behaviors are clustered together, suggesting the source and target contexts with similar behaviors and different tools transfer better. The most tightly clustered behavior is *poke*. Similar behaviors are loosely clustered together (i.e., *stirring-fast*, *stirring-twist*, and *stirring-slow*). Non-deformable tools (i.e., *metal-scissor* and *wooden-chopstick*) with a behavior in the source context are closer to the other behaviors in the target contexts. This indicates that non-deformable tools capture similar object properties across different behaviors, as such tools are less impacted by different behaviors during object interaction. These findings are consistent with the cross-tool and cross-behavior transfers' results previously outlined.

9.6 Summary

Robots can acquire implicit knowledge about object properties by performing tool-mediated behaviors on granular objects and processing non-visual modalities. However, representing implicit knowledge for each different sensorimotor context can be expensive, as it necessitates the robots to explore objects from scratch in each new context. To overcome this challenge, we proposed a framework for transfer-

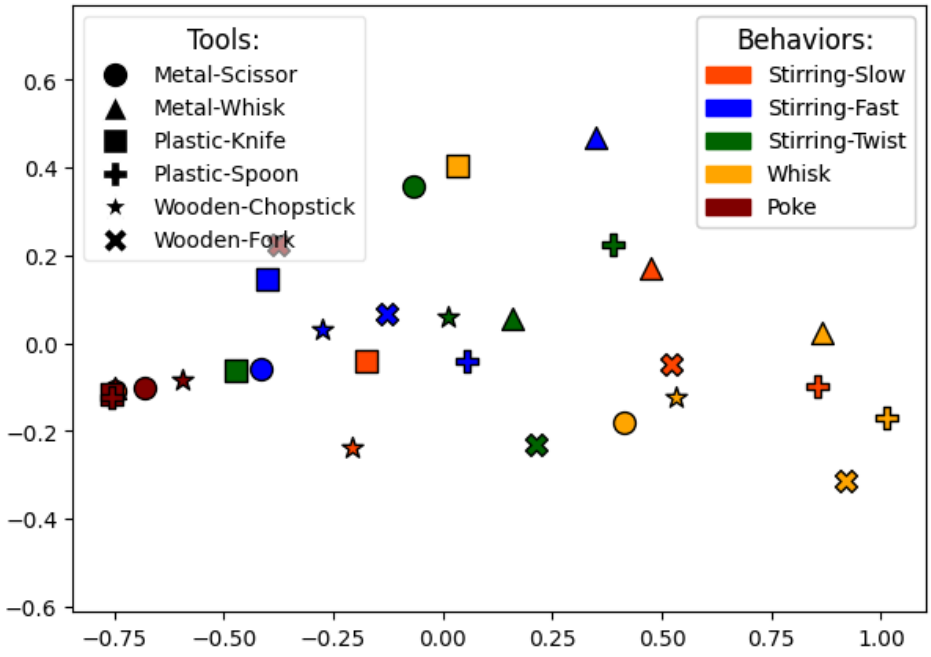


Figure 9.5: 2D PCA embedding of the $A\Delta$ matrix for cross-tool and cross-behavior projections. Every point stands for a context (i.e., a tool and behavior pair). Closer points reflect contexts across which knowledge transfer is more efficient.

ring implicit object property knowledge across different sensorimotor contexts. We evaluated the effectiveness of our approach on cross-tool and cross-behavioral transfer tasks. Our results demonstrated that transferring implicit knowledge from the source robot to the target robot accelerates the target robot's learning, even if it has explored fewer objects.

Our framework encoded different behaviors in the robot for object exploration using tools, but our future work aims to enable robots to learn behaviors for object interaction autonomously. We assumed that both source and target contexts explored objects using the same modality, and we used this modality while learning the shared latent space. We plan to perform cross-modality projections especially for cases where the target robot has a different non-visual sensor from the source robot and select sensorimotor contexts for learning projections more efficiently. We aim to automate the selection of objects to be explored to learn an effective projection faster. We envision multiple source contexts transferring their knowledge to the target context. In conclusion, our proposed framework can transfer implicit

knowledge about objects from one robot's sensorimotor context to another, leading to accelerated learning in the target context.

Chapter 10

MOSAIC: Learning Unified Multi-Sensory Object Property Representations for Robot Perception*

10.1 Introduction

Humans first acquire knowledge about object properties through physical interaction—a process that involves the integration of multiple sensory inputs, including visual, auditory, and tactile cues [TVCÖ04b, ANM10, BG06, KR23, ZAS+23, LKS+20]. For instance, we rely on vision to discern an object's color, sense of touch when we lift an object to gauge its weight, and hearing when we shake a container to determine if it is full or empty. The fusion of such multi-sensory information is pivotal in shaping our perception and guiding our decision-making processes concerning objects [BJT16, PSE12, FKL+22, CZCL23, CZBN21]. Similarly, robots can effectively engage with objects by simultaneously perceiving and processing multi-sensory signals, to tackle tasks such as object categorization [SSS+14a, TS19], material recognition [XLZ+22], and even complex actions like packing and pouring [LZZ+22].

*__This chapter is based on the following paper:__ Gyan Tatiya and Jonathan Francis and Ho-Hsiang Wu and Yonatan Bisk and Jivko Sinapov, "MOSAIC: Learning Unified Multi-Sensory Object Property Representations for Robot Perception", *Under review for IEEE International Conference on Robotics and Automation (ICRA)*, 2024. [TFW+24]

Within vision and text, large-scale Vision-Language Models (VLMs) have demonstrated their ability to provide state-of-the-art representations for both visual and textual modalities, making them exceptionally valuable for a wide range of AI applications [RKH+21, ZJM+22, ADL+22]. One such model, Contrastive Language-Image Pre-training (CLIP) [RKH+21], is trained from scratch on an extensive dataset comprising 400 million (image, text) pairs. CLIP's representations can seamlessly transfer to many downstream tasks without fine-tuning. While prior research has primarily focused on integrating audio modalities into CLIP's embedding space [WSKB22], including a robot's haptic data into this versatile space has yet to be explored. We address this gap by distilling the domain-general language grounding within CLIP and infusing it into a robot's sensory data from object interactions. This method effectively mitigates the often prohibitive costs of collecting interactive data by robots through extensive object exploration. The primary objective of this study is to expose VLMs to object property representations derived from robot interactions, highlighting how these representations can significantly improve the performance on interactive tasks by enhancing the robot's multimodal perceptual capabilities. This enhancement arises from interactive object exploration to understand the fundamentals of object properties, a perspective disembodied representations often lack.

We introduce MOSAIC (Multimodal Object property learning with Self-Attention and Integrated Comprehension), an approach to acquire versatile representations adaptable to various interactive perception tasks within robotics. MOSAIC is designed to extract unified multi-sensory object property representations, enabling understanding of object properties by leveraging diverse sensory modalities. This approach rests on the premise that natural language provides a versatile embedding space whose knowledge we can distill and align to different sensory modalities. We evaluate our approach on a publicly available dataset where a humanoid robot explored 100 objects, using 10 exploratory behaviors while recording sensory data, including vision, audio, and haptic. We evaluate on both object category recognition and the fetch object task, finding MOSAIC to be robust and adaptible. MOSAIC's

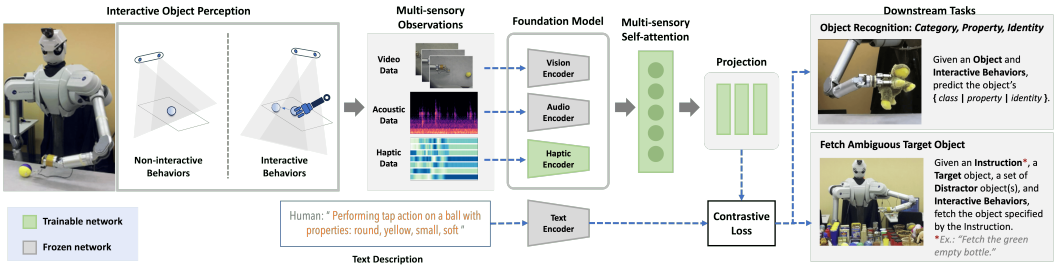


Figure 10.1: **Overview of the MOSAIC Framework**: Initially, the robot collects sensory data through object exploration, which is then used to train models for distilling unified multimodal representations guided by a pre-trained text encoder. These acquired representations are subsequently applied to a variety of downstream tasks.

performance in the object category recognition is notably competitive compared to state-of-the-art methods, showing the effectiveness of unified representations even within a straightforward linear probe setup. Furthermore, MOSAIC demonstrates exceptional capabilities in executing natural language instructions in the fetch object task under a zero-shot condition. In summary, MOSAIC offers a versatile framework for multimodal object property learning, bridging the gap between different sensory inputs and facilitating a wide range of downstream robot tasks.

In the context of the broader dissertation, this chapter presents MOSAIC, an approach aimed at acquiring versatile representations adaptable to various interactive perception tasks within robotics. MOSAIC extracts unified multi-sensory object property representations, leveraging diverse sensory modalities and distilling domain-general language grounding from Contrastive Language-Image Pre-training (CLIP). Unlike previous chapters that focused on specific transfer methods, MOSAIC is designed for learning *Transferable Unified Multi-sensory Object Property Representations*. Comparative analysis with specialized architectures proposed in Chapter 4 for raw multi-sensory data processing showcases the effectiveness of MOSAIC in learning unified representations for improved robot perception and downstream task execution.

10.2 Related Work

10.2.1 Multi-sensory Learning in Cognitive Science

Humans acquire knowledge about object properties through physical interactions, integrating multiple sensory signals [BDFP02, LS14, CT04]. Multi-sensory integration and attention processes occur at various stages in the human brain, crucially influencing our perception of objects and task performance [KBT10]. Moreover, human perception involves the dynamic interplay between sensory inputs and existing cognitive knowledge rather than processing sensory inputs in isolation [Tal15]. Our research extends these principles to robotics, extracting knowledge from pre-trained text encoders to align representations across diverse sensory modalities – mirroring how humans fuse sensory information with their established knowledge to perceive their environment holistically.

10.2.2 Robot Perception

Robotics research has showcased the remarkable capabilities of robots in interacting with objects and leveraging sensory signals for an array of tasks, encompassing object categorization [SSS$^+$14a, TS19], material recognition [XLZ$^+$22], and intricate manipulation actions like packing and pouring [LZZ$^+$22]. Most successful prior work relies on handcrafted auditory, haptic, and visual features [SSS$^+$14a], or specialized architectures for processing raw multi-sensory data to predict object categories [TS19]. Recently, Li *et al.* [LZZ$^+$22], introduced a self-attention model to fuse information from visual, auditory, and tactile sensors, significantly enhancing the robot's capability to tackle complex manipulation tasks. Our research introduces a versatile framework for learning unified multi-sensory representations *from raw sensory data* acquired during robot-object interactions, offering adaptability across diverse downstream tasks. The generality of our network architecture has been is demonstrated across various applications with strong performance, and the inclusion of self-attention mechanisms further bolsters its performance.

10.2.3 Unified Multi-Sensory Representations with Foundation Models

Recent advances have revealed the potential of contrastive objectives to yield generalized representations for both text and images [RKH+21, ZJM+22]. Contrastive Language-Image Pre-training (CLIP) [RKH+21] has delivered state-of-the-art representations that excel in diverse tasks, including zero-shot image classification, image retrieval via text, and guiding generative models [GPM+22]. While CLIP's knowledge has been distilled for audio [WSKB22], our MOSAIC approach is the first to ground sensory data obtained through robotic object exploration. MOSAIC accomplishes this by distilling knowledge from the extensive pre-trained CLIP text model. To test our learned unified representations, we rely on a dataset where a robot engages with 100 objects, executing 10 exploratory behaviors while recording multiple sensory signals. The robot tackles two tasks reliant on perceiving object properties: object categorization and the fetch object task. The results highlight the efficiency of our unified representations, clearly demonstrated in competitive performance in category recognition only by using a simple linear probe setup and in fetch object task using a zero-shot transfer approach.

10.3 Learning Methodology

Notation and Problem Formulation. Let a robot perform a set of exploratory behaviors $\mathcal{B}$ (e.g., *grasp*, *pick*) on a set of household objects $\mathcal{O}$ (e.g., *bottle*, *cup*), while recording a set of sensory modalities, $m = \{\mathrm{x}^v, \mathrm{x}^a, \mathrm{x}^h\}$, which correspond to *vision*, *audio*, *haptics*, respectively. The robot performs each behavior n times on each object. During the i^{th} exploratory trial, the robot collects sensory data m_i containing:

$$\mathrm{x}_i^v \in \mathbb{R}^{w \times h \times 3 \times t_i^v}, \mathrm{x}_i^a \in \mathbb{R}^{f \times t_i^a}, \mathrm{x}_i^h \in \mathbb{R}^{d \times t_i^h} \tag{10.1}$$

where w and h are the width and height of each image, f is the number of frequency bins in the sound spectrogram, d is the number of robot joint-torque sensors, and t_i^v,

t_i^a, and t_i^h are the number of time frames (e.g., number of images) produced during interaction for vision, audio, and haptics, respectively. Additionally, the robot has access to textual descriptions of each interaction, x_i^s, provided by human experts, complementing the sensory data.

Our primary objective is to learn a unified multimodal representation derived from the robot's observations across all modalities during an exploratory trial. To be more precise, we aim to learn the function $F_{m\to\mathcal{Z}} : x_i^v, x_i^a, x_i^h \to z_i$, where $z_i \in \mathbb{R}^{D_\mathcal{Z}}$ represents the unified multimodal embedding of dimension $D_\mathcal{Z}$. This unified representation is intended to encompass diverse object properties encountered during interactions, making it applicable to various downstream tasks that require understanding these object properties. By achieving this unified representation, the robot can rapidly adapt to different tasks by learning linear models or performing zero-shot transfers, thereby circumventing the need to train complex models dedicated to individual tasks.

Unified Multimodal Object Property Model. Our approach, MOSAIC (Multimodal Object property learning with Self-Attention and Integrated Comprehension), involves a two-stage process, illustrated in Fig. 10.1. Initially, we aim to distill unified object property representations from diverse sensory modalities, guided by text embeddings from a pre-trained text encoder. Subsequently, we leverage these unified representations to solve downstream tasks that require understanding object properties. In the following sections, we introduce various modules integrated within our framework.

10.3.0.1 Encoders and Feature Extraction

For the `Vision Encoder`, we use the CLIP's Vision Transformer (ViT-B/32) [RKH+21], which is jointly trained with a text encoder to maximize the similarity of {image, text} pairs using a contrastive loss. For each interaction's video, the image encoder extracts image embeddings, and these embeddings are then aggregated using adaptive average pooling to generate a feature vector of size $D_\mathcal{Z}$. For the `Audio Encoder`, we leverage the Wav2CLIP model [WSKB22], which is trained to project

Algorithm 1: `Training MOSAIC Framework`

$\boldsymbol{V}, \boldsymbol{A}, \boldsymbol{H}, \boldsymbol{S}$: Minibatch of aligned data (vision, audio, haptic, text)
$\boldsymbol{n}$: Size of minibatch
$\boldsymbol{MOSAIC_\theta}$: Learnable parameters of MOSAIC framework
// Extract feature vector for each modality
V_f = vision_encoder(V) // Vision Transformer
A_f = audio_encoder(A) // Wav2CLIP model
H_f = haptic_encoder(H) // ResNet18 model
S_f = text_encoder(S) // Text Transformer
// Compute unified representation
U_f = concatenation(V_f, A_f, H_f)
U_f = multihead_attention(U_f)
U_f = MLP_encoder(U_f) // MLP model
// Scaled pairwise cosine similarities
$logits = U_f \cdot S_f^\top$
// Symmetric loss function
$labels$ = range(n) // returns 1, 2, ..., n
$loss_u$ = cross_entropy_loss($labels, logits$)
$loss_s$ = cross_entropy_loss($labels, logits^\top$)
$loss = (loss_u + loss_s)/2$
Update $MOSAIC_\theta$ to minimize $loss$

audio data into the shared vision-language embedding space of CLIP; this approach enables the extraction of audio embeddings of size $D_{\mathcal{Z}}$. For the `Haptics Encoder`, we use a ResNet-18 [HZRS16] model, pre-trained on the ImageNet dataset, as the foundation. The input channels of the first convolutional layer are modified to one channel, and the output of the last fully-connected layer is adapted to match the desired embedding size of $D_{\mathcal{Z}}$; a sample haptic image is shown in Fig. 10.1. `Text Encoder`: For each exploratory trial, a corresponding natural language description is available. Leveraging CLIP's text encoder (ViT-B/32) [RKH+21], we extract embeddings of size $D_{\mathcal{Z}}$ from these text descriptions.

10.3.0.2 Multimodal Fusion

We employ a self-attention mechanism to integrate the feature sets from the three modalities. Beginning with the concatenation of feature vectors from each modality, we apply a two-step process: first, conventional multi-head self-attention [VSP+17] is applied to the concatenated features; subsequently, the resulting output is directed through a Multi-Layer Perceptron (MLP) to yield the unified multi-sensory feature of size $D_{\mathcal{Z}}$.

10.3.0.3 Training

During training, we maintain the vision, audio, and text encoders in their frozen states since they were already tuned to project into a shared embedding space. We train the haptic, self-attention, and MLP networks. Our primary aim is to create unified multimodal representations within the same embedding space as CLIP's text embeddings [RKH+21]. To accomplish this, we employ a distillation method guided by CLIP's text embeddings. We follow the approach outlined in the original CLIP paper, using a contrastive loss mechanism. This involves employing positive examples from different modalities within the same data sample while considering negative examples from the remaining batch. The fundamental implementation of this training process is shown in Algorithm 1. This strategy is predicated on the concept that natural language offers a versatile grounding basis [BHT+20], facilitating the creation of generalized representations with effective transferability across diverse downstream tasks.

10.4 Experimental Design

10.4.1 Sensory Dataset

We used the publicly accessible dataset collected by Sinapov *et al.* [SSS+14a]. In this experiment, a humanoid robot (depicted in Fig. 10.1) explored 100 household objects from 20 different categories (shown in Fig. 10.2A), using 10 exploratory behaviors. These behaviors included *Look*, *Press*, *Grasp*, *Hold*, *Lift*, *Drop*, *Poke*, *Push*, *Shake*, and *Tap* (shown in Fig. 10.2B). *Look* is a non-interactive behavior, only capturing visual data. For every interactive behavior, the robot collected sensory data including visual, audio, and haptic, acquired through three sensors: (1) A Logitech webcam capturing 320 x 240 RGB images at 10 frames per second; (2) An Audio-Technica U853AW cardioid microphone capturing audio sampled at 44.1 KHz; (3) Joint-torque sensors capturing torques from all 7 joints at 500 Hz. The robot repeated each behavior 5 times for each of the 100 objects, resulting in a total

Figure 10.2: (A) 100 objects, grouped in 20 object categories. (B) The interactive behaviors that the robot performed on the objects.

of 5,000 interactions (10 behaviors x 5 trials x 100 objects).

10.4.2 Text Dataset

The objects in our dataset were annotated with properties, shown in Table 10.1, each with corresponding values. While not all properties were applicable to every object (e.g., the *baseball* object lacked a weight property), we leveraged these properties to generate text descriptions for each interaction. To ensure diversity, we randomly selected a subset of properties for each object and used them in the descriptions. For each object's text description, we ensured that it included at least one property, and the maximum number of properties included was determined by the number of properties with values for that object. Moreover, we included the behavior's name being executed (e.g., *tap*), the object's category (e.g., *ball*), and the category of different object properties (e.g., *material*), all chosen randomly. Further variety was introduced by selecting synonyms for words within the description from a curated set of synonyms corresponding to the dataset's labels. We generated 100 unique text descriptions using this random selection process for each combination of object and behavior. For instance, an example text description might read: *"Performing tap action on a ball with properties: round, yellow, small, soft, toy"*.

10.4.3 Data Pre-processing

To ensure synchronization and consistency across all sensory modalities for each behavior $b \in \mathcal{B}$, we calculated the behavior's duration by dividing the average number of images recorded during behavior b by 10 (camera's frame rate). With the duration of each behavior now fixed, we compute the average number of time frames for each modality by multiplying this duration and the frame rate specific to that modality. These calculated averages were used for interpolation, ensuring uniform time frames for each modality during the interaction recording. For images and text, we employed the pre-processing provided by CLIP [RKH$^+$21]. Audio data was transformed from raw waveforms (1D) to spectrograms (2D) using the audio preprocessor from Wav2CLIP [WSKB22]. For haptic signals, we applied dimensionality reduction by interpolating the original 500Hz sampling rate down to 50Hz, drawing inspiration from a similar technique used in a prior study [TS19] conducted with the same dataset we used in our experiments.

10.4.4 Model Implementation

We standardized the size of the embeddings at $D_{\mathcal{Z}} = 512$. Our framework was implemented in PyTorch [PGM$^+$19], which includes the multi-sensory self-attention model and MLP encoder. We fine-tuned the model parameters over 50 epochs, using the Adam optimizer [KB15] with a learning rate of 10^{-4}.

10.4.5 Validation Procedure

Each of the 20 object categories consists of 5 unique objects. To train our framework, we selected 4 objects from each category for the training set while reserving one object for testing, resulting in a training set with 80 objects and a testing set with 20 objects. We employed a 5-fold object-based cross-validation strategy to ensure that each object appeared four times in the training set and once in the test set. Given that the robot interacted with each object 5 times, our training set contained 400 examples (80 objects $\times$ 5 trials); the test set comprised 100 examples (20 objects

Algorithm 2: `Fetch_object(`c, O, B, θ`)`

***MOSAIC*$_\theta$**: Learned parameters in Algorithm 1

t_c = text_encoder(c): // Command to fetch target
for $o \in O$: Set of objects (target and distractor(s)) **do**
  $similarity = 0$
  for $b \in B$: Set of Behaviors **do**
    $sensory\_data$ = perform_behavior(o, b)
    u_b = get_unified_repr($sensory\_data, MOSAIC_\theta$)
    $similarity$ += cosine_similarity(t_c, u_b)
  end
  Save $similarity$ for o
end
return Target Object o with highest $similarity$

Table 10.1: Property categories and associated descriptive words.

Properties	***Values***
Color	brown, blue, pink, red, white, orange, yellow, green, purple, multicolored
Deform.	deformable, rigid, brittle
Hardness	soft, squishy, hard
Material	plastic, wicker, aluminum, foam, metal, rubber, paper, styrofoam, wood
State	closed, full, empty, open
Reflection	shiny, dull
Shape	cylindrical, wide, rectangular, block, box, cone, round
Size	small, short, big, large, tall
Transp.	transparent, opaque, translucent, see-through
Usage	container, toy
Weight	light, heavy

$\times$ 5 trials) for each exploratory behavior.

10.4.6 Evaluation Tasks

After training our framework, we extracted the unified representations by freezing learned weights for all downstream tasks. We evaluated the acquired representations through two distinct tasks. The following subsections elaborate on these tasks, outlining our approach to tackling them with unified representations and discussing our performance metrics. Additionally, we discuss the baseline methods we employed for comparison with our method.

10.4.6.1 Object Category Recognition

In this task, the robot interacts with a given object to identify its category from a set of 20 categories. We use a standard multi-class linear classifier for supervised classification. Specifically, we use a Multi-Layer Perceptron (MLP) architecture that takes the unified representation as input, passes it through a hidden layer and

a ReLU activation function, and produces 20 logits for 20 categories. We train this classifier using the cross-entropy loss function for 50 epochs, using the Adam optimization with a learning rate of 10^{-4}. The trained classifier is then used to recognize the category of test objects, and we compute accuracy as a performance metric, defined as $A = \frac{\text{correct predictions}}{\text{total predictions}}$ (%). We report the mean accuracy over 5 cross-validation folds, as mentioned earlier.

10.4.6.2 Fetch Object

In this task, the robot receives a natural language instruction to fetch an object, specifying its properties (e.g., *"fetch an object that is cylindrical and short"*). The robot is then presented with a group of objects, among which one matches the specified properties (i.e., target object), while the remaining distractor object(s) differ from the target object in at least one property. To illustrate, if the robot is instructed to fetch an object that is both *cylindrical* and *short*, the distractor objects might be *cylindrical* or *short*, but not both. The robot's objective is to interact with these presented objects and correctly identify one with the requested properties. This task presents a challenge as the robot needs to detect the target object's properties given in natural language and distinguish it from the distractors by interaction. We evaluate the robot's performance on the fetch task across different levels of complexity. In this task, we refer to the given instruction as a "command" and the objects presented to the robot are carefully chosen from the previously mentioned test set, ensuring that they are entirely new to the robot. In difficulty `Level 1`, the command specifies the category name of the target object (e.g., *"fetch a ball"*); a distractor object is chosen from a different category. In `Level 2`, the command describes a specific property of the target object (e.g., *"bring an object that is hard"*). A distractor object is selected with a different property. In the `Level 3` scenario, the command includes two distinct properties of the target object (e.g., *"bring an object that is small and hard"*). The distractor object, on the other hand, possesses different properties. For `Level 4`, like Level 3, the command includes two target object properties. However, this time, two distractor objects are introduced,

each with differing properties. `Level 5` represents a variation of Level 2, where the commands only contain a property from a specific category, as illustrated in Table 10.1. For instance, in the *"Material"* category, the command might read, *"get an object that is plastic."* Level 5 was introduced to assess the robot's performance across various property categories. For each level, we created 20 commands for target objects and carefully selected corresponding distractor objects for each of the 5 previously explained folds. For each object (target and distractor(s)), we calculated its selection percentage, defined as $S = \frac{\text{number of times the object is selected}}{\text{total number of commands}}$ (%). Our results are reported as the mean selection percentage across the 5 folds.

We employ the approach outlined in Algorithm 2 to tackle this task. Initially, we convert the natural language instruction into a text embedding, denoted as t_c, using CLIP's text encoder (step 1). Subsequently, the robot interacts with the presented objects, including the target object and distractors, using various available behaviors while simultaneously recording sensory signals (step 5). To simulate this step, we randomly select a trial from our dataset among 5 trials of each object. Leveraging our trained framework, we generate unified representations, denoted as u_b, by processing the sensory inputs for each behavior (step 6). Next, we calculate the cosine similarity between the command embedding (t_c) and the unified representation (u_b) for each behavior, maintaining a cumulative similarity score (step 7). Finally, once all behaviors are considered, the object with the highest cumulative similarity score is identified as the target object, concluding the task (step 11).

10.4.7 Baseline, Ablation, and Comparison Conditions

We evaluate our full framework (MOSAIC), featuring the multi-sensory self-attention model, against an ablation framework that omits this component (MOSAIC-*w/o-SA*). These evaluations are conducted under two conditions: a non-interactive condition, where the robot solely performs the *Look* behavior, and an interactive condition, where the robot engages in all 9 interactive behaviors as listed earlier. Notably, in the *Look* behavior, only visual embeddings are employed as the unified representations after passing through the self-attention layer. Conversely, for interactive

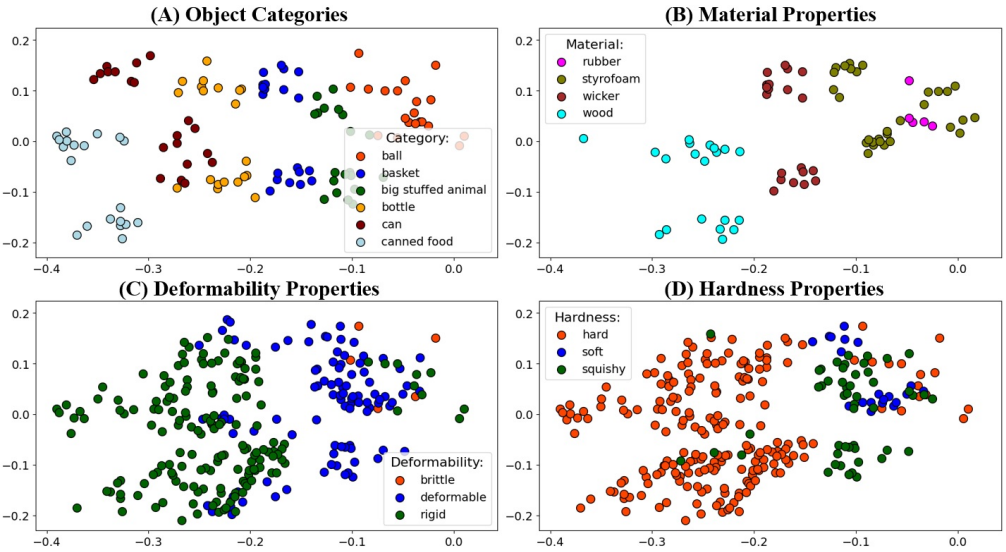


Figure 10.3: 2D unified representations derived from autoencoder trained on *Push* behavior's data: (A) Object categories, (B) Material, (C) Deformability, and (D) Hardness properties.

behaviors, all three modalities (i.e., visual, auditory, and haptic) are used to create unified representations. For the object category recognition task, we report recognition accuracy separately for the *Look* behavior, each of the 9 interactive behaviors individually, and the combination of all 9 interactive behaviors. The combined accuracy is calculated through a weighted combination of each behavior's performance on the training data. We also compare our recognition accuracy with two baseline methods: Sinapov *et al.* [SSS$^+$14a], who trained a Support Vector Machine (SVM) classifier using handcrafted auditory, haptic features, and visual features, and Tatiya et al. [TS19], who applied a deep learning approach to raw multi-sensory data for object category classification. For the fetch object task (see Algorithm 2), the set B contains only the *Look* behavior for the non-interactive condition and all 9 interactive behaviors for the interactive condition.

10.5 Results

10.5.1 An Illustrative Example

Let's consider a scenario where the robot performs the *Push* behavior on 80 objects (4 objects x 20 categories), recording visual, acoustic, and haptic data. With each

object undergoing 5 trials, this yields a dataset of 400 examples (80 objects $\times$ 5 trials). Using our MOSAIC framework, we use this data to learn unified representations. For visualization, we subjected these representations to dimensionality reduction using a linear autoencoder, resulting in a concise 2-dimensional latent space (Fig. 10.3). This visualization encapsulates four object properties: object categories, material, deformability, and hardness. Distinct colors are used to differentiate objects based on different values of these properties. To maintain clarity, we selectively plot only specific categories or objects with particular properties.

These visualizations unveil meaningful insights. Objects within the same category or material composition form tight clusters in the 2D space, showing the efficiency of our unified representations in capturing object semantics and material characteristics. The deformability properties plot demonstrates a separation between *rigid* and *deformable* objects, with *brittle* ones inclining towards *deformable*. Similarly, in the hardness properties plot, *hard* objects cluster on one side, while *soft* and *squishy* objects gravitate towards the opposite side. Essentially, our unified representations effectively encode objects with similar properties, as evidenced by distinct clusters of similar objects, even when these objects belong to different categories or material groups across various property categories. This illustrates MOSAIC's capacity to capture nuanced object attributes and relationships, a pivotal aspect of its performance across diverse tasks.

10.5.2 Object Category Recognition Results

Object category recognition results are presented in Table 10.2. Note that the *Look* behavior only relies on visual modality, and the "All behaviors" row at the bottom refers to all 9 interactive behaviors combined. Our approach, using unified representations, exhibits a remarkable level of competitiveness compared to state-of-the-art results for this dataset, demonstrating higher recognition accuracy in seven out of ten behaviors. For the remaining three behaviors, we achieved comparable accuracy. We achieved this level of performance using a straightforward linear model on top of the unified representations, a contrast to previous methods. Notably, the prior work

Table 10.2: Category recognition accuracy (%) for each behavior.

Behavior	***Sinapov** et al. **[SSS+14a]***	***Tatiya** et al. **[TS19]***	***MOSAIC**-w/o-SA*	***MOSAIC (ours)***
Look	67.7	—	86.4 ± 1.2	**87.4 ± 2.0**
Grasp	65.2	71.4	72.2 ± 6.7	**74.0 ± 5.8**
Hold	67.0	**76.8**	68.0 ± 5.3	69.6 ± 5.2
Lift	**79.0**	77.8	72.8 ± 4.2	77.8 ± 5.7
Drop	71.0	**78.0**	73.2 ± 3.8	77.2 ± 5.9
Poke	85.4	73.8	81.6 ± 2.2	**86.4 ± 1.0**
Push	88.8	67.4	85.6 ± 3.5	**89.4 ± 4.4**
Shake	76.8	83.6	81.2 ± 6.2	**84.0 ± 5.6**
Tap	82.4	81.6	81.2 ± 5.7	**84.4 ± 1.8**
Press	77.4	58.8	71.6 ± 8.7	**77.8 ± 6.4**
All behaviors	—	—	95.2 ± 3.6	**95.6 ± 3.9**

[TS19] employed a specialized neural network architecture tailored specifically for this task, while [SSS+14a] relied on handcrafted features. Furthermore, our results consistently indicate that our full framework, including self-attention, outperforms the counterpart without self-attention. This underscores the utility of the multi-sensory unified representation and the effectiveness of the self-attention mechanism in enhancing the robot's adaptability to diverse tasks.

10.5.3 Fetch Object Results

The fetch object task, whose results are summarized in Table 10.3, comprises five distinct levels designed to assess the robot's ability to execute instructions. In `Level 1`, the command specified the object category name. Our complete MOSAIC framework excelled in interactive behavior conditions, achieving an impressive target object selection rate of 99.0%, outperforming all baseline models. `Level 2 to Level 5`: These levels introduced object properties into the command instead of specifying the object category name. Generally, the interactive behaviors condition outperformed the non-interactive one, with our full MOSAIC model excelling in most cases. Interestingly, providing more object properties in the command led to better performance, exemplified by a higher target object selection rate in Level 3 compared to Level 2, across all conditions, except for *"Look"* without self-attention. This suggests that learning unified representations with self-attention prioritizes the most relevant object properties. `Level 4` presented greater challenges due to

the inclusion of two distractor objects resembling the target object. Nevertheless, our complete MOSAIC framework with self-attention consistently outperformed all baselines.

To evaluate the robot's ability to fetch objects based on specific property categories, we delved into Level 5, where the command included only descriptive words related to specific property categories. For simplicity, we focused on discussing five property categories. `Deformability and Weight`: In scenarios involving non-visual properties like deformability and weight, the interactive behaviors condition significantly outperformed the non-interactive one. This aligns with intuition, as visual observation alone may not suffice to determine these properties. `Transparency and Size`: For visual properties like transparency and size, the interactive behaviors condition performed comparably to the non-interactive behavior, suggesting that interaction with objects may not yield significantly more information in these scenarios. `Shape`: Intriguingly, for the shape property category, the interactive behaviors condition significantly outperformed the non-interactive one. This implies that interacting with objects enables the robot to observe them from various angles, enhancing its ability to predict object shape compared to merely observing from a top angle. In summary, our full MOSAIC framework demonstrated robust performance in the fetch object task, relying solely on unified representations without additional learning methods. These results underscore the adaptability and applicability of unified representations across diverse tasks, including those involving natural language instructions.

10.6 Summary

We introduced the MOSAIC framework to enable robots to generate versatile, multimodal representations through interactive object perception and to leverage these unified representations across various downstream robot learning tasks. Through extensive performance evaluation, we have showcased the effectiveness of these unified representations in tasks such as category recognition, using a simple linear probe

Table 10.3: MOSAIC's target object selection (%) in various levels of the fetch object task, with and without Self-Attention.

	Look (non-interactive)		***Interactive***	
	-w/o-SA	***MOSAIC***	*-w/o-SA*	***MOSAIC***
Level 1	74	82	97	**99**
Level 2	61	65	**84**	81
Level 3	60	74	**86**	83
Level 4	54	70	72	**77**
Level 5:				
Color	64	76	85	**89**
Deformation	45	48	71	**74**
Hardness	60	58	66	**72**
Material	69	83	91	**95**
Object State	49	55	70	**72**
Shape	85	80	**97**	95
Size	62	74	72	**75**
Transparency	62	62	51	**63**
Usage	75	68	79	**90**
Weight	52	63	**85**	**85**

setup, and the fetch object task under zero-shot conditions.

Moving forward, there are several exciting directions for future research. Firstly, we plan to consider the *transfer* of unified representations across different robot morphologies, enabling a broader range of robots to benefit from this technology. Furthermore, we envision settings where interactive behaviors are learned and composed, alongside the tasks we considered in this chapter, thereby further increasing the efficacy of object exploration. These future endeavors hold the potential to further enhance the utility of unified representations in robotics and expand their applications across a multitude of scenarios and environments. One limitation in our current study is that, for the fetch object task, we evaluated using a zero-shot transfer condition rather than a learning-based approach to find the target object. For future work, it would be important to explore learning-based policies for solving the fetch object task, potentially increasing the versatility and adaptability of our framework.

Chapter 11

Conclusion and Future Work

Transferring object property representations from experienced robots to new robots to expedite learning and bolster task efficiency is a compelling aspiration. However, this aspiration faces a challenge — the frequently occurring variance in robots' interaction capabilities, manifesting as differences in physical embodiment, sensory equipment, behaviors, or even tools. This creates a barrier to the direct transfer of perceptual knowledge from one robot to another. This dissertation tackles this challenge that arises when robots share the perceptual knowledge acquired during their interactions with objects. This dissertation introduced innovative frameworks designed to cross-transfer multi-sensory perceptual knowledge, painstakingly acquired through interactions with objects, across a diverse spectrum of robot embodiments, behaviors, sensors, and tools. A series of experiments involving robots have underscored the efficacy of these proposed knowledge transfer frameworks, showcasing their capacity to facilitate perceptual knowledge transference across distinct sensorimotor contexts. These empirical outcomes not only validate the feasibility of our approach but also underscore its potential to empower newly deployed robots, endowing them with the capacity to efficiently learn object properties through the inheritance of perceptual knowledge from more experienced robots.

11.1 Summary of Contributions

In response to the central question, ***How can robots transfer perceptual knowledge acquired through object interactions across heterogeneous robot embodiments, behaviors, sensors, and tools?*** this dissertation has made the following significant contributions:

Datasets: We have published three substantial object exploration datasets, encompassing multi-sensory signals recorded by heterogeneous robots during various object-interaction behaviors. These datasets and the source code of the proposed frameworks are publicly accessible to support further research.

Multimodal Object Categorization: We have developed methodologies that combine visual, auditory, and haptic sensory data for object categorization through interactive behavior. Chapter 4 introduces multimodal deep neural network-based architectures for object categorization, while Chapter 10 presents a method for learning unified multi-sensory representations, simplifying object categorization via simple linear models.

Generative Models for Knowledge Transfer: Our frameworks leverage generative models to map sensory data from a source robot to a semantically similar feature space for a target robot. Chapter 5 introduces Encoder-Decoder Networks (EDN) for cross-behavior knowledge transfer, while Chapter 7 extends this approach to support multiple source contexts using β-Variational Autoencoder Networks (β-VAE).

Shared Latent Feature Space: We propose frameworks for transferring perceptual knowledge between robots through a shared latent feature space. Chapter 6 incorporates kernel manifold alignment (KEMA), and Chapter 9 introduces metric learning via triplet loss for mapping sensory data into a common latent space.

MOSAIC Framework: Chapter 9 introduces MOSAIC (Multimodal Object Property Learning with Self-Attention and Integrated Comprehension), enabling multi-sensory integration for learning unified multimodal representations from robot object exploration data. These representations are transferable across tasks, significantly

reducing object exploration time.

Knowledge Transfer Evaluation: We have introduced novel ways to evaluate and analyze knowledge transfer performance. Chapter 5 presents "accuracy delta" for quantifying performance drops due to transferred features. Chapters 7 and 9 introduce accuracy delta matrices and 2D visualizations of transfer relations between sensorimotor contexts.

Object Selection Algorithm: In Chapter 7, we propose an object selection algorithm for efficient calibration set selection in scenarios with limited time for knowledge transfer mapping.

Data Augmentation Technique: Chapter 8 introduces a data augmentation technique that enhances knowledge transfer model generalization, applicable across various downstream tasks, with evaluations conducted in object-property and object-identity recognition scenarios.

These contributions collectively advance the field of multi-sensory perceptual knowledge transfer in robotics.

11.2 Interconnections among Proposed Frameworks

The three proposed frameworks presented in this dissertation, namely *Transfer using Projection to Target Feature Space*, *Transfer using Projection to Shared Latent Feature Space*, and *Transferable Unified Multi-sensory Object Property Representations*, are intricately interconnected through a common thread—the facilitation of knowledge transfer across robots to enhance their interactive perceptual capabilities. While each framework tackles distinct challenges and employs different methodologies, their unifying goal is to bridge the gap between robots, enabling them to seamlessly leverage acquired knowledge for improved perception and understanding of objects.

The first framework, *Transfer using Projection to Target Feature Space*, establishes a foundation for transferring knowledge by mapping sensory data from a source robot to a semantically similar feature space for a target robot. By lever-

aging encoder-decoder networks and principles of domain adaptation, this framework addresses the challenges of disparate physical attributes and sensor models between robots. The emphasis on semantic similarity and the incorporation of human-provided object labels lay the groundwork for effective knowledge transfer.

Building upon this foundation, the second framework, *Transfer using Projection to Shared Latent Feature Space*, introduces domain adaptation strategies, specifically kernel manifold alignment (KEMA), to align datasets and create a shared latent space. This shared space allows for a more efficient transfer of non-visual object knowledge among robots with varying physical attributes and sensor configurations. The emphasis on non-visual data and the introduction of metric learning via triplet loss contribute to the adaptability and practicality of knowledge transfer among heterogeneous robots.

The third framework, *Transferable Unified Multi-sensory Object Property Representations*, takes a unique approach by drawing inspiration from contrastive learning, particularly CLIP [RKH+21]. MOSAIC, the proposed framework within this category, distills knowledge from the CLIP text model to learn unified multi-sensory object property representations. By aligning representations across visual, haptic, and auditory sensory domains, MOSAIC contributes significantly to advancing the multi-sensory perceptual capabilities of autonomous systems.

In summary, these frameworks are not isolated solutions but rather components of a holistic strategy for enabling robots to acquire, share, and utilize knowledge about objects through interactive behaviors and multi-sensory perception. Together, they form a comprehensive approach to addressing the challenges of knowledge transfer in the context of robotic perception, laying the groundwork for more versatile and adaptable robotic systems across diverse domains.

11.3 Applicability and Boundaries

In the pursuit of developing transfer learning frameworks for robotic systems, it is imperative to outline the specific scenarios where the proposed methods exhibit

efficacy and the circumstances under which limitations may arise. The applicability and boundaries of the frameworks are contingent upon several vital factors, outlined below:

11.3.1 Applicability

Interactive Object Interaction: The frameworks are designed for robotic arms with specified degrees of freedom engaged in exploratory interactions with objects. This includes behaviors such as lifting, shaking, pushing, or employing tools for object exploration, such as exploring food objects using kitchen tools. These interactions constitute a crucial foundation for effective knowledge transfer.

Consistent Sensorimotor Context: Assumed input data is consistently structured, representing a singular behavior trajectory executed by the robot, encapsulating a specific behavior-sensory modality pair. This singular trajectory, whether it involves shaking, pushing, or another isolated behavior, ensures that the frameworks are adept at handling distinct behaviors, enhancing their practicality in real-world robotic scenarios.

Common Object Interaction: Effective application is observed when both the source and target robots have interacted with a common set of objects or objects with common object properties. These object properties serve as invariant descriptors provided by humans in the form of labeled data, and these labels are crucial for building correspondences used to learn the projection functions proposed in this dissertation. Additionally, these labeled data serve as an additional set of sensors, leveraging the rich senses of the human body, and benefit from thousands of years of language development, optimizing labels for usefulness in various contexts.

Enhanced Data Alignment for Similar Object Properties: Optimal performance is achieved when the sensorimotor contexts of both source and target robots capture similar object properties. In other words, data alignment is better when the robots perform similar behaviors (e.g., lifting, holding) or share the same sensory modality that captures similar object properties.

Source-Target Experience Discrepancy: The frameworks are particularly ben-

eficial when the target robot has limited experience in object exploration compared to the source robot. The primary objective is to enhance the performance of the less experienced target robot by transferring implicit knowledge gained through more experienced source robots via object exploration.

11.3.2 Boundaries

Non-Interactive or Disembodied Observation: Limitations emerge when robotic observation is non-interactive or disembodied, deviating from the foundational premise of the frameworks centered around interactive object exploration. It is worth noting that adapting to non-interactive observations, such as images from different cameras, has been addressed in previous research. However, this dissertation specifically focuses on adapting interactive observations from robots with different embodiments.

Inconsistent Sensorimotor Context: Challenges arise when input data lacks consistent segmentation as a sensorimotor context, especially when the data used as input is not appropriately segmented based on sensorimotor context. Simply put, the proposed frameworks may not perform well if trials for a sensorimotor context vary significantly.

Lack of Common Object Interaction: The absence of a shared set of objects or the unavailability of labeled data from humans hinders the effective functioning of the frameworks. Our frameworks are built upon the assumption that if robots interact with a shared set of objects or objects with similar properties, the produced sensory data can be used to learn a mapping between the robots' feature spaces.

Divergent Object Properties: Performance may be compromised when source and target sensorimotor contexts capture disparate object properties, disrupting the alignment necessary for successful knowledge transfer. For instance, aligning a *hold-haptic* context, which excels at capturing weight properties, with a *drop-audio* context, which excels at capturing sound properties, can be challenging due to the dissimilarity in the object properties they emphasize. This highlights the importance of considering the nature of object properties captured by different sensorimotor contexts when aiming for effective knowledge transfer between robots.

Equal or Greater Target Robot Experience: Significantly less improvement is anticipated if the target robot already possesses equivalent or more experience in object exploration than the source robot, diminishing the impact of knowledge transfer on performance enhancement. The frameworks assume that the target robot has explored fewer objects or conducted fewer trials exploring objects compared to the source robot. Therefore, the goal is to transfer additional experience from the more experienced source robot to enhance the target robot's performance effectively.

By explicitly delineating these conditions, this dissertation provides a comprehensive understanding of the nuanced scenarios where the proposed frameworks excel and where their effectiveness may be tempered. These insights contribute to a more nuanced perspective on the deployment and implications of the developed transfer learning methodologies in real-world robotic applications.

11.4 Generalization to Diverse Robotic Systems and Object Categories

The transfer learning frameworks developed in this dissertation pave the way for advancements in robotic knowledge transfer, particularly in the context of object exploration. As we conclude this work, it is essential to consider the possibilities of extending these frameworks to diverse robotic systems and a broader spectrum of object categories. The following points elucidate the potential avenues for generalization:

11.4.1 Generalizing to Diverse Robots

The developed transfer learning frameworks are not constrained by assumptions about specific robot embodiments. They can be adapted to various robotic platforms with distinct morphologies, interaction capabilities, and feature representations for a given modality. The key to this adaptability lies in identifying commonalities in sensorimotor contexts and object properties across different robot types. The modular nature of the proposed methodologies allows for flexibility in integrating

them into diverse robotic architectures.

11.4.2 Adapting to Diverse Object Categories

Our transfer learning frameworks do not impose any restrictions on specific sets of objects. Instead, we assume that humans provide labels about object properties. Through our experiments, we observed that incorporating a larger and more diverse set of objects for learning the projection function enhances the frameworks' performance in knowledge transfer. Expanding the applicability to a broader range of object categories involves considering the inherent diversity in object properties and interaction modalities. By augmenting the labeled dataset with additional object categories and properties, the frameworks can be fine-tuned to accommodate variations in object characteristics, fostering a more comprehensive understanding of different object types.

11.5 Future Work

In the realm of perceptual knowledge transfer in robotics, this dissertation has laid the foundation for several promising future research avenues, highlighting pertinent questions that remain open within the field. Below, we discuss these areas and propose directions for future investigations.

11.5.1 Efficient Object Exploration for Knowledge Transfer

While this dissertation employed random object selection for training the projection functions essential to perceptual knowledge transfer, Chapter 7 revealed that heuristic-based object selection algorithms, particularly those focused on capturing similar object properties, can significantly enhance knowledge transfer models. Future work could center on the development of algorithms capable of automating the object selection process, leading to more efficient knowledge transfer models. This involves not only selecting objects based on their properties but also exploring the possibility of identifying specific segments within a behavior that better

capture object properties for knowledge transfer. For instance, in the case of the drop behavior, automatically identifying the segment when the object is dropped and makes distinct sounds could be more informative for capturing object properties. Additionally, our experiments exhaustively transferred knowledge across all feasible sensorimotor contexts within our robotic platform. A promising direction for further research involves enabling robots to intelligently select behaviors, tools, or modalities within source and target sensorimotor contexts, thereby optimizing object exploration and knowledge transfer efficiency. Developing methods to automatically identify behavior segments that are most informative for perceptual knowledge transfer can be a significant step towards achieving this goal. This would not only enhance the efficiency of the knowledge transfer process but also contribute to the broader understanding of which aspects of behaviors are crucial for capturing object properties.

11.5.2 Enhancing Adaptability Through Learning-Based Policies

In this dissertation, once the knowledge transfer model is trained, we utilized the transferred features to address various downstream recognition tasks, such as object category recognition, object identity recognition, and object property recognition. While these tasks served as valuable benchmarks for evaluating the performance of the proposed knowledge transfer models, the field of robotics offers numerous challenges that demand more complex solutions. As demonstrated in Chapter 10, we considered a more intricate task: the fetch object task. However, our approach for this task primarily relied on heuristic-based algorithms under the zero-shot transfer condition, eliminating the need for an extensive learning stage. For future research, exploring learning-based policies is crucial to address more complex tasks that require advanced adaptability and intelligence. This involves developing policies beyond simple recognition tasks, enabling robots to adapt to diverse and intricate scenarios autonomously. Some potential avenues for research include:

Adaptive Behaviors Based on Object Properties: Investigate learning-based policies for adapting robot behaviors based on specific object properties. For in-

stance, handling fragile plastic differently from metal or adjusting the handover strategy based on the person's grasp, considering potential occlusions.

Non-Visual Signal Integration: Explore policies that incorporate non-visual signals into robotic decision-making. Examples include learning to push a charger into a socket until a distinct click sound is heard or interpreting other non-visual cues to enhance task performance.

Complex Packing Strategies: Develop policies for complex grocery packing scenarios where the robot must consider the properties of different items. For instance, they ensure that cold frozen items are separated from hot food items or avoid placing heavy items on top of delicate ones.

This expanded scope aims to pave the way for a comprehensive exploration of learning-based policies, pushing the boundaries of adaptability in robotic systems. Integrating such policies into our existing framework can significantly enhance the practicality and versatility of perceptual knowledge transfer models, contributing to the broader landscape of robotic intelligence.

11.5.3 Autonomous Learning of Exploratory Behaviors for Enhanced Knowledge Transfer

In the datasets utilized throughout this dissertation, robots are programmed to execute predefined joint space trajectories, known as behaviors, to interact with objects. These behaviors are manually encoded across multiple robots, a time-consuming process prone to variations in encoding for similar behaviors across different robots. For instance, the "shake" behavior on a UR5 robot may be encoded differently than on a Baxter robot. A more robust and efficient approach involves enabling robots to learn these behaviors autonomously. An exciting avenue for future research is the development of methods that enable robots to independently learn behaviors optimized for enhancing the performance of knowledge transfer models. Several promising methodologies within the scope of a PhD dissertation include:

Human-Imitation Learning: Conduct human studies to create datasets of humans exploring objects to learn their properties. Implement imitation learning set-

tings, allowing robots to learn behaviors from expert demonstrations, thus capturing nuanced and effective exploratory actions.

Policy Refinement for Predefined Trajectories: Investigate methods where robots begin with predefined joint space trajectories for different behaviors and autonomously learn policies that refine and adapt these behaviors to efficiently capture object properties in reduced durations.

End-to-End Reinforcement Learning Policies: Explore the training of end-to-end policies within a reinforcement learning framework for autonomous object exploration. These policies can dynamically adapt and optimize behaviors based on real-time feedback, enhancing the adaptability of robots during interactive exploration.

Utilizing Internet Videos for Behavior Learning: Leverage videos from the internet showcasing humans performing daily-life activities with objects to extract valuable insights into exploratory behaviors. Develop methodologies to transfer this knowledge to robots, enabling them to autonomously learn behaviors for efficient object exploration.

The proposed research not only addresses the limitations of manual behavior encoding but also opens the door to collaborative learning, where multiple robots collectively contribute to the refinement of exploratory behaviors. This ambitious exploration into autonomous learning of behaviors aims to significantly improve knowledge transfer efficiency.

Bibliography

[ABC+16] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, et al. Tensorflow: A system for large-scale machine learning. In *12th Symposium on Operating Systems Design and Implementation*, pages 265–283, 2016.

[ADL+22] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

[AHMJ+14] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10):1533–1545, 2014.

[AHMJP12] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, and Gerald Penn. Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 4277–4280. IEEE, 2012.

[AKA91] David W Aha, Dennis Kibler, and Marc K Albert. Instance-based learning algorithms. *Machine learning*, 6(1):37–66, 1991.

[ANM10] David Alais, Fiona Newell, and Pascal Mamassian. Multisensory processing in review: from physiology to behaviour. *Seeing and perceiving*, 23(1):3–38, 2010.

[ANN+12] Takaya Araki, Tomoaki Nakamura, Takayuki Nagai, Kotaro Funakoshi, Mikio Nakano, and Naoto Iwahashi. Online object categorization using multimodal information autonomously acquired by a mobile robot. *Adv. Robotics*, 26(17):1995–2020, 2012.

[APR+20] Jacob Arkin, Daehyung Park, Subhro Roy, Matthew R Walter, Nicholas Roy, Thomas M Howard, and Rohan Paul. Multimodal estimation and communication of latent semantic knowledge for robust execution of robot instructions. *The International Journal of Robotics Research*, page 0278364920917755, 2020.

[AWZ+18] Saeid Amiri, Suhua Wei, Shiqi Zhang, Jivko Sinapov, Jesse Thomason, and Peter Stone. Multi-modal predicate identification using dynamically learned robot controllers. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2018.

[bax] Baxter figure. `https://www.computerhistory.org/collections/catalog/102751979`. Accessed: 2022-09-10.

[BCB14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[BDBC+10] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

[BDFP02] A Borghi, A Di Ferdinando, and D Parisi. The role of perception and action in object categorisation. In *Connectionist models of cognition and perception*, pages 40–50. World Scientific, 2002.

[BG06] David A Bulkin and Jennifer M Groh. Seeing sounds: visual and auditory interactions in the brain. *Current opinion in neurobiology*, 16(4):415–419, 2006.

[BG11] David F Bjorklund and Amy K Gardiner. Object play and tool use: Developmental and evolutionary perspectives. 2011.

[BGS+20] Raphaël Braud, Alexandros Giagkos, Patricia Shaw, Mark Lee, and Qiang Shen. Robot multi-modal object perception and recognition: Synthetic maturation of sensorimotor learning in embodied systems. *IEEE Transactions on Cognitive and Developmental Systems*, 2020.

[BHS+17] Jeannette Bohg, Karol Hausman, Bharath Sankaran, Oliver Brock, Danica Kragic, Stefan Schaal, and Gaurav S Sukhatme. Interactive perception: Leveraging action in perception and perception in action. *IEEE Transactions on Robotics*, 33(6):1273–1291, 2017.

[BHT+20] Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, et al. Experience grounds language. *arXiv preprint arXiv:2004.10151*, 2020.

[BJT16] Jennifer K Bizley, Gareth P Jones, and Stephen M Town. Where are multisensory signals combined for perceptual decision-making? *Current opinion in neurobiology*, 40:31–37, 2016.

[BKC17] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[BKM17] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[BLSS19] Tapomayukh Bhattacharjee, Gilwoo Lee, Hanjun Song, and Siddhartha S Srinivasa. Towards robotic feeding: Role of haptics in fork-based food manipulation. *IEEE Robotics and Automation Letters*, 4(2):1485–1492, 2019.

[BPT+22] Samar Bashath, Nadeesha Perera, Shailesh Tripathi, Kalifa Manjang, Matthias Dehmer, and Frank Emmert Streib. A data-centric review of deep transfer learning with applications to text data. *Information Sciences*, 585:498–528, 2022.

[BRK12a] T. Bhattacharjee, J. M. Rehg, and C. C. Kemp. Haptic classification and recognition of objects using a tactile sensing forearm. In *IEEE RSJ*, pages 4090–4097, Oct 2012.

[BRK12b] Tapomayukh Bhattacharjee, James M Rehg, and Charles C Kemp. Haptic classification and recognition of objects using a tactile sensing forearm. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4090–4097. IEEE, 2012.

[BRPM16] Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Bmvc*, volume 1, page 3, 2016.

[BSO+09] Taylor Bergquist, Connor Schenck, Ugonna Ohiri, Jivko Sinapov, Shane Griffith, and Alexander Stoytchev. Interactive object recognition using proprioceptive feedback. In *Proceedings of the 2009 IROS Workshop: Semantic Perception for Robot Manipulation, St. Louis, MO*, 2009.

[Bur98] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[CBL+23] Jireh Yi-Le Chan, Khean Thye Bea, Steven Mun Hong Leow, Seuk Wai Phoong, and Wai Khuen Cheng. State of the art: a review of sentiment analysis based on sequential transfer learning. *Artificial Intelligence Review*, 56(1):749–780, 2023.

[CD89] Kevin Connolly and Mary Dalgleish. The emergence of a tool-using skill in infancy. *Developmental Psychology*, 25(6):894, 1989.

[Cha14] Ishani Chakraborty. *Object category recognition through visual-semantic context networks*. Rutgers The State University of New Jersey-New Brunswick, 2014.

[CHPS21] Xiaohui Chen, Ramtin Hosseini, Karen Panetta, and Jivko Sinapov. A framework for multisensory foresight for embodied agents. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10927–10933. IEEE, 2021.

[CLS+18] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[CMR+15] Vivian Chu, Ian McMahon, Lorenzo Riano, Craig G McDonald, Qin He, Jorge Martinez Perez-Tejada, Michael Arrigo, Trevor Darrell, and Katherine J Kuchenbecker. Robotic learning of haptic adjectives through physical interaction. *Robotics and Autonomous Systems*, 63:279–292, 2015.

[CSS+04] Gemma Calvert, Charles Spence, Barry E Stein, et al. *The handbook of multisensory processes*. MIT press, 2004.

[CT04] Gemma A Calvert and Thomas Thesen. Multisensory integration: methodological approaches and emerging principles in the human brain. *Journal of Physiology-Paris*, 98(1-3):191–205, 2004.

[CUH16] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). In *International Conference on Learning Representations*, 2016.

[CZBN21] Shaoyu Cai, Kening Zhu, Yuki Ban, and Takuji Narumi. Visual-tactile cross-modal data generation using residue-fusion gan with feature-matching and perceptual losses. *IEEE Robotics and Automation Letters*, 6(4):7525–7532, 2021.

[CZCL23] Chongyang Chen, Kem ZK Zhang, Zhaofang Chu, and Matthew Lee. Augmented reality in the metaverse market: the role of multimodal sensory interaction. *Internet Research*, 2023.

[DAHG+15] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[DBS11] Sander Dieleman, Philémon Brakel, and Benjamin Schrauwen. Audio-based music classification with a pretrained convolutional network. In *12th International Society for Music Information Retrieval Conference (ISMIR-2011)*, pages 669–674. University of Miami, 2011.

[DFBP02] Andrea Di Ferdinando, Anna M Borghi, and Domenico Parisi. The role of action in object categorization. In *FLAIRS Conference*, pages 138–142, 2002.

[DSP+16] Andreas Doumanoglou, Jan Stria, Georgia Peleka, Ioannis Mariolis, Vladimir Petrik, Andreas Kargakos, Libor Wagner, Václav Hlaváč, Tae-Kyun Kim, and Sotiris Malassiotis. Folding clothes au-

tonomously: A complete pipeline. *IEEE Transactions on Robotics*, 32(6):1461–1478, 2016.

[DWLZ21] Guoguang Du, Kai Wang, Shiguo Lian, and Kaiyong Zhao. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review*, 54(3):1677–1734, 2021.

[DXC12] Lixin Duan, Dong Xu, and Shih-Fu Chang. Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In *2012 IEEE Conference on computer vision and pattern recognition*, pages 1338–1345. IEEE, 2012.

[EB04] Marc O Ernst and Heinrich H Bülthoff. Merging the senses into a robust percept. *Trends in cognitive sciences*, 8(4):162–169, 2004.

[ECK17] Zackory Erickson, Sonia Chernova, and Charles C Kemp. Semi-supervised haptic material recognition for robots using generative adversarial networks. *arXiv preprint arXiv:1707.02796*, 2017.

[EKSW18] Manfred Eppe, Matthias Kerzel, Erik Strahl, and Stefan Wermter. Deep neural object analysis by interactive auditory exploration with a humanoid robot. In *IEEE Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[ELCK19] Zackory Erickson, Nathan Luskey, Sonia Chernova, and Charles C Kemp. Classification of household materials via spectroscopy. *IEEE Robotics and Automation Letters*, 4(2):700–707, 2019.

[ERCM18] A Gómez Eguíluz, Ignacio Rano, Sonya A Coleman, and T Martin McGinnity. Multimodal material identification through recursive tactile sensing. *Robotics and Autonomous Systems*, 106:130–139, 2018.

[EXS+20] Zackory Erickson, Eliot Xing, Bharat Srirangam, Sonia Chernova, and Charles C Kemp. Multimodal material classification for robots

using spectroscopy and high resolution texture imaging. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10452–10459. IEEE, 2020.

[FKL+22] Jonathan Francis, Nariaki Kitamura, Felix Labelle, Xiaopeng Lu, Ingrid Navarro, and Jean Oh. Core challenges in embodied vision-language planning. *Journal of Artificial Intelligence Research*, 74:459–515, 2022.

[FL12] Jeremy A Fishel and Gerald E Loeb. Bayesian exploration for intelligent identification of textures. *Frontiers in neurorobotics*, 6:4, 2012.

[FLN+19] Pietro Falco, Shuang Lu, Ciro Natale, Salvatore Pirozzi, and Dongheui Lee. A transfer learning approach to cross-modal object recognition: from visual observation to robotic haptic exploration. *IEEE Transactions on Robotics*, 35(4):987–998, 2019.

[Fra22] Jonathan Francis. *Knowledge-enhanced Representation Learning for Multiview Context Understanding*. PhD thesis, Carnegie Mellon University, 2022.

[GB10] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[GBB11] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520, 2011.

[GBKS19] Daniel Gallenberger, Tapomayukh Bhattacharjee, Youngsun Kim, and Siddhartha S Srinivasa. Transfer depends on acquisition: An-

alyzing manipulation strategies for robotic feeding. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 267–276. IEEE, 2019.

[GDL+17] Abhishek Gupta, Coline Devin, YuXuan Liu, Pieter Abbeel, and Sergey Levine. Learning invariant feature spaces to transfer skills with reinforcement learning. In *International Conference on Learning Representations*, 2017.

[GFL19] Shuhao Gu, Yang Feng, and Qun Liu. Improving domain adaptation translation with domain invariant and specific information. *arXiv preprint arXiv:1904.03879*, 2019.

[GGP20] Dhiraj Gandhi, Abhinav Gupta, and Lerrel Pinto. Swoosh! Rattle! Thump! - Actions that Sound. In *Proceedings of Robotics: Science and Systems*, 2020.

[GHKD16] Yang Gao, Lisa Anne Hendricks, Katherine J Kuchenbecker, and Trevor Darrell. Deep learning for tactile understanding from visual and haptic data. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 536–543. IEEE, 2016.

[Gib88] Eleanor J Gibson. Exploratory behavior in the development of perceiving, acting, and the acquiring of knowledge. *Annual review of psychology*, 39(1):1–42, 1988.

[GJM13] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE, 2013.

[GPM+22] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain

adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022.

[GS14] Mevlana C. Gemici and Ashutosh Saxena. Learning haptic representation for manipulating deformable food objects. In *Intelligent Robots and Systems (IROS)*, pages 638–645, Chicago, IL, USA, Sep 2014. IEEE.

[GSC+22] Ruohan Gao, Zilin Si, Yen-Yu Chang, Samuel Clarke, Jeannette Bohg, Li Fei-Fei, Wenzhen Yuan, and Jiajun Wu. Objectfolder 2.0: A multisensory object dataset for sim2real transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10598–10608, June 2022.

[GZ16] Haojun Guan and Jianwei Zhang. Multi-sensory based novel household object categorization system by using interactive behaviours. In *Robotics and Biomimetics (ROBIO), 2016 IEEE International Conference on*, pages 1685–1690. IEEE, 2016.

[HBMK16] Virgile Högman, Mårten Björkman, Atsuto Maki, and Danica Kragic. A sensorimotor learning framework for object categorization. *IEEE Transactions on Cognitive and Developmental Systems*, 8(1):15–25, 2016.

[Hel92] Morton A Heller. Haptic dominance in form perception: vision versus proprioception. *Perception*, 21(5):655–660, 1992.

[HGY22] Hung-Jui Huang, Xiaofeng Guo, and Wenzhen Yuan. Understanding dynamic tactile sensing for liquid property estimation. In *Robotics Science and System*, 2022.

[HMG+22] Asmaul Hosna, Ethel Merry, Jigmey Gyalmo, Zulfikar Alom, Zeyar Aung, and Mohammad Abdul Azim. Transfer learning: a friendly introduction. *Journal of Big Data*, 9(1):102, 2022.

[HMP+17] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. Beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2(5):6, 2017.

[HS06] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[HW17] Xinyu Hua and Lu Wang. A pilot study of domain adaptation effect for neural abstractive summarization. *arXiv preprint arXiv:1707.07062*, 2017.

[HWL+20] Lijun Han, Hesheng Wang, Zhe Liu, Weidong Chen, and Xiufeng Zhang. Vision-based cutting control of deformable objects with surface tracking. *IEEE/ASME Transactions on Mechatronics*, 26(4):2016–2026, 2020.

[HXJ+23] Yafei Hu, Quanting Xie, Vidhi Jain, Jonathan Francis, Jay Patrikar, Nikhil Keetha, Seungchan Kim, Yaqi Xie, Tianyi Zhang, Zhibo Zhao, et al. Toward general-purpose robots via foundation models: A survey and meta-analysis. *arXiv preprint arXiv:2312.08782*, 2023.

[HYW14] Cheng-An Hou, Min-Chun Yang, and Yu-Chiang Frank Wang. Domain adaptive self-taught learning for heterogeneous face recognition. In *2014 22nd International Conference on Pattern Recognition*, pages 3068–3073. IEEE, 2014.

[HZ93] Geoffrey E. Hinton and Richard S. Zemel. Autoencoders, minimum description length and helmholtz free energy. In *NIPS*, 1993.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE*

conference on computer vision and pattern recognition, pages 770–778, 2016.

[JLWS19] Shaowei Jin, Huaping Liu, Bowen Wang, and Fuchun Sun. Open-enviroment robotic acoustic perception for object recognition. *Frontiers in Neurorobotics*, 13:96, 2019.

[KANW17] Matthias Kerzel, Moaaz Ali, Hwei Geok Ng, and Stefan Wermter. Haptic material classification with a multi-channel neural network. In *Neural Networks (IJCNN), 2017 International Joint Conference on*, pages 439–446. IEEE, 2017.

[Kat25] David Katz. 1989 the world of touch. *LEA, Hillsdale, NJ*, 1925.

[KB15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, may 2015.

[KBT10] Thomas Koelewijn, Adelbert Bronkhorst, and Jan Theeuwes. Attention and the multiple stages of multisensory integration: A review of audiovisual studies. *Acta psychologica*, 134(3):372–384, 2010.

[KGS+20] Kuno Kim, Yihong Gu, Jiaming Song, Shengjia Zhao, and Stefano Ermon. Domain adaptive imitation learning. In *International Conference on Machine Learning*, pages 5286–5295. PMLR, 2020.

[KL92] Roberta L Klatzky and Susan J Lederman. Stages of manual exploration in haptic object identification. *Perception & psychophysics*, 52(6):661–670, 1992.

[KR23] Seung-Chan Kim and Semin Ryu. Robotic kinesthesia: Estimating object geometry and material with robot's haptic senses. *IEEE Transactions on Haptics*, 2023.

[KSB+10] Koray Kavukcuoglu, Pierre Sermanet, Y-Lan Boureau, Karol Gregor, Michaël Mathieu, and Yann L Cun. Learning convolutional feature hierarchies for visual recognition. In *Advances in neural info. processing systems*, pages 1090–1098, 2010.

[KSG+19] Matthias Kerzel, Erik Strahl, Connor Gaede, Emil Gasanov, and Stefan Wermter. Neuro-robotic haptic object classification by active exploration on a novel dataset. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[KW13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[LBDC21] Weiyu Liu, Dhruva Bansal, Angel Andres Daruna, and Sonia Chernova. Learning instance-level n-ary semantic knowledge at scale for robots operating in everyday environments. In *Robotics: Science and Systems*, 2021.

[LBDC23] Weiyu Liu, Dhruva Bansal, Angel Daruna, and Sonia Chernova. Learning instance-level n-ary semantic knowledge at scale for robots operating in everyday environments. *Autonomous Robots*, pages 1–19, 2023.

[LBL19] Jet-Tsyn Lee, Danushka Bollegala, and Shan Luo. "Touching to see" and "seeing to feel": Robotic cross-modal sensory data generation for visual-tactile perception. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4276–4282. IEEE, 2019.

[LC09] Dermot Lynott and Louise Connell. Modality exclusivity norms for 423 object properties. *Behavior Research Methods*, 41(2):558–564, 2009.

[LCL19] Justin Lin, Roberto Calandra, and Sergey Levine. Learning to identify object instances by touch: Tactile recognition via multimodal matching. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2019.

[LCS07] Simon Lacey, Christine Campbell, and K Sathian. Vision and touch: multiple or multisensory representations of objects? *Perception*, 36(10):1513–1521, 2007.

[LdADSOS17] André Teixeira Lopes, Edilson de Aguiar, Alberto F De Souza, and Thiago Oliveira-Santos. Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. *Pattern Recognition*, 61:610–628, 2017.

[LK87] Susan J Lederman and Roberta L Klatzky. Hand movements: A window into haptic object recognition. *Cognitive psychology*, 19(3):342–368, 1987.

[LK93] Susan J Lederman and Roberta L Klatzky. Extracting object properties through haptic exploration. *Acta psychologica*, 84(1):29–40, 1993.

[LKS15] Ian Lenz, Ross A Knepper, and Ashutosh Saxena. Deepmpc: Learning deep latent features for model predictive control. In *Robotics: Science and Systems*, volume 10. Rome, Italy, 2015.

[LKS+20] Qiang Li, Oliver Kroemer, Zhe Su, Filipe Fernandes Veiga, Mohsen Kaboli, and Helge Joachim Ritter. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics*, 2020.

[LLC22] Yawen Liu, Shihan Lu, and Heather Culbertson. Texture classification by audio-tactile crossmodal congruence. In *2022 IEEE Haptics Symposium (HAPTICS)*, pages 1–7. IEEE, 2022.

[LLL+18] Yang Liu, Zhaoyang Lu, Jing Li, Chao Yao, and Yanzi Deng. Transferable feature representation for visible-to-infrared cross-dataset human action recognition. *Complexity*, 2018, 2018.

[Llo82] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[Loc00] Jeffrey J Lockman. A perception–action perspective on tool use development. *Child development*, 71(1):137–144, 2000.

[LS14] Simon Lacey and Krishnankutty Sathian. Visuo-haptic multisensory object recognition, categorization, and representation. *Frontiers in psychology*, 5:730, 2014.

[LSJJ19] Ruijun Liu, Yuqian Shi, Changjiang Ji, and Ming Jia. A survey of sentiment analysis based on transfer learning. *IEEE access*, 7:85401–85412, 2019.

[LSSVG23] Yifan Lu, Gurkirt Singh, Suman Saha, and Luc Van Gool. Exploiting instance-based mixed sampling via auxiliary source domain supervision for domain-adaptive action detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4145–4156, 2023.

[LWL+10] Li-Jia Li, Chong Wang, Yongwhan Lim, David M Blei, and Li Fei-Fei. Building and using a semantivisual image hierarchy. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3336–3343. IEEE, 2010.

[LWL+17] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[LYA+18] Shan Luo, Wenzhen Yuan, Edward Adelson, Anthony G Cohn, and Raul Fuentes. ViTac: Feature sharing between vision and tactile sensing for cloth texture recognition. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2722–2727. IEEE, 2018.

[LZAL17] Shan Luo, Leqi Zhu, Kaspar Althoefer, and Hongbin Liu. Knock-knock: Acoustic object recognition by using stacked denoising autoencoders. *Neurocomputing*, 267:18–24, 2017.

[LZD15] Liang Lin, Ruimao Zhang, and Xiaohua Duan. Adaptive scene category discovery with generative learning and compositional sampling. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(2):251–260, 2015.

[LZZ+22] Hao Li, Yizhi Zhang, Junzhe Zhu, Shaoxiong Wang, Michelle A. Lee, Huazhe Xu, Edward H. Adelson, Li Fei-Fei, Ruohan Gao, and Jiajun Wu. See, hear, and feel: Smart sensory fusion for robotic manipulation. In *Conference on Robot Learning (CoRL)*, volume 205, pages 1368–1378, Auckland, New Zealand, dec 2022. Proceedings of Machine Learning Research (PMLR).

[McC89] John McCarthy. Artificial intelligence, logic and formalizing common sense. In *Philosophical logic and artificial intelligence*, pages 161–190. Springer, 1989.

[MFHH22] Kristína Malinovská, Igor Farkaš, Jana Harvanová, and Matej Hoffmann. A connectionist model of associating proprioceptive and tactile modalities in a humanoid robot. In *2022 IEEE International*

Conference on Development and Learning (ICDL), pages 336–342. IEEE, 2022.

[MKK+18] Zak Murez, Soheil Kolouri, David Kriegman, Ravi Ramamoorthi, and Kyungnam Kim. Image to image translation for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4500–4509, 2018.

[MM17] Janki Mehta and Angshul Majumdar. Rodeo: robust de-aliasing autoencoder for real-time medical image reconstruction. *Pattern Recognition*, 63:499–510, 2017.

[MMR09] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Multiple source adaptation and the rényi divergence. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 367–374. AUAI Press, 2009.

[MRL+15] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, pages 18–25, 2015.

[MSPGiN16] Alberto Montes, Amaia Salvador, Santiago Pascual, and Xavier Giro-i Nieto. Temporal activity detection in untrimmed videos with recurrent neural networks. *arXiv preprint arXiv:1608.08128*, 2016.

[NANI11] Tomoaki Nakamura, Takaya Araki, Takayuki Nagai, and Naoto Iwahashi. Grounding of word meanings in latent dirichlet allocation-based multimodal concepts. *Advanced Robotics*, 25(17):2189–2206, 2011.

[NGTJJ22] Nicolás Navarro-Guerrero, Sibel Toprak, Josip Josifovski, and Lorenzo Jamone. Visuo-haptic object perception for robots: An overview. *arXiv preprint arXiv:2203.11544*, 2022.

[NLWS20] Shuteng Niu, Yongxin Liu, Jian Wang, and Houbing Song. A decade survey of transfer learning (2010–2020). *IEEE Transactions on Artificial Intelligence*, 1(2):151–166, 2020.

[NMS04] Lorenzo Natale, Giorgio Metta, and Giulio Sandini. Learning haptic representation of objects. In *International Conference on Intelligent Manipulation and Grasping*, 2004.

[NPOV15] Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P Vetrov. Tensorizing neural networks. In *Advances in Neural Information Processing Systems*, pages 442–450, 2015.

[OOF18] Ilyas Ozer, Zeynep Ozer, and Oguz Findik. Noise robust sound event classification with convolutional neural network. *Neurocomputing*, 272:505–512, 2018.

[Ose11] Ivan V Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[OSSP23] Poojan Oza, Vishwanath A Sindagi, Vibashan Vishnukumar Sharmini, and Vishal M Patel. Unsupervised domain adaptation of object detectors: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[PGGG+20] Francisco Pastor, Jorge García-González, Juan M Gandarias, Daniel Medina, Pau Closas, Alfonso J García-Cerezo, and Jesús M Gómez-de Gabriel. Bayesian and neural inference on lstm-based object recognition from tactile and kinesthetic information. *IEEE Robotics and Automation Letters*, 6(1):231–238, 2020.

[PGH+16] Lerrel Pinto, Dhiraj Gandhi, Yuanfeng Han, Yong-Lae Park, and Abhinav Gupta. The curious robot: Learning visual representations via physical interactions. In *Computer Vision–ECCV 2016: 14th*

European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pages 3–18. Springer, 2016.

[PGM+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019.

[Pic15] Karol J Piczak. Environmental sound classification with convolutional neural networks. In *Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on*, pages 1–6. IEEE, 2015.

[Pow99] Thomas G Power. *Play and exploration in children and animals*. Psychology Press, 1999.

[PSE12] Cesare V Parise, Charles Spence, and Marc O Ernst. When correlation implies causation in multisensory integration. *Current Biology*, 22(1):46–49, 2012.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[RDGF16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[RK19] Benjamin Richardson and Katherine Kuchenbecker. Improving haptic adjective recognition with unsupervised feature learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[RKH+21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, volume 139, pages 8748–8763. Proceedings of Machine Learning Research (PMLR), 2021.

[RN55] F Riesz and B Sz Nagy. Functional analysis, frederick ungar pub. *Co., New York*, 1955.

[RQD00] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1-3):19–41, 2000.

[Ruf84] Holly A Ruff. Infants' manipulative exploration of objects: Effects of age and object characteristics. *Developmental Psychology*, 20(1):9, 1984.

[SAP17] Hardik B Sailor, Dharmesh M Agrawal, and Hemant A Patil. Unsupervised filterbank learning using convolutional restricted boltzmann machine for environmental sound classification. *Proc. Interspeech 2017*, pages 3107–3111, 2017.

[SBS+11] Jivko Sinapov, Taylor Bergquist, Connor Schenck, Ugonna Ohiri, Shane Griffith, and Alexander Stoytchev. Interactive object recognition using proprioceptive and auditory feedback. *The International J. of Robotics Research*, 2011.

[SBS22] Priya Sundaresan, Suneel Belkhale, and Dorsa Sadigh. Learning visuo-haptic skewering strategies for robot-assisted feeding. In *6th Annual Conference on Robot Learning*, 2022.

[SF82] William Schiff and Emerson Foulke. *Tactual perception: a sourcebook*. Cambridge University Press, 1982.

[Sho17] Elaine S. Short. *Managing Multi-Party Social Dynamics for Socially Assistive Robotics*. PhD thesis, 2017. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-06-21.

[Sin13] Jivko Sinapov. *Behavior-grounded multi-sensory object perception and exploration by a humanoid robot*. PhD thesis, Iowa State University, 2013.

[SKSS16] Jivko Sinapov, Priyanka Khante, Maxwell Svetlik, and Peter Stone. Learning to order objects using haptic and proprioceptive exploratory behaviors. In *IJCAI*, pages 3462–3468, 2016.

[SLM00] Felicity Sapp, Kang Lee, and Darwin Muir. Three-year-olds' difficulty with the appearance–reality distinction: Is it real or is it apparent? *Developmental Psychology*, 36(5):547, 2000.

[SLZ+20] Amrita Sawhney, Steven Lee, Kevin Zhang, Manuela Veloso, and Oliver Kroemer. Playing with food: Learning food item representations through interactive exploration. In *International Symposium on Experimental Robotics*, pages 309–322. Springer, 2020.

[SLZ+21] Amrita Sawhney, Steven Lee, Kevin Zhang, Manuela Veloso, and Oliver Kroemer. Playing with food: Learning food item representations through interactive exploration. In *International Symposium on Experimental Robotics (ISER)*, volume 19 of *Springer Proceedings*

in Advanced Robotics, pages 309–322, La Valletta, Malta, Nov 2021. Springer.

[SMS15] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015.

[SS08] Ladan Shams and Aaron R Seitz. Benefits of multisensory learning. *Trends in cognitive sciences*, 12(11):411–417, 2008.

[SS10] Jivko Sinapov and Alexander Stoytchev. The boosting effect of exploratory behaviors. In *AAAI*, 2010.

[SSS+14a] Jivko Sinapov, Connor Schenck, Kerrick Staley, Vladimir Sukhoy, and Alexander Stoytchev. Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robotics and Autonomous Systems*, 62(5):632–645, may 2014.

[SSS14b] Jivko Sinapov, Connor Schenck, and Alexander Stoytchev. Learning relational object categories using behavioral exploration and multimodal perception. In *International Conference on Robotics and Automation (ICRA)*, pages 5691–5698, Hong Kong, China, may 2014. IEEE.

[SSS14c] Jivko Sinapov, Connor Schenck, and Alexander Stoytchev. Learning relational object categories using behavioral exploration and multimodal perception. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 5691–5698. IEEE, 2014.

[SSSS11] Jivko Sinapov, Vladimir Sukhoy, Ritika Sahai, and Alexander Stoytchev. Vibrotactile recognition and categorization of surfaces by a humanoid robot. *IEEE Transactions on Robotics*, 27(3):488–497, 2011.

[ST99] Dale M Stack and Mary Tsonis. Infants' haptic perception of texture in the presence and absence of visual cues. *British Journal of Developmental Psychology*, 17(1):97–110, 1999.

[SVL14] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[SWS09] Jivko Sinapov, Mark Wiemer, and Alexander Stoytchev. Interactive learning of the acoustic properties of household objects. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2518–2524. IEEE, 2009.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[Tal15] Durk Talsma. Predictive coding and multisensory integration: an attentional account of the multisensory mind. *Frontiers in Integrative Neuroscience*, 9:19, 2015.

[TB99] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

[TCV16] Devis Tuia and Gustau Camps-Valls. Kernel manifold alignment for domain adaptation. *PloS one*, 11(2):e0148655, 2016.

[TDSL00] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[TFS23] Gyan Tatiya, Jonathan Francis, and Jivko Sinapov. Transferring implicit knowledge of non-visual object properties across heteroge-

neous robot morphologies. In *Intl. Conf. on Robotics and Automation (ICRA)*. IEEE, 2023.

[TFS24] Gyan Tatiya, Jonathan Francis, and Jivko Sinapov. Cross-tool and cross-behavior perceptual knowledge transfer for grounded object recognition. *Under review for International Conference on Robotics and Automation (ICRA)*, 2024.

[TFW+24] Gyan Tatiya, Jonathan Francis, Ho-Hsiang Wu, Yonatan Bisk, and Jivko Sinapov. Mosaic: Learning unified multi-sensory object property representations for robot perception. *Under review for International Conference on Robotics and Automation (ICRA)*, 2024.

[THCHS19] Gyan Tatiya, Ramtin Hosseini, Michael C. Hughes, and Jivko Sinapov. Sensorimotor cross-behavior knowledge transfer for grounded category recognition. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2019.

[THHS20] Gyan Tatiya, Ramtin Hosseini, Michael Hughes, and Jivko Sinapov. A framework for sensorimotor cross-perception and cross-behavior knowledge transfer for object categorization. *Frontiers in Robotics and AI*, 7:137, 2020.

[TJNF05] Eduardo Torres-Jara, Lorenzo Natale, and Paul Fitzpatrick. Tapping into touch. 2005.

[TPS+17] Jesse Thomason, Aishwarya Padmakumar, Jivko Sinapov, Justin Hart, Peter Stone, and Raymond J. Mooney. Opportunistic active learning for grounding natural language descriptions. In *Proceedings of the 1st Annual Conference on Robot Learning (CoRL-17)*, volume 78, pages 67–76. Proceedings of Machine Learning Research, November 2017.

[TS19] Gyan Tatiya and Jivko Sinapov. Deep multi-sensory object category recognition using interactive behavioral exploration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 7872–7878. IEEE, 2019.

[TSES20] Gyan Tatiya, Yash Shukla, Michael Edegware, and Jivko Sinapov. Haptic knowledge transfer between heterogeneous robots using kernel manifold alignment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[TSL+21] Tasbolat Taunyazov, Luar Shui Song, Eugene Lim, Hian Hian See, David Lee, Benjamin CK Tee, and Harold Soh. Extended tactile perception: Vibration sensing through tools and grasped objects. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1755–1762. IEEE, 2021.

[TSMS18] Jesse Thomason, Jivko Sinapov, Raymond Mooney, and Peter Stone. Guiding exploratory behaviors for multi-modal grounding of linguistic descriptions. In *Proceedings of the 32nd Conference on Artificial Intelligence (AAAI-18)*, February 2018.

[TSS+16] Jesse Thomason, Jivko Sinapov, Maxwell Svetlik, Peter Stone, and Raymond J Mooney. Learning multi-modal grounded linguistic semantics by playing "I Spy". In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 3477–3483, 2016.

[TSS+20] Tasbolat Taunyazov, Weicong Sng, Hian Hian See, Brian Lim, Jethro Kuan, Abdul Fatir Ansari, Benjamin CK Tee, and Harold Soh. Event-driven visual-tactile sensing and learning for robots. *Robotics: Science and Systems*, 2020.

[TT22] Meenaxi Tank and Priyank Thakkar. Text summarization approaches under transfer learning and domain adaptation settings—a

survey. In *Computational Intelligence and Data Analytics: Proceedings of ICCIDA 2022*, pages 73–88. Springer, 2022.

[TVCO04a] Thomas Thesen, Jonas F. Vibell, Gemma A. Calvert, and Robert A. Osterbauer. Neuroimaging of multisensory processing in vision, audition, touch, and olfaction. *Cognitive processing*, 5(2):84–93, 2004.

[TVCÖ04b] Thomas Thesen, Jonas F Vibell, Gemma A Calvert, and Robert A Österbauer. Neuroimaging of multisensory processing in vision, audition, touch, and olfaction. *Cognitive Processing*, 5:84–93, 2004.

[TYT18] Tadahiro Taniguchi, Ryo Yoshino, and Toshiaki Takano. Multimodal hierarchical dirichlet process-based active perception by a robot. *Frontiers in neurorobotics*, 12:22, 2018.

[ur5] Ur5 figure. `https://www.nonead.com/en/intelligence_content/9667.html`. Accessed: 2022-09-10.

[VdODS13] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In *Advances in neural information processing systems*, pages 2643–2651, 2013.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WC13] Lu Wang and Claire Cardie. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1395–1405, 2013.

[WCH+21] Junhang Wei, Shaowei Cui, Jingyi Hu, Peng Hao, Shuo Wang, and Zheng Lou. Multimodal unknown surface material classification and

its application to physical reasoning. *IEEE Transactions on Industrial Informatics*, 18(7):4406–4416, 2021.

[WLL+22] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[WM11] Chang Wang and Sridhar Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *Twenty-second international joint conference on artificial intelligence*, 2011.

[WSKB22] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567, Virtual and Singapore, May 2022. IEEE.

[WWCM07] Teresa Wilcox, Rebecca Woods, Catherine Chapa, and Sarah McCurry. Multisensory exploration and object individuation in infancy. *Dev. Psy.*, 43(2):479, 2007.

[WWW+22] Yefei Wang, Kaili Wang, Yi Wang, Di Guo, Huaping Liu, and Fuchun Sun. Audio-visual grounding referring expression for robotic manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 9258–9264. IEEE, 2022.

[XHD16] Zhenqi Xu, Jiani Hu, and Weihong Deng. Recurrent convolutional neural network for video classification. In *Multimedia and Expo (ICME), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[XLZ+22] Pengwen Xiong, Junjie Liao, MengChu Zhou, Aiguo Song, and Peter X Liu. Deeply supervised subspace learning for cross-modal ma-

terial perception of known and unknown objects. *IEEE Transactions on Industrial Informatics*, 19(2):2259–2268, 2022.

[YHSS22] Lijin Yang, Yifei Huang, Yusuke Sugano, and Yoichi Sato. Interact before align: Leveraging cross-modal knowledge for domain adaptive action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14722–14732, 2022.

[YKT17] Yinchong Yang, Denis Krompass, and Volker Tresp. Tensor-train recurrent neural networks for video classification. *arXiv preprint arXiv:1707.01786*, 2017.

[YP17] Haiyan Yin and Sinno Pan. Knowledge transfer for deep reinforcement learning with hierarchical experience replay. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[YXC+22] Jianfei Yang, Yuecong Xu, Haozhi Cao, Han Zou, and Lihua Xie. Deep learning and transfer learning for device-free human activity recognition: A survey. *Journal of Automation and Intelligence*, 1(1):100007, 2022.

[YXL22] Fuchao Yu, Xianchao Xiu, and Yunhui Li. A survey on deep transfer learning and beyond. *Mathematics*, 10(19):3619, 2022.

[ZAS+23] Xiaohan Zhang, Saeid Amiri, Jivko Sinapov, Jesse Thomason, Peter Stone, and Shiqi Zhang. Multimodal embodied attribute learning by robots for object-centric action policies. *Autonomous Robots*, pages 1–24, 2023.

[ZFJ+16] Haitian Zheng, Lu Fang, Mengqi Ji, Matti Strese, Yigitcan Özer, and Eckehard Steinbach. Deep learning for surface material classification using haptic and visual information. *IEEE Transactions on Multimedia*, 18(12):2407–2416, 2016.

[ZFT+21] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. In *Conference on Robot Learning*, pages 726–747. PMLR, 2021.

[ZG22] Lei Zhang and Xinbo Gao. Transfer adaptation learning: A decade survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[Zha16] Hao Zhang. *Building and Leveraging Category Hierarchies for Large-scale Image Classification*. PhD thesis, Carnegie Mellon University Pittsburgh, PA, 2016.

[ZJM+22] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. In *Machine Learning for Healthcare Conference*, pages 2–25. PMLR, 2022.

[ZLQ+22] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[ZYW+15] Kun Zeng, Jun Yu, Ruxin Wang, Cuihua Li, and Dacheng Tao. Coupled deep autoencoder for single image super-resolution. *IEEE transactions on cybernetics*, 47(1):27–37, 2015.

[ZYZ+20] Sicheng Zhao, Xiangyu Yue, Shanghang Zhang, Bo Li, Han Zhao, Bichen Wu, Ravi Krishna, Joseph E Gonzalez, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, et al. A review of single-source deep unsupervised visual domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):473–493, 2020.

[ZZC+12] Jianhua Zhang, Jianwei Zhang, Shengyong Chen, Ying Hu, and Haojun Guan. Constructing dynamic category hierarchies for novel visual category discovery. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2122–2127. IEEE, 2012.

[ZZW+20] Yongchun Zhu, Fuzhen Zhuang, Jindong Wang, Guolin Ke, Jingwu Chen, Jiang Bian, Hui Xiong, and Qing He. Deep subdomain adaptation network for image classification. *IEEE transactions on neural networks and learning systems*, 32(4):1713–1722, 2020.