Universität Stuttgart

Multi-Timescale Representation Learning of Human and Robot Haptic Interactions

Von der Fakultät für Informatik, Elektrotechnik und Informationstechnik der Universität Stuttgart zur Erlangung der Würde eines Doktors der Naturwissenschaften (Dr. rer. nat.) genehmigte Abhandlung

Vorgelegt von

Benjamin A. Richardson

aus South Orange, N.J., U.S.A.

Hauptberichter:	Prof. Dr. rer. nat. Marc Toussaint
Mitberichter:	Dr. Katherine J. Kuchenbecker
2. Mitberichter:	Prof. Dr.-Ing. Eckehard Steinbach

Tag der mündlichen Prüfung: 21.12.2022

Institut für Parallele und Verteilte Systeme

2022

Acknowledgments

I would like to express my utmost gratitude to my supervisor Dr. Katherine J. Kuchenbecker for taking a chance and giving me the opportunity to pursue my Ph.D. in the Haptic Intelligence department at the Max Planck Institute for Intelligent Systems (MPI-IS) in Germany. I would also like to thank her for providing me with the freedom and support to pursue my ideas about human and robot understanding of haptic properties as well as her personal and professional guidance throughout these past years. Thank you very much Katherine!

I would also like to thank Dr. Marc Toussaint and Dr. Georg Martius for joining my IMPRS-IS thesis advisory committee at the start of my Ph.D. and providing guidance and feedback throughout. I would like to thank Dr. Toussaint for his assistance with both the enrollment and graduation processes at the University of Stuttgart, which would have been much more difficult without him. I would like to especially thank Dr. Martius for his close involvement and guidance throughout my final projects at MPI-IS.

I am very thankful for the wonderful department environment that I've gotten to enjoy over the past years, and for all the friends I've gained. I would like to thank Saekwang Nam and Nataliya Rokhmanova for the many discussions and runs through the woods, Paola Forte for the many discussions about all sorts of topics, Iris Andrussow for the throwing and discussions

(especially for CoRL), and Bernard Javot for his help with purchasing and manufacturing, his boundless enthusiasm, and meat clubs. Thank you to Alexis Block for her support and enthusiasm and to Dr. Yasemin Vardar for cooperating with me on a great journal paper. I would like to thank all my other colleagues for their support and for such a social and enjoyable atmosphere.

Besides my colleagues, I would like to thank the students I supervised, especially Sean Sdahl. I am grateful for his enthusiasm, curiosity, and proficiency well beyond his years.

Last but not least, I would like to thank my family for their support and encouragement over the last years. I am very thankful to my father for the many fruitful discussions about research and to both of my parents Tom and Karen for their advice and guidance about life in general. Thank you to my brother Adam for his discussions and constant enthusiasm about new ideas.

Ben Richardson

Contents

Abstract

The sense of touch is one of the most crucial components of the human sensory system. It allows us to safely and intelligently interact with the physical objects and environment around us. By simply touching or dexterously manipulating an object, we can quickly infer a multitude of its properties. For more than fifty years, researchers have studied how humans physically explore and form perceptual representations of objects. Some of these works proposed the paradigm through which human haptic exploration is presently understood: humans use a particular set of exploratory procedures to elicit specific semantic attributes from objects. Others have sought to understand how physically measured object properties correspond to human perception of semantic attributes. Few, however, have investigated how specific explorations are perceived. As robots become increasingly advanced and more ubiquitous in daily life, they are beginning to be equipped with haptic sensing capabilities and algorithms for processing and structuring haptic information. Traditional haptics research has so far strongly influenced the introduction of haptic sensation and perception into robots but has not proven sufficient to give robots the necessary tools to become intelligent autonomous agents. The work presented in this thesis seeks to understand how *single* and *sequential* haptic interactions are perceived by both humans and robots.

In our first study, we depart from the more traditional methods of studying human haptic perception and investigate how the physical sensations felt during single explorations are perceived by individual people. We treat interactions as probability distributions over a haptic feature space and train a model to predict how similarly a pair of surfaces is rated, predicting perceived similarity with a reasonable degree of accuracy. Our novel method also allows us to evaluate how individual people weigh different surface properties when they make perceptual judgments. The method is highly versatile and presents many opportunities for further studies into how humans form perceptual representations of specific explorations.

Our next body of work explores how to improve robotic haptic perception of single interactions. We use unsupervised feature-learning methods to derive powerful features from raw robot sensor data and classify robot explorations into numerous haptic semantic property labels that were assigned from human ratings. Additionally, we provide robots with more nuanced perception by learning to predict graded ratings of a subset of properties. Our methods outperform previous attempts that all used hand-crafted features, demonstrating the limitations of such traditional approaches.

To push robot haptic perception beyond evaluation of single explorations, our final work introduces and evaluates a method to give robots the ability to accumulate information over many sequential actions; our approach essentially takes advantage of object permanence by conditionally and recursively updating the representation of an object as it is sequentially explored. We implement our method on a robotic gripper platform that performs multiple exploratory procedures on each of many objects. As the robot explores objects with new procedures, it gains confidence in its internal representations and classification of object properties, thus moving closer to the marvelous haptic capabilities of humans and providing a solid foundation for future research in this domain.

Zusammenfassung

Der Tastsinn ist eine der wichtigsten Komponenten des menschlichen Sinnessystems. Er ermöglicht uns eine sichere und intelligente Interaktion mit den physischen Objekten und der Umwelt um uns herum. Durch einfaches Berühren oder geschicktes Manipulieren eines Objekts können wir schnell auf eine Vielzahl von dessen Eigenschaften schließen. Seit mehr als fünfzig Jahren untersuchen Forscher, wie der Mensch Objekte physisch erkundet und Wahrnehmungsrepräsentationen von Objekten bildet. In einigen dieser Arbeiten wurde das Paradigma vorgeschlagen, durch das die haptische Erkundung des Menschen heute verstanden wird: Der Mensch verwendet eine bestimmte Reihe von Erkundungsverfahren, um bestimmte semantische Eigenschaften von Objekten zu erfahren. Andere Arbeiten haben versucht zu verstehen, wie physikalisch gemessene Objekteigenschaften mit der menschlichen Wahrnehmung von semantischen Attributen übereinstimmen. Nur wenige haben jedoch untersucht, wie spezifische Erkundungen wahrgenommen werden. Da Roboter immer fortschrittlicher und im täglichen Leben allgegenwärtiger werden, gehören auch allmählich haptische Sensoren und Algorithmen zur Verarbeitung und Strukturierung haptischer Informationen zu ihrer Ausstattung. Die traditionelle Haptik-Forschung hat die Einführung haptischer Empfindungen und Wahrnehmungen in Robotern bisher stark beeinflusst, sich aber nicht als ausreichend erwiesen, um Robotern die not-

wendigen Werkzeuge zu geben, intelligente, autonome Agenten zu werden. Diese Arbeit versucht zu verstehen, wie einzelne und sequenzielle haptische Interaktionen sowohl von Menschen als auch von Robotern wahrgenommen werden.

In unserer ersten Studie weichen wir von den traditionelleren Methoden zur Untersuchung der menschlichen haptischen Wahrnehmung ab und untersuchen, wie die körperlichen Empfindungen während einzelner Erkundungen von einzelnen Personen wahrgenommen werden. Wir behandeln Interaktionen als Wahrscheinlichkeitsverteilungen über einen haptischen Merkmalsraum und trainieren ein Modell, das vorhersagt, wie ähnlich ein Paar von Oberflächen bewertet wird, wobei die wahrgenommene Ähnlichkeit mit einem angemessenen Grad an Genauigkeit vorhergesagt wird. Unsere einzigartige Methode ermöglicht es uns auch zu bewerten, wie einzelne Personen unterschiedliche Oberflächeneigenschaften gewichten, wenn sie Wahrnehmungsurteile fällen. Die Methode ist äußerst vielseitig und bietet viele Möglichkeiten für weitere Studien darüber, wie Menschen Wahrnehmungsrepräsentationen von spezifischen Erkundungen bilden.

Der nächste Teil unserer Arbeit beschäftigt sich mit der Verbesserung der haptischen Wahrnehmung von Robotern bei einzelnen Interaktionen. Wir verwenden unüberwachte Feature-Learning-Methoden, um leistungsstarke Merkmale aus unverarbeiteten Robotersensordaten abzuleiten und die Erkundungen des Roboters in zahlreiche semantische haptische Eigenschaftslabels zu klassifizieren, die durch menschliche Bewertungen zugewiesen wurden. Darüber hinaus ermöglichen wir Robotern eine nuanciertere Wahrnehmung, indem wir lernen, abgestufte Bewertungen für eine Untergruppe von Eigenschaften vorherzusagen. Unsere Methoden übertreffen frühere Versuche, die alle handverlesene Merkmale verwendeten, und zeigen die Grenzen herkömmlicher Ansätze auf.

Um die haptische Wahrnehmung von Robotern über die Bewertung einzelner Erkundungen hinaus zu erweitern, wird im abschließenden Teil dieser Arbeit eine Methode eingeführt und bewertet, die es Robotern ermöglicht, Informationen über viele aufeinanderfolgende Aktionen zu akkumulieren; unser Ansatz nutzt im Wesentlichen die Objektpermanenz aus, indem er

die Darstellung eines Objekts bedingt und rekursiv aktualisiert, während es nacheinander erkundet wird. Wir implementieren unsere Methode auf einer Robotergreiferplattform, die mehrere Erkundungsprozeduren an einzelnen Objekten durchführt. In dem Maße, in dem der Roboter Objekte mit neuen Verfahren erkundet, gewinnt er an Vertrauen in seine internen Darstellungen und die Klassifizierung von Objekteigenschaften. Mit dieser Methode nähern wir uns den vielseitigen haptischen Fähigkeiten des Menschen und schaffen eine solide Grundlage für zukünftige Forschung in diesem Bereich.

Chapter 1

Introduction

As the first sense to develop, touch plays a crucial role during infant development. As babies, we instinctively grasp objects [Twi65] and mouth them, developing mental representations of haptic and tactile sensations [SF05]. As our perceptual models of the world improve, we are able to deliberately interact with new objects and influence our environment to rapidly acquire haptic information and accomplish complex tasks. Despite their relevance to our development and daily life, haptic perception and the sense of touch have received relatively little attention in research compared to other senses, particularly vision and hearing. Undoubtedly, the fundamental nature of touch makes it generally more difficult to study. Whereas vision and audition are passive senses, touch is inherently active; we cause or impact our haptic sensations through direct action. Consequently, useful haptic perception requires integrating haptic sensing with specific interaction and the ability to construct complete, general representations from piecemeal information gathered over time.

Much of the existing work in the field of haptics has focused on high-level understanding of human haptic exploration and perception. Lederman and Klatzky [LK93] created a highly influential taxonomy of exploratory

procedures (EPs) that humans use to probe and extract information from objects and surfaces. Later experiments demonstrated that when humans perform a particular task, they vary which EPs they use to extract relevant object properties. Additional work has demonstrated that people vary the parameters of EPs under certain circumstances or combine multiple EPs to elicit either redundant or more complex information [LK09]. Studies of human haptic and tactile perception are typically approached from a similarly high level [ONY13]. Physical properties of surfaces or objects are measured, typically with an engineering tool such as a compression tester, and study participants are asked to perform some task to rank the objects or surfaces along a particular semantic dimension. The subjective ordering of objects is correlated with the measured physical properties, and the more strongly correlated properties are suggested as representative of the associated semantic property. Additionally, the results often suggest one or multiple principal semantic components, such as hardness or roughness, of haptic perception. This paradigm has broadly persisted as haptic and tactile perception has been introduced to computational and robotic systems. Apart from work done to improve robot grasping and manipulation using tactile sensing, efforts to develop computational or robotic haptic understanding have typically introduced a set of exploratory procedures, extracted specific features from sensor data that are informed by previous haptic literature, and used those features either to identify the objects or surfaces being touched or to identify their semantic properties [LBDL17].

Between these two primary approaches lies a middle ground that is relatively unexplored. Specifically, *how is the raw information that is felt by humans or acquired by sensors during single and sequential explorations mapped and generalized to broader haptic representations and properties?* As a person explores an object or a surface, how does information from millisecond interactions accumulate and form a integrated perception of that interaction? In Chapter 2, we introduce and evaluate a method to study how individual people combine and weigh signals acquired during haptic explorations to make perceptual judgements. This method accumulates haptic information from micro-interactions that occur over a series of exploratory procedures

performed by individual people and learns to predict how those individuals weigh those data when they make a perceptual judgement. How can we then train a robot to learn a similar haptic perceptual structure from its own exploration data without explicitly encoding information about haptic attributes? We demonstrate that unsupervised feature-learning methods can powerfully represent tactile data acquired from robot exploration by learning to predict human haptic semantic descriptions. These methods even outperform the traditional hand-crafted features that are assumed to be congruent with these semantic properties. Finally, can we provide a robot with the necessary haptic learning framework to interact autonomously with the world, accumulating data over long periods of interaction or even a lifetime and learning representations that capture relevant and general properties. We present a method that leverages powerful assumptions about object permanence to allow a robot to accumulate haptic information over multiple exploratory procedures and develop robust and general representations of objects that inherently capture valuable haptic properties.

1.1 Outline

This thesis addresses the modeling and development of systems that can compress extremely complex physical interactions into much simpler, generalizable representations. This work is presented in three main chapters, each of which begins with a detailed introduction and ends with a summary. The thesis is capped by a conclusions chapter that summarizes the main aims and achievements of this work and presents an outlook for its continuation in the future.

Modeling Human Haptic Perception from Unconstrained Surface Exploration

The first part of this thesis explores how to represent human haptic perception of surfaces. Touch perception is somewhat unusual in that sensation

and action are connected; rarely do we perceive anything haptically without consciously interacting with our environment. Thus, instead of trying to correlate general surface characteristics with average human perception of those surfaces, we explore a new way of understanding how perceptual judgements are made directly from the experiences in single interactions. Specifically, the work presented in this chapter seeks to model the relationship between (a) raw tactile and haptic signals during specific, individual explorations of pairs of surfaces and (b) the corresponding perception of haptic similarity between those two surfaces. We additionally seek to model how individual people weigh various surface properties in their perceptual judgements. This work has been accepted for publication in a peer-reviewed archival journal as

> B. A. Richardson*, Y. Vardar*, C. Wallraven, and K. J. Kuchenbecker. 'Learning to Feel Textures: Predicting Perceptual Similarities from Unconstrained Finger-Surface Interactions'. In: *IEEE Transactions on Haptics* (2022). *Equal contribution. Accepted.

Unsupervised Feature Learning for Predicting Human Perception of Haptic Properties

Although predefined characteristics of haptic signals have been proven to correlate reasonably well with human perception both in simple tasks such as roughness perception and complex tasks like surface similarity, they are still simplistic representations of more complex phenomena. The work presented in this chapter presents and evaluates an unsupervised dictionary learning method for extracting relevant features from raw signals captured by tactile sensors as a robot probed various objects. These learned features outperform traditional predefined features in the tasks of predicting human perception of various haptic and tactile properties; thus, the learned features are more general and descriptive. Additionally, we explore how robotic exploratory procedures can capture different information about haptic properties. This work has been published in a peer-reviewed archival conference proceedings

and a peer-reviewed archival journal as

B. A. Richardson and K. J. Kuchenbecker. 'Improving Haptic Adjective Recognition with Unsupervised Feature Learning'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2019, pp. 3804–3810

and

B. A. Richardson and K. J. Kuchenbecker. 'Learning to Predict Perceptual Distributions of Haptic Adjectives'. In: *Frontiers in Neurorobotics* 13 (2020), pp. 1–16.

Implicit Robot Learning of Haptic Properties from Sequential Interactions

The ability of humans to accumulate information over diverse successive exploratory procedures is crucial to forming comprehensive representations; individual exploratory procedures can elicit only partial information about any given object. If robots are going to operate autonomously across diverse environments, they will require similar abilities to accumulate and condense information over time. The work in this chapter introduces a method that assumes object permanence to allow robots to combine haptic data from multiple physical interactions with objects into a comprehensive model that contains information about many object properties. As the robot uses new exploratory procedures that provide new information, the object representations become more precise. Additionally, the robot only needs to perform a small number of exploratory procedures to get a general object representation. This work is in preparation for submission to a peer-reviewed archival conference as

B. A. Richardson, K. J. Kuchenbecker, and G. Martius. *A Sequential Group VAE for Robot Learning of Haptic Representations*. 2023. In preparation for submission to Robotics: Science and Systems.

1.2 Contributions

As indicated by his position as first or co-first author of all four included publications, the writer of this dissertation was personally responsible for the vast majority of the reported research. All co-authors are faculty members who shared ideas and guidance but did not carry out the reported research. No other students or postdoctoral researchers were involved in the reported research to the level of co-authorship.

Chapter 2

Modeling Human Haptic Perception from Unconstrained Surface Exploration

2.1 Introduction

When humans touch a surface with their fingers, spatio-temporal fingertip deformations activate several types of mechanoreceptors which send signals to connected tactile afferents, transmitting information to the central nervous system [JF09] that derives from and relates to physical surface properties such as friction, roughness, and elasticity. The skin deformations that occur depend on the material properties and geometry of the finger and the surface [ADT+17; MSB+14], normal force [DBE+16], and speed [WSL+13], and they can vary substantially even for the same person exploring the same

texture [MPC16]. Little is known about how the brain distills the information needed to evaluate textures from the combination of skin deformation and exploratory motion.

A common approach to determining the fundamental factors underpinning texture perception is conducting psychophysical experiments in which participants rank the similarity of surfaces or give ratings for their specific features (e.g., hardness, roughness). The results are typically analyzed by a dimensional reduction technique such as multidimensional scaling (MDS) or principal component analysis (PCA), which reveals a compact representation of a resultant perceptual space. In this perceptual space, similarly rated stimuli cluster and dissimilar stimuli separate [DWCK18; ONY13]. The current consensus in the literature [HBKY00; HFRY93; ONY13] is that tactile perception of surfaces can be compressed down to three to five perceptual dimensions, with axes roughly aligned with the rating dimensions of micro and macro roughness/smoothness, hardness/softness, stickiness/slipperiness, and coldness/warmness. The perceptual dimensions obtained for any particular study, however, depend highly on the selected set of surfaces.

Although the above approach gives a general understanding of how humans make perceptual judgments about surfaces, it is inadequate to explain the fundamental relationship between the tactile information elicited from the finger-surface interaction and the resulting perception. Revealing this relation is also crucial for many applications, such as robot perception [FL12; RK20; SBKS20], product design [ETA+13], and haptic rendering [CK17a; FKPC21; IVB19]. Despite the rich, complex, and unique information available from finger-surface interaction, the existing literature has generally forgone interaction-specific analysis in favor of general surface descriptors: most studies have sought correlations between the derived perceptual space and each surface's physical features (e.g., power spectral density, friction coefficient, average power, spectral centroid, and compressibility) measured in a controlled condition (fixed speed and force) [BK06; SHCR20; VWK19; YBCH07]. This approach oversimplifies the complex finger-surface interaction and its dependence on user exploration, as people modify their exploratory movements depending on both the perceptual task and scanned

texture to make better perceptual judgments [CSDB15]. More importantly, some studies [BK06; PCC19; SHCR20] overlooked the importance of finger properties during interaction and focused on surface properties measured via a tool or specific machinery when correlating with a perceptual space that was obtained via free finger exploration.

In this chapter, we aim to understand the fundamental relationship between the tactile information obtained from unconstrained finger-surface interaction and the corresponding human perception of that interaction. Specifically, we are interested in determining the extent to which common signal features (e.g., average power, spectral centroid, friction coefficient) calculated from free finger-surface interactions play a role in human perceptual judgments. Since the values of these features change with normal force and scanning speed [CK17a], relating them to perceptual judgments is not straightforward for free exploration. To address this challenge, we first propose a methodology that enables both the conversion of finger-surface interaction signals into a distribution of features and the calculation of the distances between feature distributions from different surfaces based on perceptual similarities rated by humans. Then, based on this methodology, we present general and participant-specific models that can predict the perceptual similarity of two surfaces from their corresponding finger interaction signals. The model parameters and predictions suggest relevant physical features and their weighted roles in human texture perception.

The results indicate that our model is able to predict the perceptual judgments for surface dissimilarities with moderate accuracy despite the great variety in the measured fingertip-surface interactions for the same surface, person, and interaction. We also found evidence that people weigh features differently, suggesting they employ individual mental models when distinguishing surfaces.

The work presented in this chapter has been accepted for publication in the *IEEE Transactions on Haptics* as:

B. A. Richardson*, Y. Vardar*, C. Wallraven, and K. J. Kuchenbecker. 'Learning to Feel Textures: Predicting Perceptual Similarities from Unconstrained Finger-Surface Interactions'. In: *IEEE Transactions on Haptics* (2022). *Equal contribution. Accepted.

2.2 Methods

We tested our approach on perceptual and interaction data collected from a previous study by Vardar et al. [VWK19]: human participants explored pairs of textures drawn from a set of ten and rated each pair's similarity while their finger-surface interaction data were recorded (Section 2.2.1). First, these signals were segmented into the two key exploratory procedures used by participants, tapping and sliding. Then, we partitioned these segmented physical signals into overlapping windows and extracted simple features from each window, resulting in feature distributions for each surface (Section 2.2.2). Finally, we projected these features into a low-dimensional space such that the distances between pair-wise feature distributions match the perceived surface-pair dissimilarities (Section 2.2.3); the models and optimization procedure were implemented in PyTorch (Section 2.2.4).

2.2.1 Data Collection

The data were collected via psychophysical experiments whose details were previously described [VWK19]. However, because the physical data presented in this work were not analyzed before, we summarize the details of the experiments here.

Seven women and three men with an average age of 28.5 years (SD: 4.14) participated in the experiments. The experimental protocol was approved by the Ethics Council of the Max Planck Society (HI protocol number: 18-09B). All participants gave written informed consent. Those who were not employed by the Max Planck Society were compensated at a rate of 8 EUR per hour.

Ten surfaces from the Penn Haptic Texture Toolkit [CLK14] were used

Figure 2.1: The ten surfaces used for the study.

as stimuli; the selected surfaces vary in material properties, resulting in a haptically diverse stimulus set (Figure 2.1). During the experiments, the participant sat in front of two surfaces (Figure 2.2(a)). A black divider was placed between the participant and the surfaces, and the participant wore noise-canceling headphones to mask auditory cues. These interventions ensured that the participants used only haptic cues during the experiment. Each surface was placed on top of a force/torque sensor (Nano 17 Titanium, ATI Inc.). The contact force vector, contact torque vector, and finger acceleration vector were measured during experiments. The force and torque data were collected by a data acquisition board (PCIe 6323, NI Inc.) with a sampling rate of 10 kHz. Two custom-built digital accelerometer boards (MPU-9250, Invensense Inc.) were placed on the index fingernails of both hands of the participant. The accelerometer data were collected via a micro-controller (ATmega32U4, Atmel Inc.) with a sampling rate of 4 kHz. The scene was recorded from above by a high-resolution camera (C920, Logitech Inc.)

In the experiment, each surface pair was placed on the force sensors by taping them to the holders at the edges. After this preparation, the participant was alerted with a sound. They then freely explored the two surfaces for 5 seconds using only their index fingers. Another sound indicated it was time to remove their fingers from the surfaces. Then, the participant

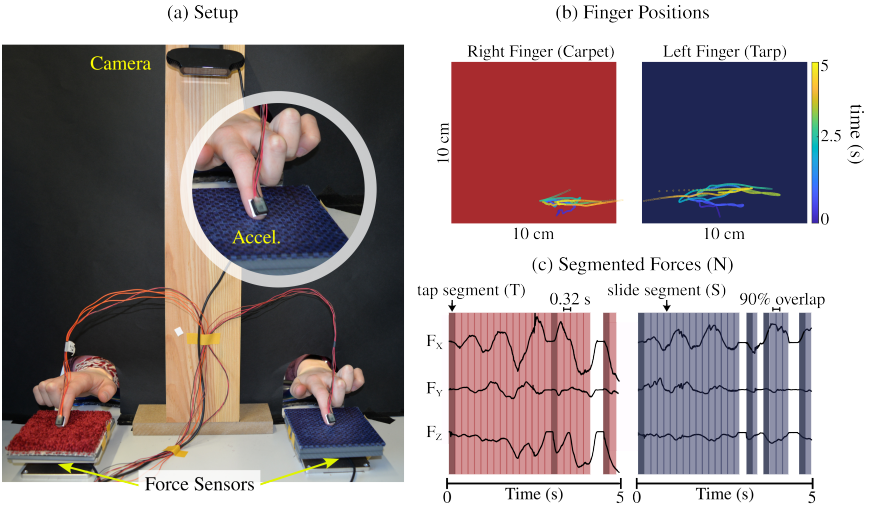


Figure 2.2: (a) The experimental setup for data collection. A participant touches a pair of surfaces. The finger-surface interaction data is collected via force sensors placed under each surface and accelerometers (indicated by "Accel.") attached to each index fingernail. A camera records the scene from above. (b) Example of calculated fingertip positions of one participant in one trial. The positions are calculated from the force-torque sensor data assuming each finger makes point contact with its surface. (c) Segmentation process. The force (and simultaneously collected acceleration) data are partitioned into tap and sliding regions based on the velocities of each finger. Each region consists of 320 samples, and each sliding segment overlaps with the former one 90%.

rated the similarity of the pair of surfaces using a nine-point scale. All 45 possible pairs of surfaces were presented twice, with each surface in the pair appearing once on the left and once on the right. Each participant touched the pairs in a different random order. Before each experiment, the participants were given instructions and asked to complete a training session. The training session included one very similar pair (stone tile and leather),

one very dissimilar pair (metal foil and carpet), and three random pairs. The very similar and dissimilar pairs were selected based on preliminary experimental results. In total there were 95 trials (5 training + (45 pairs × 2 locations)). Each participant completed the experiments in two sessions separated by a ten-minute break. The duration of the experiment was about 90 minutes.

2.2.2 Fingertip Interaction Features

As opposed to previous studies [ARSD17; BK06; KWT+13; SHCR20; VWK19; YBCH07], which represented textures as average features calculated from data collected in controlled conditions, we parse the interaction signals collected in each trial into smaller segments and then calculate features from them. As a result, we obtain a fine-grained distribution of features representing the interactions of each participant with each surface.

2.2.2.1 Segmentation

We compute two types of segments corresponding to the two key exploratory procedures used by participants: tapping and sliding. We define a tap as the moment when contact is initiated between the fingertip and surface, and we define a slide as a period of sustained tangential movement by the fingertip on the surface. To compute the tapping and sliding segments, we first transform the raw force-torque data into position and velocity (Figure 2.2(b)) by assuming each fingertip made point contact with the surface. The same technique was used in previous studies [BSB93; CK17b] to estimate the contact location of a fingertip or a tool on a surface. Before the position was computed, the force and torque signals were down-sampled to 2 kHz using MATLAB's *downsample* function. They were then low-pass filtered using a third-order Butterworth filter with a cut-off frequency of 20 Hz to capture hand motions [CK17a]. The fingertip velocity vectors were calculated by taking the time derivative of the fingertip position vectors. Given the filtered velocity signals, we use MATLAB's *findpeaks* function to select potential taps.

Only a peak that immediately follows a region of no contact (exactly zero velocity) is considered a tap peak.

We use the tap peaks to partition 2 kHz down-sampled force and acceleration signals into tap segments and sliding regions (Figure 2.2(c)). The tap segment is defined as a 320 sample (0.16 s) window starting from 19 samples before the peak. These values were determined by preliminary screening of the interaction data. Considerably shorter segments would not have captured all the relevant information from a tap interaction, whereas longer ones would have blended tap and sliding interaction data. After computing tap segments, all remaining non-zero velocity regions of the interaction are considered sliding regions. Segments are extracted from slide regions by scanning a 320 sample window (equal size to tap segments) directly after tap segments until the end of the sliding region. Each sliding segment was overlapped 90% with the previous one.

2.2.2.2 Feature Calculation

Select features were calculated from each segment of the 2 kHz signals to represent the three fundamental perceptual dimensions of surfaces: hardness/softness, roughness/smoothness, and friction (sometimes called stickiness/slipperiness) [HFRY93; YBCH07]. Features describing surface roughness/smoothness and friction were extracted from slide segments, whereas a feature representing hardness/softness was extracted from tap segments.

Our rationale behind choosing our particular set of features is as follows: previous studies [GBGB05; PDVG03] provide evidence that the roughness dimension is composed of both macro and micro roughness, and perceived roughness of the surfaces is related to the intensity and spectral content of the vibrations induced during fingertip sliding [FL12]. Hence, two metrics were selected to represent the roughness dimension during sliding segments: spectral centroid and vibration power. These two metrics were computed for both the force sensor and the fingernail-mounted accelerometer to enable comparisons between these distinct sources of information. The three-axis force and three-axis acceleration signals were first each combined into one

axis using the discrete Fourier transform 3-to-1 (DFT321) method [LRMK10]. The spectral centroid was computed by band-pass filtering the compressed signal between 5 Hz and 400 Hz and then taking the fast Fourier transform. For the vibration power, we further filtered the same signals between 20 Hz and 400 Hz and then calculated their average power.

The kinetic friction coefficient was selected as the metric to represent slipperiness. For each slide segment, the kinetic friction coefficient was calculated by fitting a Coulomb friction model to the unfiltered normal and tangential forces.

It has been previously shown that people can discriminate the hardness of a surface from the vibration that occurs after tapping on it with a tool [LaM00]. Because the spectral centroid of this vibration increases with the stiffness of the surface [CK17a], we chose it to represent hardness. Unlike the centroid described above, this spectral centroid was computed during tap segments from the force signal normal to the surface.

In summary, each sliding segment was represented by seven features: finger speed (v), normal force (F_n), kinetic friction coefficient (μ_k), and sliding power ($P_\cdot$) and spectral centroid ($C_\cdot$) calculated from force sensor ($\cdot_f$) and accelerometer ($\cdot_a$) data, whereas each tapping segment was represented by one feature: tap spectral centroid (C_{tap}) obtained from force sensor data. Therefore, the interaction data collected from one finger in each trial was reduced to the collection of seven + one different features calculated from each sliding or tapping segment of the entire interaction.

2.2.3 Modeling Framework

Our method aims to learn the relationship between the features extracted from the segments of raw tactile data and the perceptual similarity ratings provided by the participants. We do this by considering the set of segments extracted from the left- and right-handed interactions as two discrete probability distributions. We learn a mapping from feature space into a lower-dimensional embedding space such that the distances between the pairs of embedded distributions agree with the corresponding similarity

ratings. We will first introduce the problem definition and give a general overview of the entire modeling pipeline in Section 2.2.3.1. We then describe the details of the individual components of the pipeline. Figure 2.3 shows a summary of the full pipeline and a more detailed example of a single trial from the feature distributions to the distance computation in embedding space to the ranking of the computed distance relative to others.

2.2.3.1 Learning Problem

Let the set of all trials be denoted $\mathbf{S}$ and the set of corresponding similarity ratings be denoted $\mathbf{Y}$. Given a single trial $s \in \mathbf{S}$ with rating $y_s \in \mathbf{Y}$, the left- and right-handed interactions L_s and R_s with l_s and r_s segments, respectively, can be represented by matrices $\mathbf{X}_{L,s} \in \mathbb{R}^{l_s \times 8}$ and $\mathbf{X}_{R,s} \in \mathbb{R}^{r_s \times 8}$, where 8 is the total number of features. Each row of a matrix $\mathbf{X}$ contains the features calculated from a single segment of that interaction and can be written as

$$\mathbf{X}^{(i)} = \{v,\, F_n, \mu_k, P_f, P_a, C_f, C_a, C_{tap}\}, \tag{2.1}$$

where i denotes an arbitrary segment. If i is a sliding segment, the feature C_{tap} (last vector element) is assigned zero. Otherwise, the other seven features are assigned zero. Note that interaction matrices $\mathbf{X}$ can have different numbers of rows/segments. Examples of the eight columns of $\mathbf{X}_{L,s}$ and $\mathbf{X}_{R,s}$ from an arbitrary trial s are shown in the bottom left panel of Figure 2.3.

Additionally, to learn a compact representation of the features that more closely represents the human perceptual space, we define a mapping function $\Phi : \mathbb{R}^m \mapsto \mathbb{R}^n$ from the m-dimensional fingertip interaction feature space to an n-dimensional embedding space. We will describe this mapping function in greater detail in Section 2.2.3.3. This mapping function $\Phi(\mathbf{X})$ embeds each row of $\mathbf{X}$ as a unique point in $\mathbb{R}^n$. The projections of the left- and right-handed interactions $(\Phi(\mathbf{X}_{L,s}),\ \Phi(\mathbf{X}_{R,s}))$ can be represented as discrete

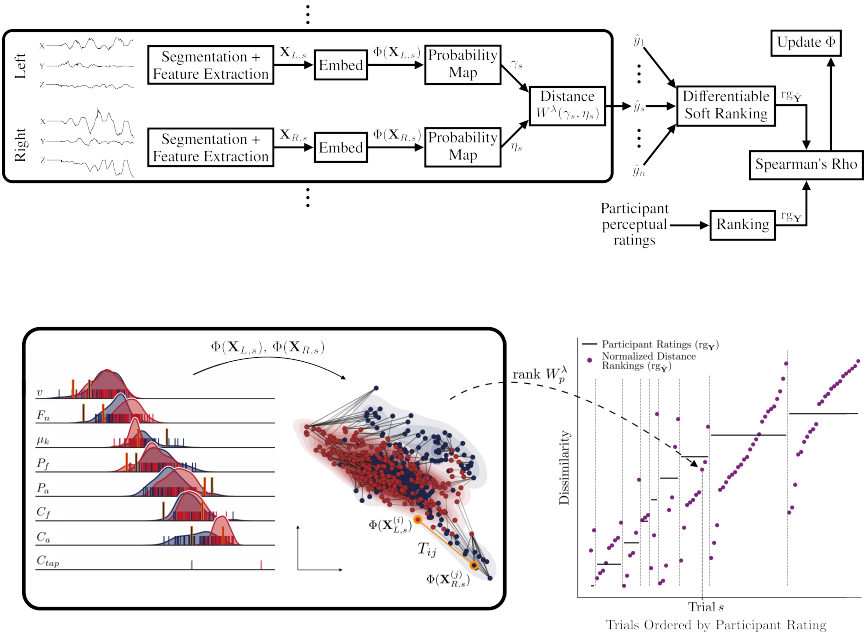


Figure 2.3: An arbitrary trial s is comprised of left- and right-handed interactions with different surfaces. The recorded 3D force (shown) and acceleration signals are parsed into many segments over time. Features are then extracted from each of these segments such that each segment is represented as a single point $X_{\cdot,s}^{(i)} \in \mathbf{X}_{\cdot,s}$ in multidimensional feature space (distributions and individual points shown in the bottom left panel). The two sets of points $\{\mathbf{X}_{L,s}, \mathbf{X}_{R,s}\}$ are then mapped via the function ϕ into an embedding space. The point sets are then converted to probability densities γ_s and η_s by assigning probability mass to each embedded point. The optimal transport distance $W_p^\lambda(\gamma_s, \eta_s)$ is computed between the left- and right-handed densities (bottom central panel). Finally, the resulting distance $\hat{y}_s$ is ranked relative to the distances of all other trials and compared to rankings of the human similarity ratings (in detail in bottom right panel). The function ϕ is optimized to maximize the Spearman's correlation between distances and rankings.

probability distributions γ_s and η_s, with

$$\gamma_s = \sum_{i=0}^{l_s} \mathbf{g}_i \delta_{\Phi_i(\mathbf{X}_{L,s})} \quad \text{and} \quad \eta_s = \sum_{i=0}^{r_s} \mathbf{h}_i \delta_{\Phi_i(\mathbf{X}_{R,s})}, \tag{2.2}$$

where $\mathbf{g}$ and $\mathbf{h}$ are non-negative vectors summing to 1 and $\delta_{\Phi_i(\cdot)}$ is the Dirac delta function centered at the point indicated by the i-th row of $\Phi(\mathbf{X})$. Then, $\hat{y}_s \in \hat{\mathbf{Y}} := \{\hat{y}_s \ \forall \ s \in \mathbf{S}\}$ is defined as the distance between probability distributions γ_s and η_s for the specific trial s. Specifically, we use the Wasserstein distance function, which we describe in Section 2.2.3.2.

Given this notation, the learning problem can generally be described as optimizing a parameterized mapping function Φ that maximizes the correlation between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$. Because Likert scales provide qualitative, ordinal data, we are specifically interested in maximizing the *rank correlation* between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$. This is called the Spearman's correlation, and it can be defined specifically for this problem as

$$\rho_{\text{sp}}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{\text{cov}(\text{rg}_{\hat{\mathbf{Y}}}, \text{rg}_{\mathbf{Y}})}{\sigma_{\text{rg}_{\hat{\mathbf{Y}}}} \sigma_{\text{rg}_{\mathbf{Y}}}}, \tag{2.3}$$

where $\text{rg}_{\hat{\mathbf{Y}}}$ and $\text{rg}_{\mathbf{Y}}$ are the rank variables of $\hat{\mathbf{Y}}$ and $\mathbf{Y}$, $\text{cov}(\text{rg}_{\hat{\mathbf{Y}}}, \text{rg}_{\mathbf{Y}})$ is the covariance of the rank variables, and $\sigma_{\text{rg}_{\hat{\mathbf{Y}}}}$ and $\sigma_{\text{rg}_{\mathbf{Y}}}$ are the standard deviations of the rank variables. We implement a differentiable ranking function that is described in Section 2.2.4.

2.2.3.2 Regularized Wasserstein Distance

To compute the distance between probability distributions, we use the p-Wasserstein distance, which is the solution to the traditional optimal transport problem and essentially measures the minimum cost of transporting the mass from one probability distribution to another in a metric space [Vil08]. Although there are other popular methods of measuring the similarity between probability distributions, such as the Kullback–Leibler (KL) and Jensen-Shannon divergence, we chose the Wasserstein metric because it

is symmetric (unlike KL-divergence), can be computed on distributions that do not share a support set, and has usable gradients over the entire support set [KPMR18]. This distance can be extremely costly to compute for both continuous and discrete distributions. Thus, we use the entropy-regularized p-Wasserstein distance, which approximates the true Wasserstein distance but admits a simpler solution that can be computed orders of magnitude faster using GPUs [Cut13]. Given two discrete measures γ and η with G and H (in our case l_s and r_s) support points, respectively, the discrete, entropy-regularized p-Wasserstein distance with regularization parameter λ is defined as

$$W_p^\lambda(\gamma,\eta)^p = \min_{T\geq 0} \operatorname{tr}(D^p T^\top) - \frac{1}{\lambda}h(T) \tag{2.4}$$

$$\text{s.t.} \quad T\mathbf{1} = \gamma, \quad T^\top \mathbf{1} = \eta,$$

$$\text{with} \quad h(T) = -\sum_{i=1}^{G}\sum_{j=1}^{H} T_{i,j}\log(T_{i,j}).$$

$D^p \in \mathbb{R}_+^{G\times H}$ is a matrix of distances with $D_{ij}^p = d(x_i,y_j)^p = ||x_i - y_j||_p^p$ and $T \in \mathbb{R}_+^{G\times H}$ is the discrete transport plan with T_{ij} the probability mass transported from γ_i to η_j [Cut13; FMS19]. $T\mathbf{1} = \gamma$ and $T^\top\mathbf{1} = \eta$ are the marginal constraints on T. The optimal T can be solved for using Sinkhorn's fixed point iteration. The black lines between points in the bottom central panel of Figure 2.3 display the elements T_{ij} of an example transport plan with a single element highlighted in orange. More information about optimal transport and the Wasserstein distance can be found in [Vil08], and specific details about the discrete Wasserstein distance with entropic regularization appear in [Cut13; FMS19].

Given this probability metric, $\hat{y}_s = W_p^\lambda(\gamma_s,\eta_s)$, where γ_s and η_s from Equation (2.2) are the discrete probability distributions defined over the embedding space for trial s.

2.2.3.3 Mapping Functions

We use two different types of mapping functions Φ in our experiments to embed the features extracted from the tactile data: affine maps and fully connected neural networks. These two choices represent two different levels of embedding complexity, with the affine maps having the simpler, more constrained embedding resulting from fewer degrees of freedom compared to the neural networks. For the affine maps,

$$\Phi_{\text{af}}(\mathbf{X}) = \theta \mathbf{X}^\top + \beta, \tag{2.5}$$

where $\theta \in \mathbb{R}^{m \times n}$ are the linear mapping parameters and $\beta \in \mathbb{R}^n$ are the biases.

For the neural network, we employ a single hidden-layer architecture with rectified linear unit (ReLU) activation functions. The general structure is then

$$\Phi_{\text{nn}}(\mathbf{X}) = \theta_{(o)} \cdot \mathrm{ReLU}(\theta_{(h)} \mathbf{X}^\top + \beta_{(h)}) + \beta_{(o)}, \tag{2.6}$$

where $\theta_{(h)} \in \mathbb{R}^{m \times k}$ and $\beta_{(h)} \in \mathbb{R}^k$ are the weights and biases of the hidden layer with output dimension k and $\theta_{(o)} \in \mathbb{R}^{k \times n}$ and $\beta_{(o)} \in \mathbb{R}^n$ are the weights and biases of the output layer with dimension n.

2.2.4 Implementation

All optimization of the parameters θ of Φ was performed using stochastic gradient descent and back-propagation with a loss function of

$$\mathcal{L}(\theta) = 1 - \rho_{\text{sp}}, \tag{2.7}$$

where ρ_{sp} is the Spearman's correlation from Equation (2.3).

One difficulty of implementing this loss function is that computing rank variables (e.g., $\mathrm{rg}_{\hat{\mathbf{Y}}}$ and $\mathrm{rg}_{\mathbf{Y}}$) is typically non-differentiable. To address this issue, we use a regularized, differentiable soft-rank function that approxi-

mates exact rankings [BTBD20]. The soft-rank function uses regularization to trade off between a more accurate ranking (smaller regularization) and a more strongly convex (and continuously differentiable) optimization (larger regularization).

The full optimization procedure was implemented in Python and PyTorch. The built-in Adam optimizer was used with a learning rate of 0.01 and default values for the remaining parameters. The ranking of the Wasserstein distances was performed using the soft-rank PyTorch implementation from Blondel et al. [BTBD20] with a regularization of 0.1, a value which provided a reasonable trade off between accuracy and convexity in preliminary experiments.

We used the regularized 1-Wasserstein distance (with the distance function $d(x_i, y_j)$ the L1 norm) and computed it using the auto-differentiable Sinkhorn implementation by Gabriel Peyré[1] with a regularization of 0.1 chosen from preliminary experiments. Additionally, the two weight vectors $\mathbf{g}$ and $\mathbf{h}$ from Equation (2.2) were defined such that probability mass was distributed uniformly across all points in an interaction. That is, for trial s with l_s and r_s segments, $\mathbf{g}_s = \mathbf{1}/l_s$ and $\mathbf{h}_s = \mathbf{1}/r_s$.

2.3 Modeling Procedure and Computational Experiments

Computational experiments were conducted to both evaluate the performance of the method and to learn more about the perceptual models of individual participants. As such, it was important to balance model interpretability with performance.

With this goal in mind, we first compared the performances of more complex, non-linear models with simpler affine models across a variety of embedding dimensions, demonstrating that simpler, more interpretable models are sufficient.

We then trained simple models to test the generalizability of the method to unseen participants and fine-tuned those general representations to indi-

[1] https://github.com/gpeyre/SinkhornAutoDiff

vidual participants. We analyzed and compared the model structures to try to understand differences between the general, "average" representations and the representational perceptual structures of the individual participants. Additionally, this analysis allows us to look at differences between individual participants.

Finally, we systematically evaluated the performance of much larger and deeper networks with various hyperparameter combinations to demonstrate that the smaller models are sufficient to capture meaningful information in the data.

2.3.1 Constructing General Models

General models were trained in two distinct ways. First, we ran a preliminary experiment to compare the performance of neural networks and affine maps as a function of the embedding dimension. We trained both types of models on data from all participants. We used a small neural network architecture of one hidden layer with eight nodes.

Second, we trained general affine map models on data from a subset of participants and evaluated those models on unseen participants. We did not perform this second training procedure with neural networks because the neural networks' slight edge in performance in the first experiment did not outweigh the greater interpretability of the affine maps. This finding is explained in greater detail in Section 2.4.1.

2.3.1.1 Model Comparison

For the first case, five-fold nested cross-validation was used to train preliminary comparison models. To form the folds, the samples from each participant were partitioned into five equally-sized, stratified groups, with each group having a roughly equal distribution over the ratings. Then, each of the five groups was added to a separate fold. A single fold was held out of the training process for testing, and a model was trained and evaluated on every possible three-one split of the remaining four folds. Thus, there were

four models trained for each hold-out. Each fold was held out as a test set, yielding a total of 20 trained models (4 per fold $\times$ 5 folds). For each training run the features were mean-centered for each participant independently using the data in the three training folds.

2.3.1.2 General Affine Models

For the training procedure of the general affine models, there were ten folds with each fold containing all the data from a different participant. The same process described above was performed, yielding a total of 90 trained models (9 per fold $\times$ 10 folds). In this case, the features for each participant were independently mean-centered using all their data.

In all cases, models were trained with a batch size of 180 for 200 epochs. The model state with the best validation performance over the 200 epochs was kept. Additionally, the loss was calculated on a per-participant basis and then averaged over participants. The participant-wise loss differs slightly from Equation (2.7) and can be formulated as

$$\mathcal{L}(\theta) = 1 - \frac{1}{|J|} \sum_{j \in J} \rho_{\text{sp}}(\hat{\mathbf{Y}}_j, \mathbf{Y}_j), \tag{2.8}$$

where J is the set of participants and $\hat{\mathbf{Y}}_j \subset \hat{\mathbf{Y}}$ and $\mathbf{Y}_j \subset \mathbf{Y}$ are the subsets of distances and ratings, respectively, for participant j. That is, the Spearman's correlation ρ_{sp} was calculated independently for each participant.

2.3.2 Participant-specific Modeling

To measure how the perceptual representations of individual participants differed from the generalized representations trained on other participants, we tuned general models to specific participants instead of training participant models from random initial conditions. Specifically, the participant-specific models for a particular participant were initialized using the best-performing (on the validation set) of the nine general models that were trained with that participant held out. To train the participant-specific models, a participant's

data were split into the same five folds used in the comparison model training. The models were trained for 100 epochs instead of 200 while the rest of the training, validation, and testing procedure remained the same. Features were mean-centered using the data in the training folds.

2.3.3 Evaluating More Complex Architectures

Initial testing with various model sizes and architectures suggested that smaller models performed comparably to larger ones. We performed additional, systematic experiments with networks that are larger and deeper than those used in our main experiments. Additionally, we tested these architectures with combinations of various nonlinear activation functions and regularization schemes and a much larger embedding space.

All the models were trained in the same way as those described above, except that they were trained for 1000 epochs with a learning rate of 0.001. All models have three hidden layers with 128 nodes each with an output embedding of ten dimensions. We tested combinations of three different activation functions (ReLU, Leaky ReLU, and Sigmoid), three different L2 regularization values (corresponding to the weight decay parameter in the Adam optimizer), and batchnorm and dropout layers. Each combination of hyperparameters was trained five times for all combinations of five folds.

2.4 Results

2.4.1 Model Type and Embedding Dimension

To measure the modeling performance as a function of model type and embedding dimension, we trained and evaluated neural networks and affine map models with outputs from one to five dimensions using five-fold nested cross-validation, as described in Section 2.3.1.1. Four models were trained for each testing fold, and of those four models, the one that performed the best on the validation set was then evaluated on the test set. Thus, for every full training procedure, five models were evaluated, one for each

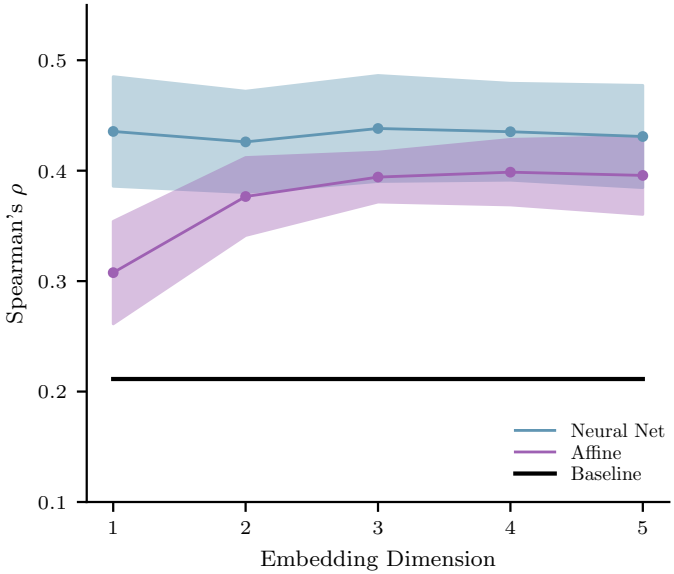


Figure 2.4: Mean and standard deviation of model performance vs. embedding dimension by model type. The neural networks all have one hidden layer with eight nodes. The baseline loss on the dataset with no feature mapping is indicated by the solid black line.

fold. To account for the random initialization of model parameters, the full modeling procedure described above was performed ten times for each embedding dimension and each model type. Thus, there are a total of 50 (5 folds $\times$ 10 random model seeds) evaluated models of each type (neural net and affine map) for each embedding dimension. The means and standard deviations of these test set evaluations are shown in Figure 2.4. To make the results clearer, we show $1 - \mathcal{L}(\theta)$ instead of $\mathcal{L}(\theta)$, which represents the Spearman correlation ρ between the predictions and psychophysical ratings. The baseline represents the loss on the original features with no mapping, i.e., $\Phi = \mathbf{1}$.

To demonstrate that our smaller models are sufficient, we also evaluated much larger, more complex models on the same learning task. The perfor-

mance of the much larger neural networks confirms that the small models are approaching the maximum performance given our particular assumptions and constraints. The average scores of the large models across all different hyperparameter combinations are shown for the training, validation, and testing sets in Table 2.1.

As can be seen in Figure 2.4, the ability to train an additional, low-dimensional embedding represents a considerable increase in performance for all values of embedding dimensionality. Furthermore, the neural network models marginally outperform the affine models, especially for a low embedding dimensionality of 1. However, there seems to be no additional benefit of adding further dimensions for neural network mappings. Given that affine maps in general are more interpretable compared to neural networks and that their performance saturates at an embedding dimension of three, we exclusively learned affine map models into three dimensions for our remaining experiments.

2.4.2 Generalizability

To test the generalizability of the modeling method to unseen participants, we trained affine maps into three dimensions on a subset of participants and evaluated them on unseen participants, as described in Section 2.3.1.2. Again, we analyze the performance of the best models by evaluating only the best validation model on the associated test fold (remember, each fold is a single participant). However, evaluating only the top-performing models could introduce bias if particular validation sets were always modeled more accurately than others. Thus, we also measure the ensemble performance of all the models trained for each test fold. Specifically, we compute $\hat{\mathbf{Y}}$ for each of the nine models, normalize each $\hat{\mathbf{Y}}$ so that all distances are between 0 and 1, take the average across all $\hat{\mathbf{Y}}$, and then compute the Spearman's correlation between the averaged distances and the corresponding similarity ratings.

As above, we repeat the full modeling process ten times to account for randomness in the initial model parameters. The mean performances of the

Table 2.1: Average Spearman's correlations for each hyperparameter combination.

Batchnorm	Dropout	Weight Decay	Activation	Spearman's		
				Training	*Validation*	*Testing*
True	True	0.0	ReLU	0.787	0.485	0.386
			Leaky ReLU	0.779	0.490	0.353
			Sigmoid	0.515	0.480	0.379
		0.1	ReLU	0.770	0.522	0.354
			Leaky ReLU	0.747	0.507	0.363
			Sigmoid	0.327	0.410	0.278
		0.3	ReLU	0.799	0.520	0.374
			Leaky ReLU	0.757	0.511	0.372
			Sigmoid	0.322	0.400	0.285
	False	0.0	ReLU	0.779	0.490	0.367
			Leaky ReLU	0.798	0.470	0.367
			Sigmoid	0.530	0.471	0.373
		0.1	ReLU	0.746	0.499	0.372
			Leaky ReLU	0.728	0.507	0.364
			Sigmoid	0.334	0.426	0.288
		0.3	ReLU	0.751	0.509	0.375
			Leaky ReLU	0.760	0.511	0.361
			Sigmoid	0.310	0.408	0.270
False	True	0.0	ReLU	0.664	0.465	0.360
			Leaky ReLU	0.678	0.470	0.349
			Sigmoid	0.488	0.452	0.356
		0.1	ReLU	0.694	0.490	0.358
			Leaky ReLU	0.787	0.497	0.356
			Sigmoid	0.324	0.411	0.282
		0.3	ReLU	0.763	0.505	0.366
			Leaky ReLU	0.754	0.537	0.374
			Sigmoid	0.322	0.427	0.279
	False	0.0	ReLU	0.801	0.478	0.346
			Leaky ReLU	0.723	0.495	0.360
			Sigmoid	0.492	0.473	0.379
		0.1	ReLU	0.796	0.485	0.370
			Leaky ReLU	0.754	0.488	0.335
			Sigmoid	0.335	0.406	0.292
		0.3	ReLU	0.745	0.488	0.365
			Leaky ReLU	0.743	0.499	0.352
			Sigmoid	0.314	0.413	0.272

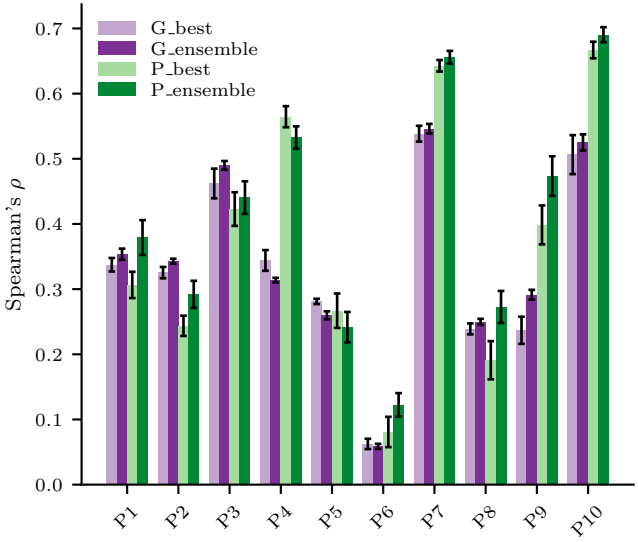


Figure 2.5: Means and standard errors of the best general (G_best), general ensemble (G_ensemble), best participant-specific (P_best), and participant-specific ensemble (P_ensemble) models.

best validation models (G_best) and the ensembles (G_ensemble) are shown in Figure 2.5, with error bars indicating the standard error of the mean.

Although the generalization performance differs substantially by participant, the average performance across all participants is very similar to the performance of the 3D affine model. Additionally, there is little change in performance between the best and ensemble predictions. Participants 3, 7, and 10 are modeled fairly well, whereas participant 6 is almost completely unpredictable. This finding suggests that much of the information about how most participants rated similarity is either not captured by the model or not contained in the data at all.

2.4.3 Participant-specific Model Tuning

From each of the ten randomized runs, the best general affine model for each participant hold-out was used as the starting configuration to train participant-specific models. Using this method, we can make direct comparisons between the tuned models and the original general models. We measure performance in the same way as above, evaluating both the best validation model for each test fold and the ensemble predictions. The mean performances of the best tuned models (P_best) and the ensembles (P_ensemble) are shown side by side with the general model performance in Figure 2.5.

In general, there is an improvement in performance when the models are individually tuned to individual subjects, particularly for participants 4, 7, 9, and 10. The performance for participant 3 is still relatively good, although there is no increase in accuracy. While there is a minor improvement for participant 6, the performance is still particularly poor. Again, there is little difference between the accuracy of the best and the ensemble models in most cases.

2.4.4 Model Analysis for Perceptual Characterization

One method of analyzing a simple affine model is to project the original feature axes into the embedding space and measure the relative scales of the axes. Because the Wasserstein distance depends on the distances between points in the metric space, a feature axis with a larger scale contributes more to the overall Wasserstein distance than a feature axis with a smaller scale.

To compute the relative axis scales for a single model, the unit vector along each feature axis can be projected into the embedding space. The projected vector lengths can all be divided by the magnitude of the longest vector to scale them between zero and one. Different models can be compared by normalizing the projected vector lengths for all models. This process was performed for the general models trained with participant holdouts and for the models that were tuned to specific participants. Figure 2.6 shows the density estimates of the relative axis lengths by participant for the general

(purple) and participant-specific (green) models. We show the results for the ensemble of models as opposed to only the best performing models. These are the same models whose performance is plotted in dark purple and dark green in Figure 2.5.

There are some observable patterns across subjects and different modeling scales. Clearly, the tap spectral centroid (C_{tap}) is consistently one of the largest embedded feature dimensions, meaning it contributes more to the overall Wasserstein distance than other dimensions. Conversely, the average vibration powers measured from both the force sensor (P_f) and accelerometer (P_a) are the smallest feature dimensions. Thus, the average vibration power does not greatly contribute to the Wasserstein distance. Additionally, for both pairs of features that were computed from both sensors, the feature computed from the accelerometer is always smaller than the corresponding feature computed from force data.

Interestingly, the models seem to get less consistent when they are tuned. For many features, the spread (height of the densities) actually increases from the general to the tuned models. This trend is most clearly demonstrated by the friction coefficient (μ_k) and force sensor slide spectral centroid (C_f).

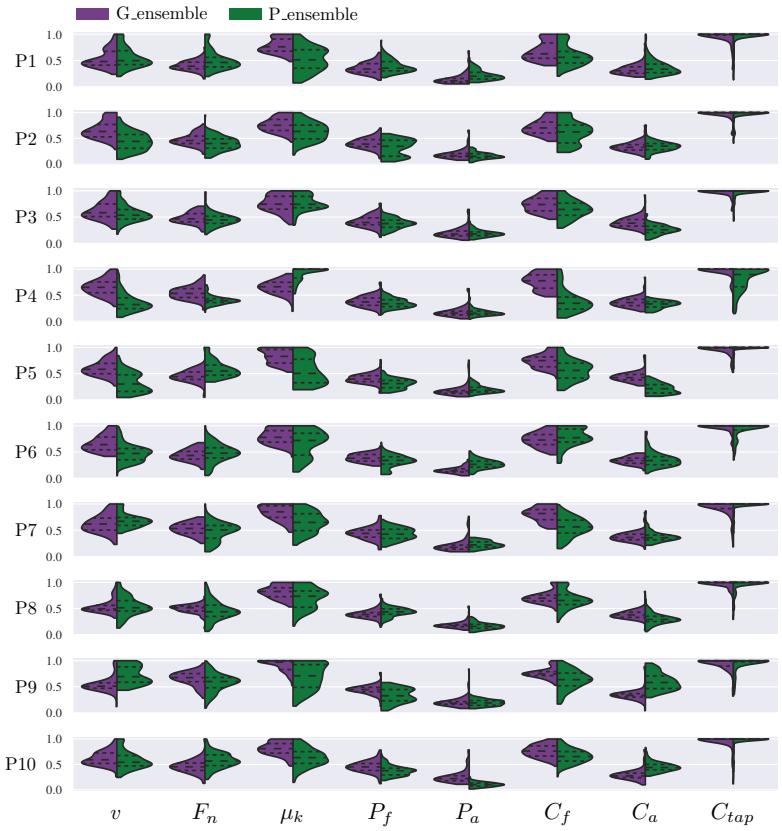


Figure 2.6: Distributions of normalized feature axis lengths by participant. Purple regions show distributions of normalized axis lengths for the trained general models by participant holdout. Dark green shows the distributions of normalized axis lengths for the fine-tuned participant models.

2.5 Discussion

The work presented in this chapter tried to solve the unique problem of predicting human perception from individual haptic experiences by aiming to understand the physical factors governing these perceptual judgments.

We presented a method that predicts the perceived similarity of two surfaces from the features extracted from the physical signals elicited during the interaction. The results demonstrate that this method somewhat works on both general and participant-specific levels. General representations learned on a subset of participants can partially predict the perceptual similarities of unseen participants with accuracies ranging from low to high depending on the participant. Analysis of the model structures provides a method to interpret the weights of different haptic properties in the perceptual similarity judgments of different people, albeit with limited confidence due to the model performance.

2.5.1 Complex Versus Simple Models

A key question about this method is whether a simple model is sufficient to capture the relationships between the tactile features and similarity ratings. The results shown in Figure 2.4 answer this question, demonstrating that simple affine models are comparable in performance to more complex neural networks despite having fewer than half the parameters; the additional experiments on deep models confirm this finding. The neural network models do perform marginally better, but the small improvement demonstrates that the method is not primarily limited by the model type, at least for this particular dataset and choice of features. The consistent performance as a function of the number of embedding dimensions, particularly for neural networks, provides additional evidence that the performance limitations are not due to the model architectures and that the Wasserstein metric has large representational capacity across a number of embedding dimensions.

Overall, the average performance reaches "only" levels of $\rho = 0.4$. One reason behind this moderate performance could be the significant noise in the participant ratings. The participant agreement can be measured by computing the Spearman's correlation for each pair of participants over all 90 trials and averaging, yielding an inter-rater agreement of 0.707 [VWK19]. Thus, the consistency of ratings across participants likely provides an approximate upper bound on the modeling performance. It is possible but highly

unlikely that all the rating noise can be explained by the data contained in each interaction, as humans are imperfect perceptual machines subject to inconsistency, distraction, and fatigue. Additionally, finding strong correlations between surface properties and human perception has been proven to be difficult. For example, Bergmann Tiest and Kappers [BK07] had subjects order a set of surfaces by roughness and found Spearman's correlations from 0.4 to 0.8 (depending on the subject) between the perceptual orderings and the physical roughness measures of the surfaces.

Another underlying reason for the moderate prediction performance of our model could be its use of selected features. Although we included the most common physical factors mentioned in the literature, ones that we did not consider (e.g., thermal conductance, spatial finger deformation, or skewness and kurtosis of the segments) may have significant effects on similarity judgments. It is also possible that human tactile processes do not estimate physical quantities but seek to estimate statistical variations in the tactile signals. This hypothesis has also been proposed for visual [FS19; PS00] and audio [MS11] senses. In a recent study [MT22], Metzger and Toscani trained a deep neural network with unsupervised learning to reconstruct vibratory signals elicited by human exploration of surfaces using a tool. They found that the learned latent space could classify different material categories similar to perceptual distances rated by human participants. If this is the case, it would be advantageous to construct a mapping from this latent space to the perceptual space without segmenting and calculating physical features from the original tactile signals. This work did not consider that option as we wanted to find relations between physical factors and perception.

2.5.2 Generalization and Specialization

By training models on subsets of participants and testing the performance on unseen participants, we demonstrate that our method can find an average perceptual representation across multiple people that can reasonably predict the perceptual similarity judgments of unseen participants. Tailoring

these general representations to individual participants suggests that the perceptions of each participant differ uniquely from the average but mostly can be captured by the tuned models.

Figure 2.5 demonstrates that the general models perform quite differently depending on the participant. They perform exceptionally well for participants 3, 7, and 10, but perform terribly for participant 6. This difference in performance likely indicates that there is some consistency across participants in how they judge similarity, but there are many differences that cannot be explained in an average model. However, it is possible that participants 3, 7, and 10 all employ a more similar rating strategy than the rest of the participants.

When the models are tuned, the accuracy improves most significantly for participants 4, 7, 9, and 10. Participant 3 still performs well even though the tuned models are not more accurate. This result provides evidence that, at least for these participants, a large part of their perceptual similarity judgments can be explained by the simple models and features that we used. It is particularly interesting that participants 4 and 9 improve quite clearly. It is possible that each of them relies primarily on the features that we included, but they treat them differently from all the other participants.

There are a variety of possible explanations for the comparatively worse performance on the other participants. For example, they may have relied more heavily on tactile signals that were not captured in our small feature set. As mentioned before, one feature in particular that was not included was the thermal conductivity of the surfaces. Temperature perception could have been a dominant cue in many cases, particularly for surface pairs that included aluminium [HJ06]. Other explanations could be that these participants used unique strategies to determine similarity or were inconsistent in applying their strategy. An example strategy could be to consider a surface pair very dissimilar if it differs dramatically in only a single dimension. An alternative strategy could be to consider a surface pair as similar unless it dramatically differs across multiple dimensions. Our method does not currently account for the use of different strategies, although we will discuss how this might be addressed in Section 2.6.

2.5.3 Inferring Perceptual Structure

The main benefit of using affine maps instead of neural networks is that their simplicity allows us to interpret the learned models and draw inferences about the participants' tactile perceptual representations. We focus on comparing the relative scales of the original feature axes projected into the learned embedding spaces. Despite the large amount of variance in perception that is not captured by our models, we propose that the larger features can be interpreted as more perceptually relevant. Given this assumption, it is immediately clear that, overall, the tap spectral centroid (C_{tap}) is a relevant feature. There are typically many fewer tap segments than slide segments, which means that much less probability mass is assigned to the tap segments overall. The large relative scale of C_{tap} demonstrates that despite the low mass, the tap segments provide unique information and are very important in modeling similarity. This holds true across all participants in both the general and tuned models. Considering the large variety in hardness of the selected surfaces (Figure 2.1) and that every trial started with a tap, it is indeed reasonable that hardness-relevant cues played an important role in similarity judgments.

Friction (μ_k) and the slide spectral centroid (C_f) are also relatively important compared to other features. Interestingly, a recent study [FKPC21] also found these features correlated with the two main axes in the perceptual space of fine textures created on friction modulation displays. Hence, the results suggest that friction and the slide spectral centroid could be relevant physical parameters for surface perception via direct fingertip touch.

On the other hand, both average vibration power features (P_f and P_a) are consistently the smallest of the features, with P_a being especially small. This means that these features did not contribute substantially to the distance between surface pairs. Thus, it is unlikely that the participants considered vibration power a relevant cue when measuring the similarity of the selected surfaces. Nonetheless, earlier studies [BK06; YBCH07] found that vibration power correlated with one of the main perceptual dimensions. A likely reason for this discrepancy is the difference in data collection. In both of

these earlier studies, the physical interaction data was collected via a tool, whereas we analyzed data that occurred during finger-surface interactions. The variety of selected surfaces and the range of motions used could also contribute to this discrepancy.

Interestingly, both features computed from the accelerometer (P_a and C_a) are typically smaller than their counterparts computed from the force sensor (P_f and C_f). This likely means that the force sensor mounted rigidly to the surface more accurately captured the fingertip-surface interaction than the accelerometer mounted to the fingernail; it is possible that the accelerometer data is even confounding. Due to the complex mechanical properties of the human finger and the fact that vibrations do not travel well from the fingerpad to the fingernail [SK21; WKWD06], the vibrations transmitted to the accelerometers likely differed substantially from those measured at the force sensors. Additionally, the limited sensitivity and noise susceptibility of the fingernail-mounted accelerometers compared to the force sensor could cause this discrepancy in sensor relevance.

For many features, the height of the densities (i.e., the spread of relative feature scales) actually increases from the general to the tuned models. However, we believe that the increase in spread is caused by the much smaller amount of data on which the tuned models are trained and the high variance in the data across folds. With more training examples for individual participants, the models would likely become more uniform and the feature densities narrower.

There is visible variability in the features that different participants relied on when making similarity judgments (Figure 2.6). For example, participant 4 seems to consider friction (μ_k) as highly relevant compared to the other participants. Additionally, the narrower densities of many features in the tuned models could explain why the performance increases dramatically from the general to those tuned models; participant 4 models similarity in a predictable way, but somewhat differently from all the other participants. Participant 9 also has tuned model distributions that differ substantially from the general models, particularly with regard to the velocity (v) and slide spectral centroid (C_f and C_a). On the other hand, participants 3, 7, and 10

have tuned model distributions more similar to the corresponding general model distributions, meaning that the general model was able to explain these participants' perceptual similarity judgments as well as possible with the given data.

Nonetheless, it is difficult to conclude much about the participants for which the modeling does not perform well. The predicted models of these participants could be accurate representations of their perceptual structure within the limitations of the used dataset. The poor prediction performance of their models could be explained by their inconsistent rating strategies among different surfaces. It is also possible that they relied on other tactile cues not presented in the data (e.g., thermal conductivity, stickiness, absorbency).

2.6 Summary

This chapter presented a new method for modeling and explaining how individual people make perceptual similarity judgements from specific haptic interactions. Unlike more traditional approaches, our method considers information taken from short windows of time during specific interactions instead of general surface characteristics. Because the method considers interaction specifics, it can potentially provide a deep level of explainability.

Although our method performed moderately for predicting general perceptual representations and better for some individual participants, this work has several limitations and sources of variability that we believe limited the potential performance; many of these factors could be individually addressed in future work and experiments.

The dataset has a limited number of participants who each made a limited number of surface comparisons. Likely, with more participants, more surfaces, and more surface comparisons, there would be less noise in the similarity ratings, and it would be possible to learn more predictive models. Additionally, the participants never compared two of the same surface. Comparing identical surfaces could provide valuable information about the consistency of user ratings as well as a powerful comparison that the model

might have been able to use to more strongly cluster similar surfaces.

We used a limited set of haptic features to represent the finger-surface interactions. While these features do correspond to primary tactile perceptual dimensions, it might be that secondary properties also contribute to similarity perception. As mentioned earlier, surface thermal conductivity was not included. There are additional vibration-related features, such as the spread or skewness of the frequency spectrum [SBKS20], that we did not include, and that could be included in future studies. Additionally, there is some evidence that not only temporal but also spatial features of surfaces play a role for perception during both static and dynamic exploration [WLH11; WSL+13]. As explained earlier, it is also possible that human similarity judgments do not rely on estimation of physical quantities but rather solely on statistical variations in the tactile signals [FS19; MT22]. In the future, this hypothesis can be tested by implementing unsupervised learning methodologies on unsegmented tactile signals elicited from finger-surface interactions.

Our method did not account for the possibility that people can use varying strategies to judge surface similarity. However, we believe that with minor changes this method could be extended to account for at least some strategic variance. The opportunity to provide strategic diversity lies in how probability mass is assigned to individual interaction segments, specifically how the vectors $\mathbf{g}$ and $\mathbf{h}$ are defined in Equation (2.2). As described in Section 2.2.3.2, we assigned mass uniformly across all segments. Because segments are sampled using discrete time windows, this means that low-velocity regions of the interactions automatically have a higher concentration of probability mass than high-velocity regions and thus contribute more to the Wasserstein distance. As a strategy, this could be described as participants weighing regions of low-velocity more heavily than others. However, normalizing the probability mass assignment by velocity (low-velocity segments have lower mass and high-velocity segments have higher mass) represents a different strategy where unique regions of the feature space are weighed independently of the velocity. These are just two examples, but there are many more strategies that can be captured by modifying the probability mass

assignment.

Overall, our method was able to model similarity judgments of many participants with moderate accuracy. The general model performances demonstrate that similarity judgments are extremely complex, and more information and method flexibility are necessary to capture judgments more accurately. However, even with our limited number of features, small model size, and simple mass-assignment strategy, we did find some consistent patterns explaining similarity judgments. By tuning models to specific participants, we found that the judgments can be explained more accurately in many cases. We believe these initial results are promising for the utility of this method to explain complex perceptual processes and how different people weigh various tactile features; future experiments could more precisely test how individual participants use different features. Moreover, given surface-finger interaction data or computed features from two different surfaces, our model can give a good approximation of the perceived similarity of these two surfaces without the need for time-intensive perception experiments.

In general, we believe our approach can help derive a deeper understanding of human tactile perception that can be applied across multiple domains. For example, by considering which tactile properties are relevant in an individual's texture preferences, recommender systems could suggest particular clothing or other textured objects. These properties could be captured by a haptic robot that learns what exploratory procedures most efficiently elicit the relevant data. Alternatively, haptic rendering systems could generate more realistic virtual textures by altering specific characteristics of the haptic output to better match the patterns seen in real textures over short time windows.

Chapter 3

Unsupervised Feature Learning for Predicting Human Perception of Haptic Properties

In Chapter 2, we investigated how humans perceive haptic similarity during specific interactions with different surfaces. The modeling method used a fine-grained approach, comparing distributions of features captured from short time segments of full interactions. We used a small set of predefined features, developed a model directly by comparing multiple pairs of interactions, and tried to incorporate mechanisms into the method to account for individualistic behavior. While this method is a useful tool for understanding the perceptual patterns of individual humans, it is not clear how to generalize it to robotic haptic perception, where a more general understanding of haptic interactions might be preferred.

The work in this chapter focuses on bridging the gap between robot and human tactile perception. We present two separate studies where we predict human adjective descriptions of objects from data gathered by a robot during various exploratory procedures. In the first, we predict a large number of binary adjective labels for many objects, and in the second, we predict scaled adjective ratings for those same objects. In both cases, we use unsupervised feature learning methods to extract compressed representations of interactions. The work presented in this chapter has been published as:

B. A. Richardson and K. J. Kuchenbecker. 'Improving Haptic Adjective Recognition with Unsupervised Feature Learning'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2019, pp. 3804–3810,

and

B. A. Richardson and K. J. Kuchenbecker. 'Learning to Predict Perceptual Distributions of Haptic Adjectives'. In: *Frontiers in Neurorobotics* 13 (2020), pp. 1–16.

3.1 Introduction

Much of modern machine learning focuses on modeling tasks for which inputs are sorted into discrete categories, such as image classification for visual data and speech recognition for audio data, e.g., [DDS+09; GBC16]. In the domain of haptics, machine learning is used to pursue similar classification tasks in which models aim to recognize specific objects [SLCD16; XLF13] or surfaces [BK17; FL12] from data gathered during haptic interactions. Typically, a model is trained on a large amount of tactile data that are manually labeled; given new tactile data, it can then predict the object or surface from which the data were captured. Although haptic recognition is an important task that humans perform well [KLM85], it is limited in its applications because the classification categories are constrained to a specific set, which restricts the experiences that can be recognized and

prevents generalization. For example, if a robot is trained to recognize specific textures or objects, it has no way to identify anything that it hasn't experienced before. Given the limitations of recognition tasks, learning higher-level semantic attributes that can be applied to new experiences can improve generalization; these attributes could include structural haptic cues, like size, or substance-related adjectives, like hardness and texture [KLR87].

Additionally, these machine-learning implementations are rarely trained directly on raw data; instead, they are usually applied to a set of representative features extracted from the data, such as the maximum value of a time-varying signal. In most cases, these features are meticulously designed according to the data and for the specific application [CMR+15; SS13; SSS15]. The main disadvantages of these hand-crafted features are that they require expertise to design, are developed for specific tasks, and depend on substantial assumptions about the relevant information in the data. Although Hoelscher et al. [HPH15] demonstrated that simplified hand-crafted features, such as a signal's mean, can slightly outperform the more complex features used by Chu et al. [CMR+15], this evaluation was performed on a simpler multiclass classification task.

Various methods exist for extracting representations from raw data without relying on carefully designed features. Neural networks can extract many levels of abstracted representations from data while making very few assumptions about the underlying structure [GBC16]. However, the learned representations typically depend to some extent on the specific training task. While research in transfer learning has shown that learned representations can be transferable to other tasks [Ben12; PY10], other methods can find underlying structure independently of any task. Autoencoders, for example, learn representations of data by compressing raw data into a lower-dimensional space and then uncompressing the middle layer to match the input data as closely as possible [HS06]. In the haptics domain, Madry et al. [MBKF14a] avoided designed features by using an unsupervised feature learning method called dictionary learning, but they tested their learned features only on the concrete tasks of object classification and grasp stability prediction.

This chapter presents work towards enabling a robot to understand common objects through autonomous tactile exploration. We apply unsupervised feature learning methods to haptic data gathered with rich tactile sensors to learn abstract tactile descriptions. Specifically, we apply K-SVD [AEB06] and Spatio-Temporal Hierarchical Matching Pursuit (ST-HMP) [MBKF14a] to extract features from the haptic data in the Penn Haptic Adjective Corpus (PHAC-2) created by Chu et al. [CMR+15]. We first detail how these algorithms are applied to the data to extract features. We then perform two experiments with these learned features.

In the first experiment, we compare the learned features directly to the hand-crafted features of Chu et al. [CMR+15] in multiple binary classification tasks. We use SVM to perform the classification. For the second experiment, we develop and evaluate a modified ordinal regression method to predict distributions over scaled haptic adjective ratings from the same learned features. In both cases, we measure the contribution of different exploratory actions and haptic sensor modalities to the learning and prediction of the adjectives.

3.2 Background

3.2.1 Learning high-level representations

Various work has focused on teaching robots abstract concepts and representations that can be extended to new environments and tasks. Instead of classifying specific objects, Sinapov et al. trained an algorithm on tactile and visual data to determine whether two separate observations are from the same object [SS13]. By training on limited comparisons, the algorithm accurately individuated many unseen test objects. This framework could incrementally train with new observations while simultaneously identifying new objects. Researchers have also used machine learning to accurately identify certain tactile events, such as slip [SHC+15; VVPH15]. These events are object independent and can be applied to a variety of haptic interactions. Chu et al. trained multiple SVMs on tactile and kinematic data

to identify which adjectives from a predefined set should be attributed to an object [CMR+15]. Zineb et al. trained a zero-shot learning algorithm to classify unseen objects by identifying haptic attributes of an object and then matching them to the object attribute labels [AGCC18].

3.2.2 Unsupervised learning

Although they are not frequently used in haptics, feature-learning algorithms have largely outperformed and replaced traditional methods of hand-crafted feature selection in computer vision. Researchers have used deep learning to demonstrate that these learned representations work well and can often be used in multiple learning tasks [Ben12].

Autoencoders are a type of artificial neural network that learns representations, or codings, of data by compressing data into a lower-dimensional space and then uncompressing the data to match the original input as closely as possible [HS06]. Because they try to reduce the error between the input and the reconstructed output, they learn the coding in an unsupervised manner and are task independent. Autoencoders have been shown to generate robust features [VLBM08].

Another method of generating features is unsupervised dictionary learning, in which dictionaries of basis vectors are learned. One well-known dictionary learning method that was inspired by the k-means clustering algorithm is K-SVD [AEB06]. In an unsupervised manner, K-SVD learns overcomplete dictionaries of basis vectors, or codewords, for sparse representation of raw data. Hierarchical Matching Pursuit (HMP) [BRF11] is an extension of K-SVD, learning dictionaries on small spatial patches of images; its inventors have shown the utility of this method in learning features for object, scene, and event recognition in images. ST-HMP extends HMP to tactile data by incorporating temporal information into the feature extraction [MBKF14a]. The features learned with both HMP and ST-HMP have been successfully used for multiple, distinct tasks.

3.2.3 Scaled Adjective Ratings

A standard way to capture richer information about human perception is to allow human raters to classify samples with discretization levels that are finer than a binary decision. One experimental method that yields this richer information is a sorting task. By allowing raters to sort materials by similarity and then analyzing the results using multidimensional scaling, Bergmann Tiest and Kappers [BK06] were able to compare perceived compressibility and roughness across many different materials. Hollins et al. [HFRY93] used a similar procedure to determine that hardness/softness and roughness/smoothness are primary, orthogonal dimensions of tactile perception, and that springiness, or the elasticity of a material, might correspond to an additional primary dimension. Using similar methodology, Hollins et al. [HBKY00] identified sticky/slippery as a third, less salient dimension of tactile perception. Another method is to have subjects rate tactile stimuli on a scale, which was the method used to gather the dataset that we will use for our experiments.

3.3 The PHAC-2 Dataset

In an effort to understand the relationship between raw tactile information and human perception of haptic interactions with objects, Chu et al. [CMR+15] collected the PHAC-2 dataset using two similar experiments. For the first, a robot equipped with state-of-the-art tactile sensors repeatedly touched 60 objects. For the second, human participants explored the same 60 objects in controlled conditions, providing multiple types of haptic descriptions for each object. The experiments were designed to provide the robot and humans with maximally similar experiences.

The 60 objects were selected from everyday items and constructed from common materials with the goal of providing a wide range of tactile experiences that would stay consistent throughout the study. To be included, an object had to be able to stand stably on a table and provide two approximately parallel, vertical, opposing surfaces with the same uniform texture.

All objects are between 1.5 and 8.0 cm thick and at least 10 cm tall to facilitate two-fingered exploration. The selected objects can be clustered into the following eight categories: 16 foam objects, 5 organic objects, 7 fabric objects, 13 plastic objects, 12 paper objects, 2 stone objects, 2 glass objects, and 3 metal objects.

3.3.1 Robot Exploration

As shown in Figure 3.1, a Willow Garage Personal Robot 2 (PR2) equipped with two BioTac tactile finger sensors (SynTouch LLC) was used to gather multi-modal haptic data. It performed an identical series of interactions with each of the 60 objects ten times, for a total of 600 trials. The BioTac, which is designed to imitate the sensing capabilities of a human fingertip, measures overall pressure, vibration, temperature, heat flow, and fingertip deflection. The robot performed the same four exploratory procedures (EPs) [LK93] for each trial in the following order: *Squeeze*, *Hold*, *Slow Slide*, and *Fast Slide*. These EPs were designed to imitate the frequently used human EPs of Pressure, Static Contact, and two speeds of Lateral Motion. Because humans prefer to determine distinct object properties using individual EPs [LK93], it is reasonable to expect that certain robot EPs might discriminate some object properties better than others. Each BioTac measured the absolute steady-state fluid pressure (P_{DC}), dynamic fluid pressure (P_{AC}), steady-state temperature (T_{DC}), heat flow (T_{AC}), and voltages on 19 spatially distributed impedance-measuring electrodes ($E_{1:19}$). P_{AC} was sampled at 2.2 kHz, and the other channels were sampled at 100 Hz.

To perform *Squeeze*, the PR2 slowly closed its gripper at constant velocity until the value of P_{DC} reached a predefined threshold, after which it slowly opened the gripper to the original position. During the *Hold* EP, the gripper was closed for ten seconds to a position that was halfway between the gripper distance at initial contact with the object and at the P_{DC} threshold during *Squeeze*. To perform *Slow Slide* and *Fast Slide*, the gripper was closed by 20% and 10%, respectively, of the *Squeeze* distance, moved downward by 5 cm at 1 and 2.5 cm/s, respectively, and then released. For a more detailed description

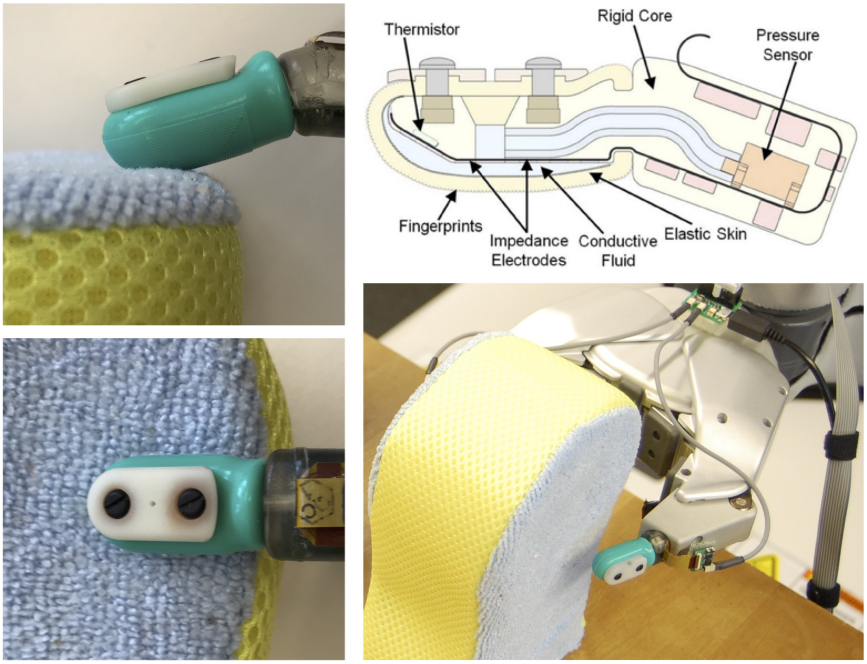


Figure 3.1: Detailed views of the BioTac-equipped PR2 hand interacting with the Blue Sponge object, and a diagram showing the internal components of the BioTac sensor.

of the robot experiment, please see Chu et al. [CMR+13; CMR+15].

3.3.2 Human-Participant Study

To capture how humans describe haptic interactions, thirty-six people took part in an experiment in which they haptically explored objects and provided descriptions. All procedures were approved by the University of Pennsylvania's Institutional Review Board under protocol #816464. Participants gave informed consent and were compensated $15 for participation. The cohort of participants contained 34 right-handed and 2 left-handed people, with 10 males and 26 females between the ages of 18 and 21 years. All partic-

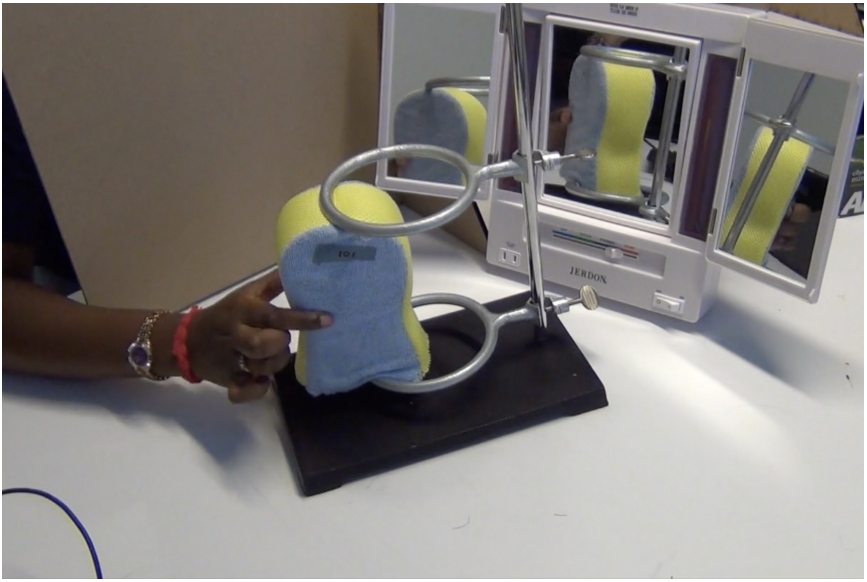


Figure 3.2: A human participant touching the Blue Sponge object during the experiment.

ipants were students at the University of Pennsylvania and had normally functioning arms and hands.

3.3.2.1 Experimental Procedure

The participant sat at a table at which the objects were presented. Individual objects were suspended from a ring stand above the table surface so that the participant could neither lift nor move the object. A large vertical panel prevented the participant from seeing their hand or the object. Additionally, the participant wore noise-cancellation headphones playing white noise to block ambient noise and any sound generated during interaction with the objects. To imitate the limitations of the PR2, the participant was instructed to use only their thumb and index finger from one hand. Additionally, they were allowed to use only a fixed set of exploratory procedures when probing

the objects: pressure, enclosure, static contact, and lateral movement. Figure 3.2 shows an image of a participant mid-experiment. Because Chu et al. [CMR+15] wanted to understand natural perceptually grounded language, participants were not coached in any way about how to define or apply the haptic adjectives used in the study.

To make the experiments more manageable, the 36 participants were split into three groups of 12, each of which was assigned a unique set of 20 objects (one third of the full set of 60 objects). The 12 participants from each group interacted only with the 20 objects assigned to their group. For each participant, the experiment was split into two stages. The first was used to familiarize the participant with the procedure, and the second was used to gather concrete data. In both cases, all 20 objects were presented in a random order, and the participant touched a compliant stress ball between objects to cleanse his or her haptic "palate." In the first stage, the participant freely described the feeling of each object to the experimenter. In the second stage, the participant was asked to rate each object on both binary and scaled ratings of pre-determined haptic adjectives while they were interacting with the object. The participant first selected the binary labels from a list of 25 haptic adjectives that were displayed in random order on a screen. Then the participant rated the object on a five-point scale for the ten basic haptic adjectives **hard**, **soft**, **rough**, **smooth**, **slippery**, **sticky**, **cold**, **warm**, **moldable**, and **springy**. Motivated by a lack of consensus in the literature, these scaled ratings were collected to test whether certain basic haptic adjectives have antonymous relationships and can be considered to lie along relevant tactile dimensions [GDM+10; PDVG03]. The 25 binary haptic adjectives were investigated in detail by Chu et al. [CMR+15]; however, the scaled ratings were not studied.

3.3.2.2 Scaled Adjective Ratings

Each of the 60 objects was rated on a scale that included 1 – "not at all (e.g., hard)", 2 – "slightly (hard)", 3 – "somewhat (hard)", 4 – "(hard)", and 5 – "very (hard)", for the ten basic haptic adjectives listed above. These

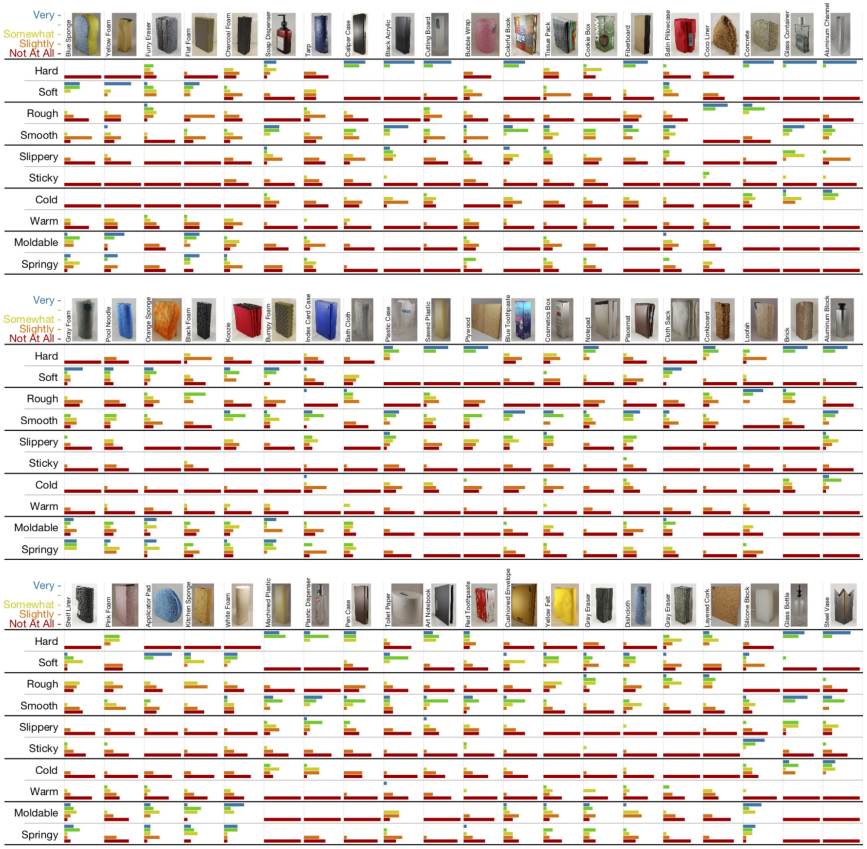


Figure 3.3: The 60 objects of the PHAC-2 dataset along with all the scaled adjective ratings given by participants. The objects are shown in the same three groups of 20 that were used in the study. Colored bar length is proportional to the number of responses the indicated rating received. At a glance, it is clear that **hard** and **soft** are antonyms, whereas **moldable** and **springy** seem to be synonymous.

adjectives are considered by some to comprise five basic antonym pairs that lie along relevant, and in some cases primary, dimensions of tactile

perception [HBKY00; ONY13]. The posited antonym pairs are **hard – soft**, **rough – smooth**, **slippery – sticky**, **cold – warm**, and **moldable – springy**. The full set of responses for all 60 objects is shown in Figure 3.3, including the names and small pictures of the objects.

3.4 Unsupervised Feature Learning

In this work we apply dictionary learning methods to the PHAC-2 dataset. We use dictionary learning algorithms because they are intuitive, fast to train, have a limited number of parameters, and have proven effective on tactile data. By learning features from such a diverse dataset and using them for multiple tasks, we demonstrate the effectiveness, versatility, and robustness of unsupervised feature learning methods on haptic data. We will compare the learned features to hand-crafted features on binary classification tasks for each haptic adjective, and we will use the same features to learn to predict the distributions of the scaled adjective ratings. In the following subsections we provide additional detail the unsupervised feature learning methods K-SVD and ST-HMP and how we use those to extract features from the raw PHAC-2 BioTac data.

3.4.1 K-SVD for Dictionary Learning

K-SVD [AEB06] is a well-known algorithm for learning a matrix dictionary composed of unit normal vectors, conventionally referred to as atoms. The learned dictionary is then used to represent data as sparse linear combinations of atoms. More precisely, given a data array $Y = [y_1, ..., y_M] \in \mathbb{R}^{n \times M}$ with M observations, each a vector of length n, K-SVD learns a K-atom dictionary $D = [d_1, ..., d_K] \in \mathbb{R}^{n \times K}$ and the corresponding matrix of sparse codes $X = [x_1, ..., x_M] \in \mathbb{R}^{K \times M}$ by solving the optimization problem

$$\min_{D,X} ||Y - DX||_F^2 \text{ subject to } ||x_m||_0 \leq T, \quad \text{for } m = 1, ..., M \ , \tag{3.1}$$

where $||\cdot||_F$ denotes the Frobenius norm, $||\cdot||_0$ denotes the L0 norm (which simply counts the nonzero entries), and T is the sparsity constraint, which upper-bounds the number of nonzero entries in each column of X.

K-SVD solves the above optimization problem using a greedy alternating iterative approach. In the first step of each iteration i, the dictionary $D^{(i-1)}$ is held constant and used to find X by solving the following M distinct problems

$$\min_{x_m} ||y_m - D^{(i-1)}x_m||_2^2 \text{ subject to } ||x_m||_0 \leq T, \quad \text{for } m = 1, ..., M \ . \tag{3.2}$$

In general, this minimization is NP-hard, so pursuit algorithms are typically used to find an approximate solution. K-SVD typically uses the greedy-style algorithm called orthogonal matching pursuit (OMP) [PRK93] to compute x_m. During each iteration, OMP calculates the residual $r_m^{(i)} = y_m - D^{(i-1)}x_m^{(i-1)}$, where $x_m^{(i-1)}$ is the most recent estimate, finds the atom index k that minimizes $||r_m^{(i)} - d_k^{(i-1)}||_2$, and updates the corresponding entries of the estimate to minimize the residual. This process repeats until T atoms are selected.

In the second step of each iteration, the dictionary and nonzero coefficients are updated simultaneously using SVD. Only a single atom d_k and its corresponding coefficients x^k, the kth row in X, are updated at a time. To prevent the introduction of new nonzero elements, SVD considers only the observations y_m that use d_k, and thus only the nonzero elements of x^k. After each atom has been updated, the new dictionary is used to compute the next sparse code matrix.

The minimization can be performed for a predefined number of iterations or until the reconstruction error reaches a predefined threshold. Once a dictionary is learned, it can be used to compute sparse code representations of new observations. These codes can be used directly or pooled to create more abstract features.

3.4.2 Spatio-Temporal Hierarchical Matching Pursuit

ST-HMP applies K-SVD to individual frames from a temporal sequence of spatially distributed tactile data, where each frame is a 2D tactile image [MBKF14a]. To construct the observation matrix Y used to train a dictionary, the individual tactile images are partitioned into small overlapping 2D spatial patches with size $p \times p$. The observations corresponding to each 2D patch from each image are treated as single elements y_i of Y. Thus, each tactile image corresponds to several columns of Y. Once a dictionary is learned, it can be used to compute sparse code representations of the patches from individual tactile images.

To extract features from the sequences of tactile images, the sparse codes are spatially and then temporally max pooled. To perform spatial pooling, the tactile image is split into spatial cells C_s, each containing a number of patches. To form the feature vector for a cell, the sparse codes representing each patch in a single cell, $\{x_i | i \in C_s\}$, are max pooled over each component x_i^m, where x_i^m is the m-th component of x_i. This pooling is done for all cells at varying scales, and the feature vectors for each cell are concatenated. A similar process is performed for temporal max pooling, where the feature vectors from tactile images within a temporal cell C_t are max pooled over each component. This pooling is also done at varying scales, resulting in a single feature vector for each tactile sequence.

3.4.3 Feature Extraction

Whereas Chu et al. used hand-crafted features to learn haptic adjectives, we use the unsupervised feature-learning methods described in the previous section to extract representations from the multi-modal haptic data. This is not a trivial problem given the diversity of tactile signals and variability in the lengths of interactions. The PHAC-2 database contains sequences of both scalar (P_{AC}, P_{DC}, T_{AC}, T_{DC}) and spatially distributed data ($E_{1:19}$) from four EPs of varying length, examples of which are shown in Figure 3.4. We use K-SVD with the addition of temporal max pooling to learn features

for the scalar signals and the ST-HMP algorithm to extract features from the electrode signals. First, dictionaries are learned on tactile sequences. These dictionaries are then used to compute sparse code matrices for the individual tactile sequences. Finally, the sparse codes are max pooled to create the feature sets. This section describes in greater detail how we adapt the existing methods to the tactile data from the PHAC-2 dataset.

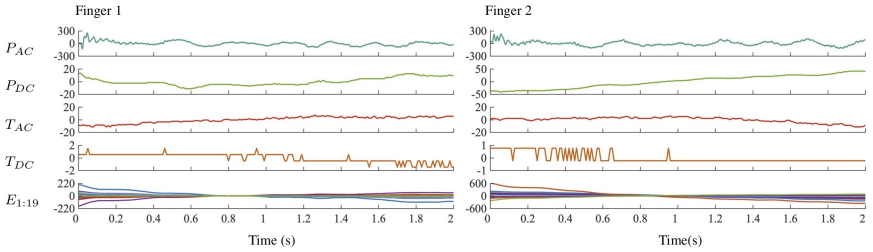


Figure 3.4: Scalar and electrode signals from the robot's two fingers over time during execution of the *Fast Slide* EP on the Blue Sponge object.

3.4.3.1 Dictionary Construction

A set of dictionaries was learned for each combination of the five sensor signal types and the four EPs for a total of $5 \times 4 = 20$ sets of dictionaries. Each dictionary was trained on data taken only from a single sensor signal type during a single EP. Six randomly selected trials per object comprising 60% of the total number of trials formed the training set for each dictionary.

3.4.3.2 Scalar Signal Feature Extraction

Motivated by the success of researchers who have used K-SVD to perform successful classification and forecasting from scalar time-series data [CZHL15; RDE16], we use K-SVD with temporal max pooling of sparse codes to extract features from the scalar BioTac data signals.

To construct the observation matrix $Y \in \mathbb{R}^{n \times M}$, we cut tactile sequences from the dictionary training set into M overlapping vectors of length n. Each

vector becomes an observation y_i in Y. The learned dictionary can then be used to compute sparse codes for individual sequences.

To extract features from a tactile sequence, we temporally max pool the sparse codes. A sequence is first split into temporal cells of multiple sizes. The sparse codes that represent the observations contained in or overlapping each cell are max pooled. Finally, the aggregated sparse codes from each cell are concatenated to form the feature vector for a single sequence. In our case, almost all of the sequences were partitioned into 16, 8, 4, 2, and 1 cells for a total of 31 temporal cells. *Fast Slide* sequences from P_{DC}, T_{AC}, and T_{DC} were only long enough to split into 8, 4, 2, and 1 cells.

Because each observation y_i contains time-series data, the dictionary atoms represent common temporal patterns of length n. When the value of n changes, the signal is filtered in different ways. If n is large, atoms will represent common low-frequency patterns in the data and will tend to filter out high frequencies. On the other hand, if n is small, atoms will represent common high-frequency patterns. To ensure that we captured many frequency components from the signals, we trained dictionaries for multiple values of n. For the 100 Hz signals (P_{DC}, T_{AC}, T_{DC}), the values of n were 10, 25, 50, and 100, which correspond to 0.1, 0.25, 0.5, and 1 s, respectively. The observations overlap by 5, 15, 40, and 90 frames, respectively. The larger overlaps were necessary to acquire enough observations from a single trial. For the 2.2 kHz P_{AC} signal, the values of n were 22, 44, 110, and 220, which correspond to 0.01, 0.02, 0.05, and 0.1 s, respectively. In each of these cases, the observations overlap by $0.5 \times n$ frames.

3.4.3.3 Electrode Array Feature Extraction

Because the electrode signals are spatially distributed on the BioTac sensor, we use ST-HMP to extract features from this data. Following work by Chebotar et al. [CHS+16], we arrange the 19 electrode measurements from each finger into a 7×3 rectangular array. The approximate relative positions of the electrodes are maintained in the array, and each of the two extra array values is interpolated from the electrodes surrounding it. An example of how the

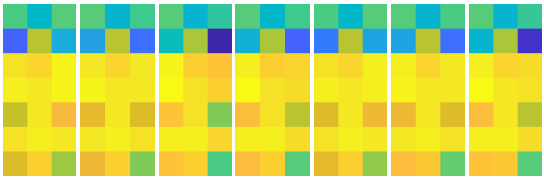


Figure 3.5: One of the electrode arrays from the signals shown in Figure 3.4. Darker colors in the array represent a lower measured voltage, which corresponds to more inward deformation of the finger surface.

electrode array values change over time during an EP is shown in Figure 3.5. Following the procedure described by Madry et al. [MBKF14a], the 7×3 arrays from the two BioTac fingers are then concatenated along one of the long edges to form a larger 7×6 array. There are complex arrangements that might more accurately represent the possible spatial relationships between electrodes, but we will not explore them here.

ST-HMP is performed on the sequences of 7×6 tactile images. A 3×3 patch is scanned over each training image and added to the observation matrix, which is in turn used to learn the dictionary. Sparse codes extracted from individual sequences are max pooled as described in Section 3.4.2. We divided each image into 9, 4, and 1 cells for a total of 14 spatial cells. The tactile sequences were divided into 16, 8, 4, 2, and 1 cells for a total of 31 temporal cells.

3.5 Experiment 1: Binary Adjective Classification

To evaluate the effectiveness of the learned features for separately classifying each binary adjective in the PHAC-2 database, we trained multiple classifiers that each use a linear support vector machine (SVM) with the L2 norm metric. The classifiers were used both to optimize various parameters of the feature-learning algorithms and to test the effectiveness of the learned

features on adjective classification. Dictionaries learned on scalar signals were optimized for observation length n, and all dictionaries were optimized over dictionary size K and sparsity constraint T.

3.5.1 Training and Testing Sets

To train adjective-specific classifiers, we used different training and test splits for each adjective. 10% rounded up of both the positively and negatively labeled objects were randomly selected to form the test set for each adjective classifier. The rest of the objects were put in the corresponding training set. All ten trials for each object were put in the same set to prevent the classifier from accidentally learning to classify objects instead of adjectives. So that we could directly compare our results, we used the same training and test sets as Chu et al. [CMR+15]. However, in an effort to avoid the most severely imbalanced sets, we analyzed only 19 of the 25 original adjectives, excluding those with fewer than three positively labeled objects.

3.5.2 Training the Classifier

To analyze the many sets of features extracted from each BioTac sensor for every EP, we trained multiple classifiers for each adjective. Cross-validation sets were created by randomly selecting 10% rounded up of both the positively and negatively labeled objects from the training set. Because many of the adjectives have imbalanced labels, we measure the performance of our classifiers with the F_1 score, which is calculated using the equation

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}, \tag{3.3}$$

where precision is the fraction of total positive classifier predictions that are correct, and recall is the fraction of positive examples that are correctly classified. The F_1 score was averaged over 100 randomly selected cross-validation sets to measure the performance of each classifier. Cross-validation was used to optimize dictionary learning and sparse code pooling parameters

as well as the SVM regularization parameter C and the decision threshold.

3.5.3 Results

To obtain the following results, we first trained a set of dictionaries for each EP-signal pair using cross-validation to optimize dictionary parameters as described above. Using the features extracted with the optimized dictionaries, we trained classifiers on the adjective-specific training sets, as described in Section 3.5.1, and then evaluated on the testing sets, which were held out of the entire training and optimization process. The F_1 score was again used to evaluate the classifier performance on the test sets.

3.5.3.1 Optimized Dictionary Parameters

Dictionary parameters were separately optimized for each EP-signal pair. Specifically, the optimal parameters were chosen to maximize the average F_1 score across all adjectives. In some cases, multiple sets of parameters performed equally well during cross-validation. In those cases, multiple dictionaries were learned for a single EP-signal pair. As an example, the optimized parameters for the EP *Fast Slide* are shown in Table 3.1. In all cases, the sparsity constraint T was 3 and 4 for the scalar and spatial signals, respectively. The observation length n was similar for each sensor signal type across EPs with the exception of P_{DC}, which had longer observation lengths for *Squeeze*, *Hold*, and *Slow Slide*.

Of note are the shorter observation lengths for P_{AC} and P_{DC}, which demonstrate that high-frequency components and transient information from the pressure sensors are more relevant than long-term trends. Conversely, high-frequency components are less relevant in the temperature sensors. Additionally, the smaller dictionary sizes for the temperature signals and electrodes demonstrate that fewer basis vectors are needed to accurately represent their variations and therefore that there is less variation in these data.

3.5.3.2 Performance by Exploratory Procedure

To compare our approach to that of Chu et al. [CMR+15], we first analyzed how well classifiers trained on data from individual EPs performed on each adjective. A classifier was trained for each EP using the concatenated features extracted from each sensor signal type. The results are shown in Table 3.2, with adjectives ordered by the number of positively labeled objects in the training set. We compared our F_1 scores directly to the maximum scores achieved between both the static and dynamic features used by Chu et al. The learned features substantially outperform the hand-crafted features for the majority of adjectives and EPs.

The mean over all of the individual F_1 scores is 0.673, which is a significant improvement over the results of both the static or dynamic classifiers from [CMR+15]. Averaging their best scores across the static and dynamic feature classifiers yields an F_1 score of 0.371, so our approach performs about 80% better. We also outperform their highest scoring combined-features classifier, the MKL classifier, which achieved a score of 0.620, as reported in the corrigendum to that article [CMR+16].

Table 3.1: Optimized dictionary parameters for the *Fast Slide* EP.

Signal	Observation Length n	Dictionary Size K	Sparsity Constraint T
P_{AC}	22	80	3
	44	80	3
P_{DC}	25	50	3
T_{AC}	100	50	3
T_{DC}	50	10	3
	100	10	3
$E_1 : E_{19}$	N/A	10	4

Table 3.2: F_1 scores across adjectives and EPs.

The symbols $\gg$ and $>$ represent relative increases in performance from Chu et al.'s hand-crafted features [CMR+15] of more than 0.15 and 0.05, respectively. $\approx$ represents a difference of no more than 0.05. $\ll$ and $<$ represent relative decreases in performance of more than 0.15 and 0.05, respectively. Darker shadings indicate higher performance.

	Squeeze	*Hold*	*Slow Slide*	*Fast Slide*	PE*
smooth	0.709 $\approx$	0.600 $>$	0.600 $\gg$	0.600 $\gg$	25
solid	1.000 $\gg$	1.000 $>$	1.000 $\approx$	1.000 $\gg$	22
squishy	0.938 $\gg$	0.929 $>$	0.821 $\gg$	0.822 $>$	21
compressible	0.938 $\gg$	0.983 $>$	0.966 $>$	0.929 $\gg$	20
hard	1.000 $\gg$	1.000 $\gg$	0.984 $\approx$	1.000 $\approx$	20
textured	0.551 $\gg$	0.444 $\gg$	0.581 $\gg$	0.444 $\gg$	16
soft	0.829 $>$	0.923 $\gg$	0.821 $\gg$	0.757 $\gg$	13
absorbent	0.889 $>$	0.750 $\gg$	0.684$\ll$	0.857 $\gg$	9
rough	0.364 $\gg$	0.545 $\gg$	0.500$\ll$	0.667 $>$	9
thick	0.776 $\gg$	0.581 $\gg$	0.522 $\gg$	0.444 $\gg$	9
cool	0.667 $\gg$	0.667 $>$	0.692 $\gg$	0.645 $\gg$	8
slippery	0.842 $\gg$	0.800 $\gg$	0.571 $\gg$	0.643 $\gg$	8
fuzzy	0.303 $\gg$	0.278 $\gg$	0.270 $\gg$	0.333 $\gg$	6
porous	1.000 $\gg$	0.690 $\gg$	0.533 $\gg$	0.377 $\gg$	6
springy	0.541 $\gg$	0.439 $\gg$	0.389 $>$	0.383 $\gg$	6
scratchy	0.462 $\gg$	0.467 $\gg$	0.467 $\gg$	0.298 $\gg$	5
hairy	0.412 $\gg$	0.533 $\gg$	0.500 $\gg$	0.471 $<$	4
bumpy	0.429 $\gg$	0.516 $\gg$	0.857 $\gg$	0.952 $\gg$	2
metallic	0.889 $\gg$	0.727 $>$	0.667 $\gg$	0.667 $\gg$	2

*PE indicates the number of positive examples in the training set.

3.5.3.3 Performance by Signal Type

As mentioned in Section 3.3, the BioTac is designed to imitate human tactile sensing, using different types of sensors to measure skin deformation, pressure, vibration, and temperature. Thus, it is of interest to determine whether certain adjectives are more accurately classified by different haptic

Table 3.3: F_1 scores across adjectives and signals.

	P_{AC}	P_{DC}	T_{AC}	T_{DC}	$E_{1:19}$	PE*
smooth	0.600	0.600	0.621	0.625	0.600	25
solid	0.983	1.000	1.000	1.000	1.000	22
squishy	0.852	0.912	0.909	0.938	1.000	21
compressible	0.967	1.000	0.938	1.000	1.000	20
hard	0.983	1.000	1.000	1.000	1.000	20
textured	0.444	0.552	0.444	0.444	0.449	16
soft	0.679	0.976	0.769	0.927	0.952	13
absorbent	0.950	0.667	0.870	0.737	0.900	9
rough	0.696	0.462	0.750	0.824	0.391	9
thick	0.559	0.800	0.723	0.696	0.462	9
cool	0.783	0.643	0.833	0.667	0.692	8
slippery	0.769	0.783	0.857	0.667	0.700	8
fuzzy	0.385	0.267	0.290	0.286	0.274	6
porous	0.727	1.000	0.741	0.947	0.952	6
springy	0.364	0.353	0.333	0.444	0.609	6
scratchy	0.333	0.621	0.367	0.372	0.400	5
hairy	0.643	0.467	0.353	0.270	0.444	4
bumpy	0.455	0.690	0.621	0.444	0.900	2
metallic	0.857	0.714	0.818	0.667	0.571	2

*PE indicates the number of positive examples in the training set.

signal types. For example, do P_{AC} and P_{DC} do a better job of classifying texture-related adjectives like *smooth* and *rough* while T_{AC} and T_{DC} excel at classifying more temperature-related adjectives like *cool* and *metallic*? A classifier was trained for each signal type using the concatenated features extracted from each EP. The results are shown in Table 3.3. The overall performance for each adjective is comparable to that of the EP-specific classifiers. The five sensor signal types perform similarly for many adjectives, but there are some adjectives, such as *bumpy*, for which certain signal types vastly outperform others.

3.5.4 Discussion

Our results demonstrate that learned features greatly outperform hand-crafted features on the task of adjective classification in almost all cases. Additionally, we found that certain exploratory procedures and sensor signal types are better at identifying specific adjectives, although performance across all EPs and sensor signal types is similar for many adjectives.

When grouped by EP, the learned features consistently and significantly improved the classification performance across all adjectives and EPs. All three cases in which the learned features under-perform occur for the EPs *Slow Slide* and *Fast Slide*. However, other EPs classify those adjectives more successfully. Additionally, as expected, our classifier generally performs worse as the number of positive examples decreases. Interestingly, the EPs *Squeeze* and *Hold* classify *slippery*, which is typically considered a texture-related adjective, more accurately than the two sliding EPs. Unsurprisingly, however, the EPs *Slow Slide* and *Fast Slide* outperform the others on both *rough* and *bumpy*. Similarly to Chu et al. [CMR+15], our classifiers struggle to accurately identify *smooth* and *textured*, even though they both have a relatively large number of positive examples. The learned features also perform poorly on many of the texture-related adjectives. As Chu et al. mention, this poor performance probably stems from the fact that the ridges on the BioTac skin degraded over the course of data collection [CMR+15].

The learned features also perform well when grouped by sensor signal type, displaying similar trends across all the adjectives. Again, many adjectives are classified consistently across all the signal types. However, there are some adjectives for which certain signal types perform much better than others. For example, P_{AC}, T_{AC}, and $E_{1:19}$ perform very well for *absorbent*, whereas P_{DC}, T_{DC}, and $E_{1:19}$ perform very well for *porous* and *soft*. It is interesting to note that the signal types seem to perform well in the pairs $\{P_{AC}\ ,\ T_{AC}\}$ and $\{P_{DC}\ ,\ T_{DC}\}$. This pattern could indicate that there is coupling between these signals in the BioTac. One surprising result is that T_{DC} classifies *rough* more accurately than any of the other sensor types.

One potential reason for the inconsistency in performance across adjectives

could be how the dictionary learning parameters were optimized. Although we chose parameters that maximized the F_1 score averaged across all adjectives, we noticed that these choices did not maximize the F_1 score for each individual adjective. On the contrary, the optimized parameters rarely achieved the best results on any single adjective, but instead they were the most consistent. Another strategy to optimize parameters could be to determine which EP-signal pairs achieve the highest scores for each adjective and then optimize each EP-signal dictionary to achieve the best results for the associated adjective.

3.6 Experiment 2: Scaled Adjective Rating Prediction

Although unsupervised feature learning performs well, there are some limitations to reducing haptic properties to binary ratings. One drawback is that the binary labels are determined by taking the consensus of binary labels provided by multiple humans, as described in Section 3.3.2 (or in other examples by thresholding measured mechanical properties such as stiffness [BRK18b]). In either case, a rich, continuous perceptual space for humans is reduced to a much simpler binary space for an artificial system, which requires selection of an arbitrary threshold and ignores any perceived differences in the strength of attributes. Additionally, associating a single label with a trial ignores the natural variability in perception across individuals and interactions. A self-aware human recognizes that some other people would respond differently and might even be able to estimate the distribution of reactions a population would provide. The work in this section explores the more complex ordinal (scaled) ratings from the PHAC-2 dataset.

There are a variety of ordinal regression and classification algorithms that attempt to model a latent variable underlying ordinal data [GPS+16]. However, these approaches typically account for a variable that underlies the entire distribution of responses. In the case of the scaled ratings from the PHAC-2 dataset, each of the 60 objects has its own distribution of labels for each attribute, which depends on both the object and on the entire underlying

perceptual distribution of that attribute. Said another way, different people have different opinions about how to apply specific descriptions. For example, some people might say a particular blanket is soft, while others perceive it to be very soft. With enough data, these variations across people can be captured. Thus, given a single interaction with an object, it should be possible to predict the distribution of labels that interaction and object would receive if experienced by a large number of people. Such functionality would be useful for companies selling tangible products to quickly understand how a particular material will be perceived by a range of possible customers. However, we could not find any algorithm that can predict a distribution of responses from a single interaction; all of them predict single labels.

In this section, we first present a method for capturing these perceptual distributions and learning to predict them from raw data. Second, we present the details of the model training procedure and our experiments. Finally, we analyze both the modeling results and the collected haptic adjective ratings. An overview of the full feature extraction and modeling process is shown in Figure 3.6. As in the first experiment, we use the features generated by K-SVD and ST-HMP. However, we did not do a comprehensive grid search over hyperparameter values. Instead we simply used a 50% window overlap for K-SVD with $K = 40$ for P_{AC}, $K = 25$ for P_{DC} and T_{AC}, and $K = 10$ for T_{DC} and $E_{1:19}$.

3.6.1 Capturing Perceptual Distributions

As described in Section 3.3, each of the 60 objects has approximately 12 rated responses for each of the 10 adjectives. With each response selected from five possible rating classes for each adjective, each object can be given a distinct five-dimensional label $L_{a,o} = \{n_1, n_2, n_3, n_4, n_5\}$ for each adjective a, where o represents the object and n_{x_i} is the number of times that the particular rating x_i was chosen by the participants for the selected adjective-object pair.

Given the collected ratings, there exists a unique probability distribution of ratings for any given adjective-object pair, where for a given rating x, the $P(x|a, o) = {}^{n_x}/_{\sum_i n_{x_i}}$. Additionally, because the ratings are ordinal, there is

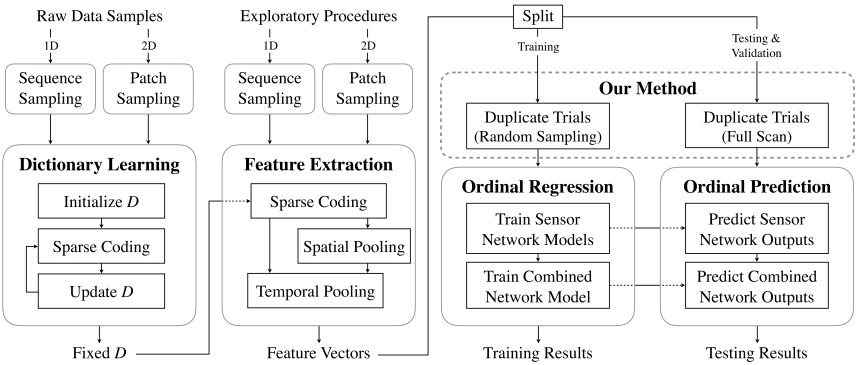


Figure 3.6: Summary of the data-processing pipeline. Samples from raw sensor data (either subsequences or patches of 1D and 2D signals, respectively) are collected and used to learn dictionaries in an unsupervised manner. These are then used to extract features from full exploratory procedure trials. A subset of the feature vectors is to train neural networks to perform ordinal regression. The learned models are tested on a distinct subset of feature vectors.

a corresponding cumulative distribution function (CDF) defined for a discrete random variable X such that $F_{X,(a,o)}(x) = P(X \leq x) = \sum_{x_i \leq x} P(x_i|a,o))$. More generally, the probability of a particular response is a function of the random variable.

In order to predict a probability distribution of adjective responses for a single trial, we designed a method that trains a model to learn an approximation of the *inverse* of $F_{a,o}(x)$ for all (a,o) pairs, along with how that inverse function depends on the features extracted from raw data. Then, given new features, the model can predict the inverse of $F(x)$ for that specific trial, and thus an approximate distribution of expected responses. The inverse of $F(x)$ is called the quantile or inverse cumulative distribution function and is defined as $F^{-1}(p) = \inf\{x \in \mathbb{R} : p \leq F(x)\}, p \in [0,1]$. The inverse CDF for each adjective of the Blue Sponge object is shown in Figure 3.7. This approach differs from traditional cumulative link models [Agr02] because it

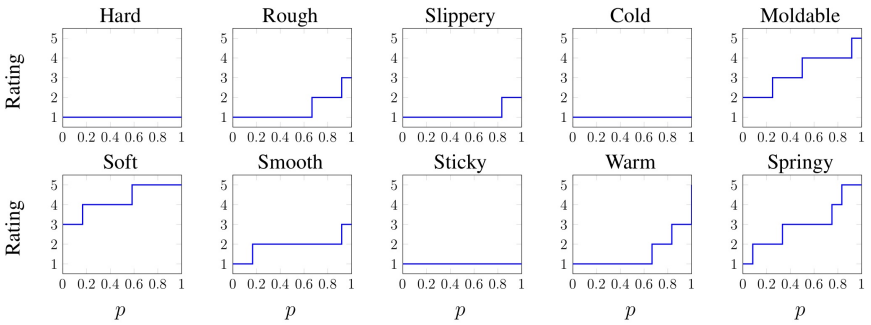


Figure 3.7: Inverse cumulative distribution function of each adjective for the Blue Sponge object. Recall the meaning of the ratings: 1 – "not at all (e.g., hard)", 2 – "slightly (hard)", 3 – "somewhat (hard)", 4 – "(hard)", and 5 – "very (hard)."

learns an inverse cumulative distribution function for each specific object instead of for an entire population. The method works as follows.

Method Description During model training, each trial feature vector f_t is duplicated a fixed number of times W. For each duplicate $f_{t,w}$, one extra feature $p_w \sim \mathcal{U}\{0, 1\}$ is added to the end of the feature vector. Thus, each duplicate of a trial is identical except for the last feature. The single labels $x_{i,(t,w)}$ for the modified duplicates are assigned using $F_o^{-1}(p_w)$, where $F_o(x)$ is the cumulative distribution function for the object being explored during that particular trial. One can think of p_w as indicating the position of the rater in the population; it shows in a continuous way whether the associated rating is near the low end, the middle, or the high end of the distribution of all ratings for this interaction.

To predict the distribution of labels for a new trial, the feature vector is again duplicated. However, in this case the extra variable is simply incremented from 0 to 1. For each modified duplicate, one rating is predicted. Therefore, any changes in the predicted rating across duplicates depend only on the added variable. This method can thus predict the inverse cumulative distribution function for single trials. The separate training and testing

processes are highlighted in the last two columns of Figure 3.6.

3.6.2 Ordinal Classification

Using the features extracted by dictionary learning and our method for capturing perceptual distributions, we trained models to learn how to predict label distributions for new interactions. Because the adjective ratings are ordinal (i.e., they have a relative order but no defined scale), we use ordinal regression instead of traditional multi-class classification. Ordinal regression accounts for the ordered nature of the ratings, whereas multi-class classification ignores it.

Specifically, we used the proportional odds model neural network (NNPOM) algorithm [GTH14]. NNPOM is an extension of the proportional odds model (POM). POM estimates the inverse CDF of ordinal labels as a linear model of the inputs [McC80]. NNPOM uses a single hidden layer of neurons between the input and the POM; it thus estimates the inverse CDF as a linear model of nonlinear basis functions from the hidden neurons. We chose this algorithm because it has sufficiently high performance with low training times compared to other common ordinal classification algorithms like support vector machine (SVM) methods.

Separate and Combined Models With 20 separate feature sets for each combination of sensor modality and EP, it was natural to train an adjective-specific model for each feature set to determine which combinations perform well for which adjectives. We used NNPOM with a sigmoid activation function to train each model. A total of 20 models, one for each feature set, was trained per adjective.

We again run experiments to determine how each robot sensor modality contributes to the learning and prediction of different haptic properties. To learn the contribution of each sensor modality to adjective perception and to determine whether performance is improved by including all sensor modalities in one model, we trained additional NNPOM models for each EP; these models merge one EP's five learned representations from the sensor-specific models. Specifically, the outputs of the hidden layer neurons from

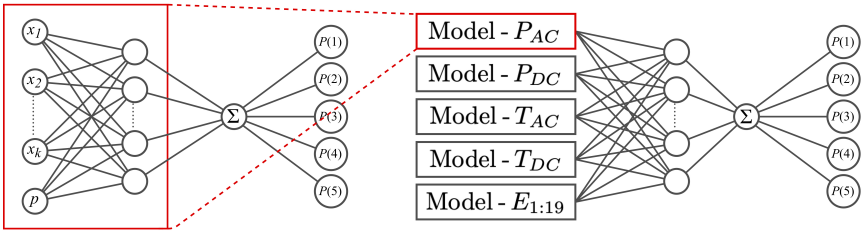


Figure 3.8: Neural network structure for the individual sensor model and the combined model, each with one hidden layer. The combined model takes as input the outputs from the hidden layers of five individual models, one from each sensor modality.

the optimized sensor-specific models were used as the inputs to a combined NNPOM model. The structure of the combined model is shown in Figure 3.8. A total of four fully combined models, one for each EP, was trained for each adjective to measure the overall performance change. To compare the individual contributions of the sensor types, additional combined models were trained while holding out the features from a single sensor (by setting their features all to zero). Five of these holdout models were trained for each EP-adjective pair. To train both the sensor-specific and combined models, we used the NNPOM implementation developed by Gutiérrez et al. [GPS+16].

3.6.3 Model Training

To train and validate the models, we split the 60 objects into separate training, validation, and testing sets for each adjective. Six objects were used for each of the validation and testing sets, and the remaining 48 objects comprised the corresponding training set. To prevent the classifiers from learning to understand objects instead of adjectives, all ten trials for each object were kept together in the same set.

We performed cross-validation by training models on the training set and measuring their accuracy on the validation sets. This approach was used to optimize the model parameter N, the number of neurons in the hidden layer, over the set $\{1, 5, 10, 20, 30\}$, and the parameter λ, the regularization

parameter, over the set $\{0.001, 0.01, 0.1, 1, 10\}$. During model training the validation error was measured every ten iterations. After 150 iterations with no decrease in error, the training stopped, and the model with the best performance was kept.

Each model was trained according to the process described in Section 3.6.1. Each of the training feature vectors was duplicated 15 times, a different random number $p \sim \mathcal{U}\{0, 1\}$ was added to each duplicate, and the duplicates were labeled using $F_o^{-1}(p)$ of the corresponding object o (for a total of 15 duplicates $\times$ 10 trials $=$ 150 training examples per object). The validation and test trials were duplicated 99 times with the added extra variable p incremented by 0.01 from 0 to 1 noninclusive, and the ground-truth labels were assigned in the same way as they were for the training samples.

Then for each adjective, four EP-specific combined models were optimized, where each model trains using information from all five sensory modalities. As shown in Figure 3.8, the outputs of the hidden layers of the optimized sensor-specific models are used as inputs to the combined model. Again, cross-validation was used to optimize N, λ, and the number of training iterations. The training, validation, and test sets were again prepared according to the process in Section 3.6.1 with some minor changes. In this case, the training trials for the combined model are each comprised of the feature vectors from all five sensor modalities. Each combined trial was duplicated 15 times, and a different random number $p \sim \mathcal{U}\{0, 1\}$ was added to each combined duplicate and copied to each sensor-specific feature vector. The labels for the combined duplicates were assigned in the same way as above, and the validation and testing trials were prepared similarly.

For each of the 40 combined models (4 EPs $\times$ 10 adjectives), five additional holdout models were trained to measure the contribution of each sensor modality to the system's overall performance. Each holdout model has the same parameters (N and λ) as the corresponding combined model, and the number of training iterations was optimized on the validation set as described above. For each of the five holdout models, the features from a different single sensor model were held out of training and testing. By measuring the difference in test error between the combined model and

each of the holdout models, we can measure the relative contribution of each sensor modality. There are a total of 200 holdout models (5 sensor types $\times$ 4 EPs $\times$ 10 adjectives) in addition to the 40 combined models. For each EP-adjective pair, there are a total of six types of grouped models: the combined (nothing held out), P_{AC}-holdout, P_{DC}-holdout, T_{AC}-holdout, T_{DC}-holdout, and $E_{1:19}$-holdout models.

In all validation and testing, the performance of the models was measured by taking the average across all trials of the per-trial macroaveraged mean absolute error (MAE^M) metric, as defined by Baccianella, Esuli, and Sebastiani [BES09]. We use MAE^M because it measures error for imbalanced ordinal datasets more precisely than traditional error metrics such as Mean Absolute Error. Specifically, it normalizes the contribution to the error by class. To define it for a single trial t, let the set of duplicate feature vectors $f_{t,w}$ and associated labels $y_{t,w}$ be denoted Td_t, and let X_t be the set of ratings x_i that are represented in Td_t. With these definitions in mind, the per-trial MAE^M can be defined as:

$$MAE^M\left(\hat{\Phi}, Td_t\right) = \frac{1}{|X_t|}\sum_{x_i \in X_t}\frac{1}{|Td_{t,x_i}|}\sum_{f_{t,w}\in Td_{t,x_i}}\left|\hat{\Phi}(f_{t,w}) - y_{t,w}\right| \quad (3.4)$$

where $\hat{\Phi}$ represents the learned model, Td_{t,x_i} denotes the set of duplicates with true labels $y_{t,w} = x_i$, and $|X_t|$ and $|Td_{t,x_i}|$ denote the cardinality of the respective sets.

3.6.4 Evaluating the Human Participant Ratings

When considering the participant ratings, we first wanted to investigate how well participants agreed on how to apply each set of scaled haptic adjective ratings to each object. We quantified interrater agreement for each adjective-object combination by calculating r_{wg}, the most common such metric used in the literature [ONe17]. It is defined as

$$r_{wg} = 1 - \frac{S_X^2}{\sigma_{eu}^2} = 1 - \frac{S_X^2}{\left(\frac{A^2-1}{12}\right)}, \quad (3.5)$$

where S_X is the observed variance in the participants' ratings with the chosen adjective scale on the chosen object and σ_{eu} is the variance of the null distribution, which we set to the variance of a uniform distribution across our $A = 5$ categories. This metric is equal to one when all participants choose the same adjective rating for an object, and it is zero when they choose randomly among the categories. Negative values indicate less agreement than what stems from random guessing; we do not set negative values to zero, as is sometimes done, to preserve the information provided by the calculation [ONe17].

Second, given the uncertainty in the current literature, we investigated the extent to which participants actually used the five adjective pairs as antonyms; we were particularly uncertain about the antonym relationships between **slippery** and **sticky**, and between **moldable** and **springy**, which have not been firmly established as antonym pairs. We investigated this question by calculating Spearman's rank-order correlation, ρ, between all possible pairs of adjective ratings. Spearman's ρ is a nonparametric measure of rank correlation, similar to the Pearson product-moment correlation for parametric data; we calculated it using the MATLAB function `corr` with the `'type'` option set to `'spearman'`. The magnitude of the resulting value shows the strength of the association between the two involved adjectives, with values near zero indicating no correlation. The sign of ρ shows the direction of the association; synonyms have a large positive correlation, while antonyms have a large negative correlation. We also evaluate the p-value associated with each observed correlation, using $\alpha = 0.05$ to determine significance.

3.6.5 Results

We analyze how the study participants used the scaled haptic adjective ratings, and then we investigate the extent to which features automatically extracted from the raw tactile data can be used to learn distributions over scaled adjective ratings.

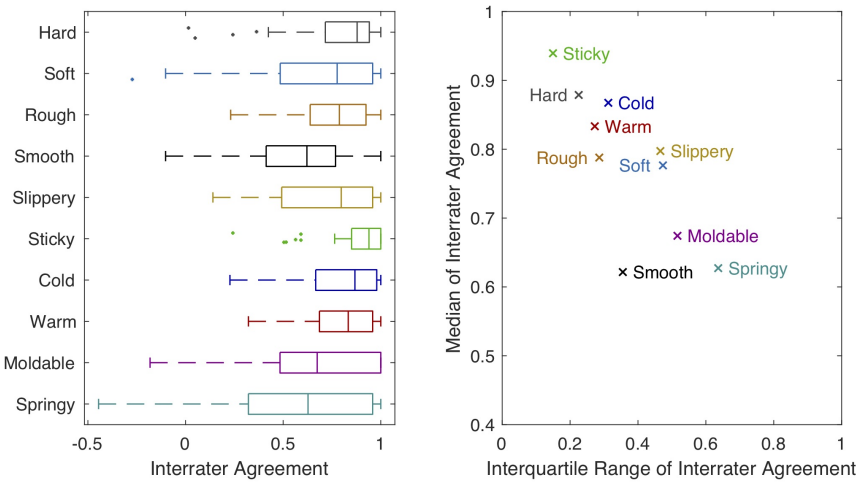


Figure 3.9: The boxplot on the left shows interrater agreement (r_{wg}) for each of our ten haptic adjective scales. The central mark of each box indicates the median of the distribution across objects. The edges of the box are the 25th and 75th percentiles; the whiskers extend to the most extreme datapoints that are less than 1.5 times the interquartile range (IQR) away from the closer 25th or 75th percentile mark. Outlier points outside this range are plotted individually. The graph on the right plots the median of r_{wg} against its IQR for each haptic adjective scale.

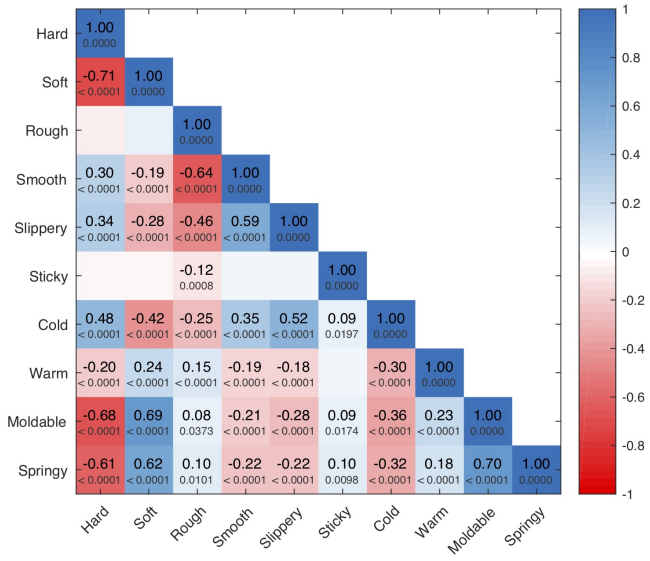


Figure 3.10: Spearman's rank-order correlation ρ for all pairs of haptic adjective scales, along with the associated p-value. To improve readability, we omit these values for insignificant correlations. Boxes showing strong synonyms are colored blue (including the self-synonyms along the diagonal), while strong antonyms are colored red.

3.6.5.1 Human Perception

Figure 3.9 shows the distribution of interrater agreement r_{wg} across all 60 objects for each of our ten adjective scales. The adjectives **sticky**, **hard**, **cold**, **warm**, and **rough** all have relatively high median values (> 0.75) and relatively small IQRs (< 0.35). **Soft** and **slippery** also have relatively high medians but more variation across objects. **Smooth**, **moldable**, and **springy** have the lowest medians (< 0.70) paired with higher IQRs.

Our correlation analysis appears in Figure 3.10. We see a strong, significant

antonym relationship between **hard** and **soft** ($\rho = -0.71, p < 0.0001$), as well as between **rough** and **smooth** ($\rho = -0.64, p < 0.0001$). **Sticky** and **slippery** are uncorrelated. **Cold** and **warm** appear to be weak, significant antonyms ($\rho = -0.30, p < 0.0001$), whereas **moldable** and **springy** show a strong, significant synonym relationship ($\rho = 0.70, p < 0.0001$). Both **moldable** and **springy** are strongly positively correlated with **soft**, showing participants used these three adjectives largely synonymously. **Slippery** is strongly correlated with **smooth** (and anti-correlated with **rough**), showing that participants used this pair largely synonymously. **Hard** and **cold** are also significantly positively correlated with **smooth** and **slippery**. Interestingly, **sticky** has no strong positive or negative correlations.

3.6.5.2 Robot Perception

To obtain the following results, models were first trained and optimized on separate training and validation sets. To account for the variation in neural network performance caused by the random initialization of the weights, ten final models were trained for each of the six types of grouped models (all sensory data streams together plus five holdouts) for every EP-adjective pair, and these models were all evaluated on a testing set that was completely held out during training and optimization. As a sample test-set result from a single combined model, the predicted inverse CDFs of the adjective **cold** for all ten *Fast Slide* trials from the plastic Cutting Board (CB) object are shown in Figure 3.11 and compared to $F^{-1}_{\text{cold},CB}$. The average MAE^M across these ten trials is 0.4355, which is less than half a point on the scale from 1 to 5. Each trial has a different distribution because the recorded tactile data are unique, due to slightly different initial conditions. Some predictions are clearly quite close to the true labels, and in other trials the predicted distribution differs from the true distribution by approximately one rating point.

Model performance was measured by calculating the macroaveraged mean absolute error per trial and then averaging over all the testing trials. The average performance of every set of ten models is shown in Figure 3.12.

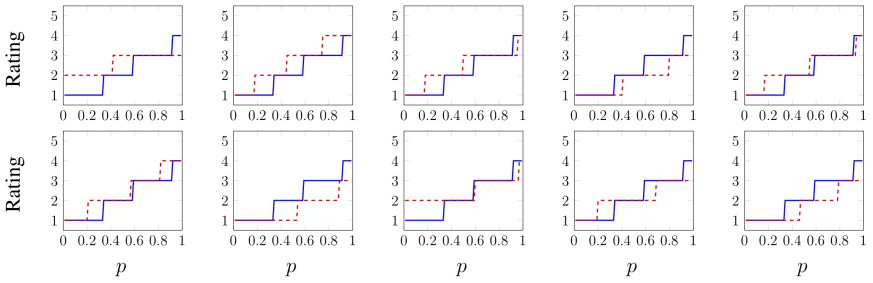


Figure 3.11: Predicted distributions of the adjective **cold** for all ten *Fast Slide* trials from the plastic Cutting Board object. The predicted inverse CDFs are shown in dashed red, whereas $F^{-1}_{\text{cold,CB}}$ is shown in blue (and is the same across all ten trials). The ratings mean 1 – "not at all cold", 2 – "slightly cold", 3 – "somewhat cold", 4 – "cold", and 5 – "very cold." The average MAE^M across all ten trials is 0.4355.

The bars labeled "$-None$" display the average performance of the models in which no sensors were held out. The labels for the remaining bars indicate the sensor type that was held out. Error bars display the standard deviation of performance across the ten models. The Kruskal-Wallis test was used to determine whether the observed differences in performance between the holdout models and the combined models are statistically significant; an asterisk indicates $p < 0.05$. For certain adjectives, some EPs perform better than others. For example, *Fast Slide* outperforms the other EPs for **rough**. Additionally it is clear that certain sensory modalities are important for modeling particular adjectives, and that these influential sensors can differ across EPs for a single adjective.

3.6.6 Discussion

In this section we introduced a learning method for predicting perceptual distributions of haptic adjectives from single interactions. We used this method to additionally test the effectiveness of unsupervised feature learning and how certain exploratory procedures and sensory modalities influence haptic

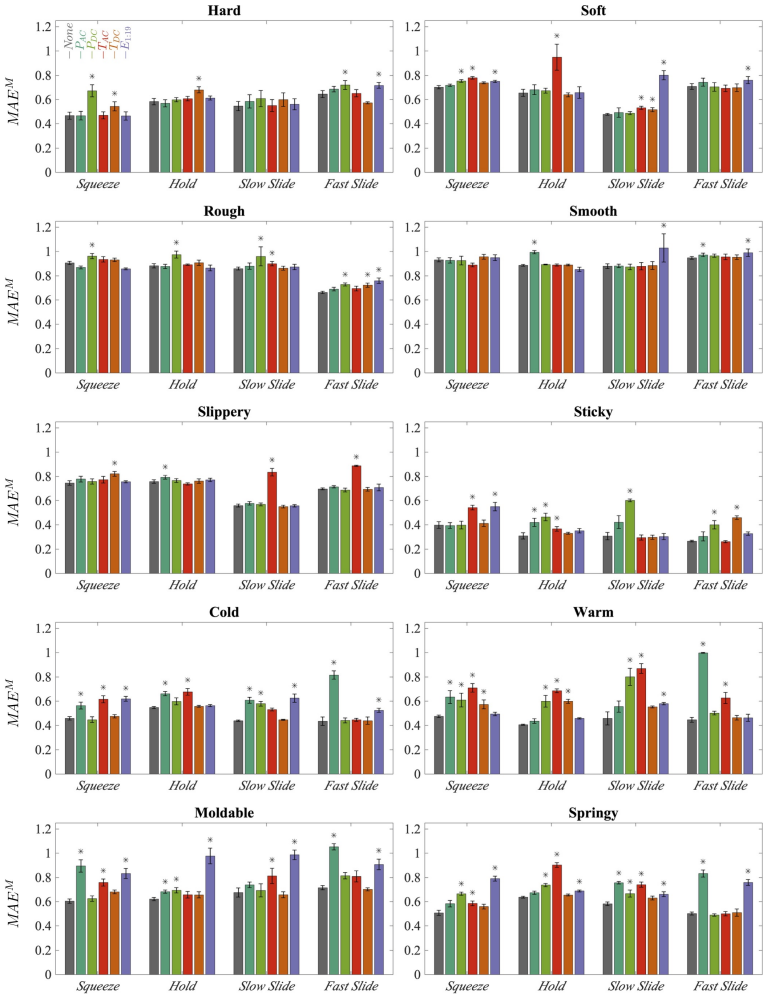


Figure 3.12: Average error of all ten trained models for each type of grouped model, sorted by exploratory procedure and adjective; lower error is better. Error bars display standard deviations across ten trained models. The label of a single bar indicates the sensor type that was held out during model training. Asterisks mark statistically significant decreases in average performance compared to the combined model "$-None$."

adjective prediction. The presented results demonstrate that our learning method can successfully model a distribution of possible adjective labels for a single interaction with an object that has never been previously touched. Additionally, we found that certain sensory modalities and exploratory procedures were more significant to predicting specific haptic adjectives than others. The analysis of the human labels allows us to evaluate how people interpret the meaning of certain haptic adjectives and whether the adjective pairs are indeed used as antonyms.

3.6.6.1 Human Labels

Haptics researchers have proposed the ten adjectives we studied as possible antonym pairs representing both relevant and primary dimensions of perception. We wanted to further test these propositions and also validate the collected labels for our subsequent machine-learning investigations.

Even though this labeling task is in principle more straightforward that the similarity rating task from Chapter 2, we found that the study participants used some haptic adjective scales more consistently than others. These patterns may stem from underlying dis/agreement about the definitions of the employed adjectives, or they might come from the design of the experiment, such as the chosen set of objects. **Sticky** stands out as having high median agreement with low variation in agreement across objects. As seen in Figure 3.3, only one object (Silicone Block) was rated "very **sticky**." Most other objects were rated "not at all **sticky**," yielding the overall high agreement about the use of this adjective. **Sticky** has no strong positive or negative correlations with the other studied adjectives, but this is because there are very few objects that were rated as sticky. Thus, we cannot make strong claims about the relationship between **sticky** and other haptic adjectives.

The full 1–5 scale was used much more frequently for **hard**, **cold**, **warm**, and **rough**. Thus, we believe their high median agreement and relatively small agreement variation across objects indicates that participants were generally consistent with one another in how they applied these haptic ad-

jectives. Indeed, all four of these adjectives have only one physically relevant definition in a modern American dictionary [SL10], with the possible exception of **warm**, whose physical definitions pertain both to temperature itself and to the ability of a material to keep the body warm. It is thus reasonable to expect that all participants were applying approximately the same definition as they made their **hard**, **cold**, **warm**, and **rough** ratings. The weak, significant antonym relationship between **cold** and **warm** reinforces the conclusion that participants used these adjectives consistently; a stronger antonym correlation might have been observed if we tested thermal adjectives that were more closely matched in intensity, such as cool/warm or cold/hot. Interestingly, we did find significant correlations between **hard** and **cold** despite the strong agreement about definitions that don't seem related. This phenomenon could be explained by hedonics, which argues that human sensory perception is affected by emotional attributes. For example, **hard** and **cold** could be correlated with higher arousal, whereas **soft** and **warm** might be correlated with higher comfort [GDM+10].

Participants used the full range of ratings for both **soft** and **slippery** but agreed less on their use than on that of the aforementioned adjectives. The disagreement about **soft** most likely stems from the fact that it has two distinct physically relevant meanings [SL10]: one pertains to being easy to compress (the antonym to **hard**, as substantiated by a strong negative correlation between these adjectives), while the other pertains to texture. In contrast, **slippery** has only one physical definition [SL10], so the disagreement on its use may instead stem from disagreement about intensity – how **slippery** is "very **slippery**?"

The relatively low agreement about the words **smooth**, **moldable**, and **springy** may be a warning to other researchers interested in using these words in their studies. As with **slippery**, participants used the full range of ratings for **smooth**; this haptic adjective has only one definition [SL10], so the observed disagreement most likely stems from variations in how people perceive smoothness intensity. We do not know why this adjective's use suffered more than others from the fact that we did not provide adjective definitions or ground our scales with physical examples. Encouragingly,

smooth was reliably used as an antonym to **rough**, again substantiating our belief that variations in scaling (and not the fundamental definition of the word) are responsible for **smooth**'s low interrater agreement.

In contrast to the other eight adjectives, **moldable** and **springy** are uncommon words in American English; **moldable** does not even have its own dictionary entry [SL10]. Thus we believe that a lack of knowledge of the intended meanings of these adjectives (centered on whether the surface quickly returns when pressed and released) prevented participants from being able to apply them consistently. This physical property is also difficult to judge on hard materials, as they do not deflect perceptibly when squeezed; consequently, the disagreement about **moldable** and **springy** may simply reflect a human inability to perceive such differences for many of the chosen objects. Without guidance, it seems that participants use both of these words in a similar way as **soft**.

These findings validate the collected labels and shed insights on how these ten haptic adjectives are used by everyday Americans. We believe other researchers studying human and robot perception of haptic properties will be able to design their own studies more efficiently by considering these results.

3.6.6.2 Model Performance and Influence of Sensory Modalities

The variance of human perception is rarely represented in the labeling of data or captured by machine learning. However, our proposed method demonstrates that it is indeed possible to model this variance. We found interesting differences in performance across adjectives and across EPs within single adjectives. Additionally, by holding out each sensor modality separately and training multiple models with the same architecture, we were able to measure whether certain tactile data types are better predictors of certain adjectives within single exploratory procedures. Many of the results make intuitive sense, suggesting that our method captures relevant structure that can describe various haptic attributes. As far as we are aware, ours is the first method to predict the probability distribution over an ordinal variable

from a single test trial.

For discrimination of **hard**, P_{DC} seems to be the single most important sensor modality; the increase in error for the EP *Squeeze* is by far the largest increase for any holdout model for the adjective **hard**. Surprisingly, T_{DC} is also a valuable predictor. However, this finding could be explained by the positive correlation between **hard** and **cold**, as shown in Figure 3.10. Similar patterns are apparent in the perception of **soft**; again, pressure and temperature seem to be important contributors. However, in this case the spatially distributed fingertip deformation readings, $E_{1:19}$, are more important than P_{DC}, probably because the perception of **soft** heavily relies on cutaneous information [SL95].

Rough and **smooth** are more texture-related properties than **hard** or **soft**. As might be expected, they depend more on P_{AC}, P_{DC}, and $E_{1:19}$. However, overall performance is weak, which could explain why no individual sensor contributes to prediction dramatically more than any other. This low performance aligns with previous analysis of this dataset, which found that it is difficult to accurately predict **rough** and **smooth** even in a simpler binary classification task [CMR+15], most likely due to the degradation of the BioTac surface ridges over the course of data collection.

For **slippery**, it is interesting that the only large increases in error occurs when T_{AC} is held out, and that this increase occurs only for *Slow Slide* and *Fast Slide*. Such behavior is reasonable because **slippery** pertains to sliding friction and has a relatively strong correlation with **cold**. However, it is surprising that the electrodes $E_{1:19}$ don't seem to play a significant role. For *Squeeze* and *Hold*, it seems like slip information is encoded in every sensor, although performance is weaker on average. The models predict **sticky** very well. However, this good performance is almost certainly because the labels for **sticky** have a strong bias toward "not at all **sticky**," which makes it easier to learn a model for **sticky** from these data. As such, it is more accurate to say that the robot learned only an absence of **sticky**, and not actually the feeling of **sticky**.

Cold is influenced more by pressure than by temperature sensors, whereas **warm** is influenced more by the temperature modalities. Although it is not

surprising that P_{AC} is so important to prediction for *Fast Slide*, given the dynamic nature of this EP, it is surprising that P_{AC} seems to have more influence on temperature-related adjectives than texture-related adjectives. This unexpected dependence on pressure could be a limitation of the object set, in that a majority of the thermally conductive objects are both **hard** and **smooth**. It is possible that these correlated properties are easier to detect than **cold** itself. **Warm** depends more on temperature sensors, which is reasonable given that it was found to be more independent from the other adjectives than **cold**.

The models for **moldable** and **springy** depend on many of the same sensor modalities. For both adjectives, the electrodes $E_{1:19}$ are significant for every EP. Additionally, the EPs *Squeeze* and *Slow Slide* are both dependent on T_{AC}. These sensor modality influences are similar to those for **soft**. Interestingly, both of these adjectives are highly correlated with **soft** and each other, as shown in Figure 3.10. This finding may demonstrate that certain object properties that are significant to humans' judgment of multiple haptic attributes are being captured by the robot sensors and used in the modeling of adjectives.

There are a variety of potential limitations to our implementation of these methods. Particularly, the dictionaries were not optimized for this learning task. Thus, it is possible that certain sensory modalities provided less information than might be expected. Additionally, the individual sensor models were optimized separately from the combined model. By optimizing the individual and combined models simultaneously, the learned representations could likely be improved.

We also did not evaluate the model performance as a function of the number of random samples taken from the label distributions. Undersampling could prevent models from learning how the distribution of labels correlates with the tactile data, whereas oversampling could cause the model to overfit the object label distributions. A potentially useful improvement could be to determine how many random samples to take given the total number of ratings for a particular object-adjective pair. Additionally, evaluating whether certain training samples appear to be outliers from the primary

response distribution could be useful. Similarly, we did not look deeply into performance on a per-object basis. Our initial analysis demonstrated that some models perform terribly on one or two objects while performing excellently on the majority. Using a larger and more diverse set of objects and collecting ratings from more human participants would likely improve all of our results.

Because our ordinal regression method evaluates each adjective individually, it ignores the strong positive and negative correlations between adjectives. It might be possible to improve both performance and training efficiency by implementing an algorithm that can learn all adjectives simultaneously, therefore incorporating these inter-adjective relationships into the learning process.

We analyzed how the models performed over the full range of responses when data from certain sensors were removed. However, it is possible that certain sensory modalities might not have equivalent predictive power across the full response range. For example, to determine the probability distribution of an interaction for the adjective **rough**, a model could use P_{DC} to make a distinction between the ratings {1,2,3} and {4,5}, but be unable to use it to discern ratings within those two groups. Similarly, the electrodes $E_{1:19}$ could provide information that allows the model to discriminate between ratings 4 and 5. Analyzing how the contributions of sensor modalities vary across the full range of ratings could provide greater insight into what type of information is used to determine the haptic attributes of objects.

3.7 Summary

In this chapter, we used K-SVD and ST-HMP to extract features from multimodal haptic data. We used these features for binary adjective classification, comparing them to hand-crafted features for 19 different adjectives, and ordinal adjective prediction. We additionally evaluated how each individual robot EP and signal type performed across tasks.

In the first task, using learned features greatly improved the classification

of adjectives compared to using hand-crafted features, demonstrating the viability of these methods for processing haptic data. Although the second task didn't have a baseline with hand-crafted features, the learned features were sufficient to quite accurately predict ordinal haptic adjective labels. The results also support previous work which suggests that individual EPs are necessary to more accurately discern various object properties. Additionally, it is necessary to acquire rich, multi-modal haptic sensory data, as certain properties can be more accurately identified by different sensors. To further evaluate the strength of these features, the methods could be compared to similar algorithms like bag-of-features [SSS+09] and deep learning methods.

Although the learned features perform very well, separate dictionaries were created for each EP-signal pair. However, it is not necessarily the case that there need be separate representations for each EP; it's possible that a single dictionary or just a subset of atoms could be used for all EPs. Thus, future work could explore the relationships between different dictionaries, comparing how strongly dictionaries overlap across EPs.

We believe the work presented in this chapter is an important step toward fully capturing the robustness and richness of human haptic perception. Furthermore, because unsupervised dictionary learning and our method are easily adapted to different sensor and data types, we believe our research broadens the range of tasks that can be tackled with machine learning.

Chapter 4

Implicit Robot Learning of Haptic Properties from Sequential Interactions

In Chapter 3, we investigated how to use unsupervised feature learning methods to predict human adjective ratings of objects from data gathered during robot interactions with those objects. However, we looked only at how well individual exploratory procedures were able to predict different haptic adjectives. This choice ignores a fundamental aspect of autonomous haptic exploration in the world, which is that information is accumulated across many exploratory procedures over time.

In this chapter, we introduce and evaluate a method by which a haptically sensitive robot can learn robust, general representations of objects by accumulating information over multiple exploratory procedures in an unsupervised manner. We use a variational autoencoder to learn compressed representations of the data, and the representation is updated over time as more information is gathered through exploratory procedures. We rely on

the concept of object permanence, assuming that a representation of a single object remains relatively consistent across multiple interactions with that same object despite never knowing its identity. The work presented in this chapter is in preparation for Robotics: Science and Systems:

> B. A. Richardson, K. J. Kuchenbecker, and G. Martius. *A Sequential Group VAE for Robot Learning of Haptic Representations*. 2023. In preparation for submission to Robotics: Science and Systems.

4.1 Introduction

When people physically interact with unknown items, they very quickly form a perceptual representation of each object [KLM85]. This representation is typically developed by integrating prior knowledge with new information that is gathered by performing exploratory procedures (EPs) [Gib62; LK87]. Each characteristic EP elicits information about certain object properties that are both implicit and explicit, but no one EP alone can provide a complete picture. For example, by pressing into an object, we can determine its stiffness but not necessarily its mass, shape, or size. By enclosing an object we can learn its shape and size, whereas shaking an object provides information about its dynamic properties and whether there are loose contents inside. Thus, we might think that two objects are very similar after one EP, but we can quickly disambiguate them by accumulating information from additional exploration. To operate in and act upon the real world, autonomous robots will need to have a similar ability to interact physically with objects in their environment, accumulate information across sequential exploratory procedures, and form haptic representations that are useful for real-world tasks.

One challenging aspect of this goal is how to accumulate information over a sequence of interactions with the world. Information accumulation has been investigated in haptic exploration for a variety of tasks including surface classification [DGÉC14; FL12], haptic property identification [GS14], and contour following [LAC17]. While high performance was obtained in each of these tasks, they rely on task-specific supervised learning instead of

learning general representations applicable to many tasks.

Besides simply perceiving object properties, learning robots need to acquire and update useful representations of physical interactions with the world. It is unlikely that a robot can be given a priori all the knowledge or broad representative features it will need throughout its existence. Thus, it should have the ability to learn descriptive factors for a wide range of physical tasks. Unsupervised learning has been demonstrated to discover expressive representations of tactile and haptic data that can perform well across a variety of tasks [KSG+19; LSG+17; NGW+12; RK20; THHS20]. However, these representations are learned on huge amounts of data and do not include explicit mechanisms for being updated.

We consider the task of learning comprehensive latent representations of haptic properties from sequences of interactions with objects. Performing a variety of EPs on objects generates multiple observations. No single EP can elicit information about every object property, and thus many observations must be accumulated to build a comprehensive latent representation. We propose a sequential Group Variational Autoencoder (VAE) based on the Multi-Level VAE [BTN18]. Our method learns generic representations that can be used to infer object type and properties and that contain uncertainty about the inference when the representation is learned on insufficient EPs. We analyze our method on a synthetic MNIST variant with multiple sequential crops and on real data from a robot arm and hand that use four exploratory procedures to interact with 52 objects that vary across multiple haptic property dimensions.

The work in this chapter contributes a novel recursive Group-VAE architecture that (i) accumulates information from sequential EPs to learn generic representations of haptic properties and objects, (ii) uses those learned representations to infer observations from unseen EPs, and (iii) contains uncertainty when the EPs are insufficient to elicit information about certain haptic properties.

4.2 Related Work

4.2.1 Haptic Representation Learning

Machine-learning techniques have become more popular in haptics in recent years, being used to classify objects/surfaces [FL12; KSG+19; NGW+12; SLCD16; SSIS17] and semantic properties [CMR+15; GS14; LSG+17] from both tactile and proprioceptive data. A common approach has been to define features informed by the sensing capabilities of human mechanoreceptors or by human haptic perception (e.g. vibration spectral centroid is correlated with hardness perception [LaM00]), extract those features from raw haptic data captured by a tool or robot, and classify a set of objects from those features [FL12; SSIS17]. Others used these types of predefined features to learn semantic attributes of objects which can be applied to new, unseen examples [AGCC18; CMR+13; CMR+15; GS14]. One additional approach has been to use unsupervised learning or compression techniques to develop latent representations of haptic interactions [KSG+19; LSG+17; MBKF14b; NGW+12; RK19; RK20; SSS+09; THHS20], which typically outperform hand-crafted features when used to learn downstream tasks. Bag-of-words models [SSS+09] and dictionary learning [MBKF14b; RK19; RK20] have demonstrated good generalization across many haptic property identification tasks. In particular, Tatiya et al. [THHS20] adopt a β-VAE to learn latent representations of objects by encoding data from one action and decoding it to predict data from a different action.

4.2.2 Haptic Information Accumulation

Information accumulation can occur on multiple timescales during haptic interactions. During single EPs, data can be processed with recurrent models (e.g., LSTMs) that learn to estimate and predict instantaneous state, learning representations of very short moments in time. These methods have demonstrated great effectiveness for a variety of tasks, including hardness detection [BRK18a; YZO+17] and clothing material perception [YMWA18]. Alternatively, features can be learned on small segments and concatenated

to form large feature vectors that represent full EPs [CKS+16; MBKF14b; RK19]. However, this is a different task from accumulating information over multiple exploratory procedures, where relevant information is processed at a different timescale. One method of capturing information across multiple EPs is simply to learn a separate representation of the data captured from each EP and then concatenate those representations [SD14], but this is limited if any downstream model has a fixed number of inputs. A more flexible approach is to update a representation as more information is gathered. Fishel and Loeb [FL12] use Bayesian inference to improve texture classification over sequences of parameterized sliding EPs. Gemici and Saxena [GS14] also use Bayesian inference, but they update beliefs over sets of haptic properties as they perform more EPs. To perform active contour and shape-feature following, Lepora, Aquilina, and Cramphorn [LAC17] use recursive Bayesian inference to modify the control of a tactile sensor tip while it traces shape features like edges and corners. Each of these cases uses supervised learning to adjust relatively simple models to perform specific tasks. This rigid structure limits their ability to learn representations that generalize to new tasks. On the other hand, Dallaire et al. [DGÉC14] perform unsupervised clustering of surfaces using Pitman-Yor process mixture models. This approach assumes that observations come from a discrete set of underlying distributions and is powerful for data that is sampled from multinomial or categorical distributions.

4.2.3 Learning Group Representations

As a robot explores an object over time, it should not assume that the individual observations are independent. If we assume that a robot understands object permanence (that objects continue to exist even when not immediately perceived) [PGV77], at least during the course of a sequence of interactions, we can leverage that knowledge to build grouped representations. Bouchacourt, Tomioka, and Nowozin [BTN18] propose the Multi-Level Variational Autoencoder (ML-VAE) for learning disentangled representations. The ML-VAE splits the latent representation space into two components and forces

all samples from a single group to share a single latent vector in one of those components. This approach can learn general representations of classes on multiple datasets. Sato et al. [SNMU22] use ML-VAE to perform few-shot anomaly detection of images, grouping samples by domain instead of class.

4.3 Methods

Rather than assuming that all data from a sequence of interactions is processed together, our method handles shorter observations from discrete EPs to build up object representations over time, with high flexibility that matches the diverse ways in which robots can interact with objects in the world. Specifically, given observations from a sequence of EPs of an object, our method uses a β-variational autoencoder (β-VAE) to model a probability distribution of the latent variable representation of the observations; this distribution is updated iteratively as each observation in the sequence is processed. Each progressive representation is conditioned on the most recent observation and the previous latent representation, which we call the context. Finally, at each update step, a random sample is drawn from the latent distribution and fed into a decoder network to try to reconstruct all observations in the sequence. We will first introduce the VAE and then a context-aware VAE method called Multi-Level VAE [BTN18]. Finally, we will describe our method.

4.3.1 Background

Variational Autoencoder (VAE) In the standard VAE framework [KW14], we assume a dataset $\mathbf{X} = \{x_1, ..., x_n\}$ composed of independent and identically distributed (i.i.d.) observations generated by a stochastic process from an underlying random variable z, with $z \sim p_\theta(z)$ and $x_i \sim p_\theta(x \mid z)$. The goal is to learn a variational approximation $q_\phi(z \mid x)$ of the true (but typically intractable) posterior over the latent variable $p_\theta(z \mid x)$. Typically, both the distributions $q_\phi(z \mid x)$ and $p_\theta(x \mid z)$ are modeled by encoder and decoder neural networks parameterizing a Gaussian (or normal) distribu-

tion, although other choices are possible. Parameters ϕ and θ are learned simultaneously by minimizing the evidence lower bound $\mathcal{L}$ of the marginal log-likelihood of the data, $\log p_\theta(\mathbf{X})$ (more on this in Section 4.3.2).

Multi-Level Variational Autoencoder (ML-VAE) The ML-VAE [BTN18] does not assume i.i.d. observations, but instead that there are disjoint subsets of observations that come from distinct groups $g \in \mathcal{G}$. Groups are independent from each other, but samples within a single group are not independent. Observations are assumed to be generated from two sets of latent variables: the content C and the style S. Each observation $x_i \in \mathbf{X}_g$ from a group $g \in \mathcal{G}$ is generated from the same latent content variable $\mathbf{c}_g$ and a unique style variable s_i. That is, the likelihood is given by $p_\theta(x_i \mid \mathbf{c}_g, s_i) | \forall x_i \in \mathbf{X}_g$. Because the content and style are assumed to be independent, the variational approximation for a sample $q_\phi(c_i, s_i \mid x_i)$ can be decomposed into the product of $q_{\phi_c}(c_i \mid x_i)$ and $q_{\phi_s}(s_i \mid x_i)$. In this case, q_{ϕ_c} and q_{ϕ_s} are chosen to be normal. A group content variable is approximated by multiplying together all the approximate individual content variables $q_{\phi_c}(\mathbf{c}_g \mid \mathbf{X}_g) \propto \prod_{i \in g} q_{\phi_c}(c_i \mid x_i)$.

4.3.2 Iterative Latent Update via Group VAE

For our method, we assume a group setting where each independent group is an object $o \in \mathcal{O}$. Let the set of observations from object o be denoted $\mathbf{X}_o$. We observe sequences $\mathbf{x}_o \subseteq \mathbf{X}_o$ of observations, where $\mathbf{x}_o = \{x^1_{a^1}, ..., x^t_{a^t}, ..., x^n_{a^n}\}$, $a^t \in A = \{a_1, ..., a_n\}$ indicates which of the n actions was performed to generate that observation, and t indicates the order in the sequence. Like in ML-VAE, we assume that the observations are generated from two independent underlying latent random variables, the content C and style S, in our case also conditioned on the action. Since our robot will have access to only a particular sequence of actions on the same object, and object identities do not have to be known to the robot between trials, we assume that content is shared across a single sequence, such that every element $x^t_{a^t}$

of the sequence $\mathbf{x}_o$ shares the same latent content variable $\mathbf{c}_o$[1]. The style formulation remains unchanged, where each element x^t has its own style variable s^t, and $\mathbf{s}_o = \{s^t \forall x^t \in \mathbf{x}_o\}$. Thus, within a sequence $\mathbf{x}_o$, individual observations $x^t_{a_i}$ are assumed to be generated from the latent variables according to $x^t_{a_i} \sim p_\theta(x \mid \mathbf{c}_o, s^t, a_i)$. Again like the ML-VAE, the variational approximation of a sequence $q_\phi(\mathbf{c}_o, \mathbf{s}_o \mid \mathbf{x}_o)$ decomposes into the product of $q_{\phi_c}(\mathbf{c}_o \mid \mathbf{x}_o)$ and $q_{\phi_s}(\mathbf{s}_o \mid \mathbf{x}_o)$. We also assume these distributions are normal, with

$$q_{\phi_c}(\mathbf{c}_o \mid \mathbf{x}_o) = \mathcal{N}\left(\mathbf{c}_o \mid \mu(\mathbf{x}_o, \phi_c), \Sigma(\mathbf{x}_o, \phi_c)\right), \tag{4.1}$$

$$q_{\phi_s}(\mathbf{s}_o \mid \mathbf{x}_o) = \mathcal{N}\left(\mathbf{s}_o \mid \mu(\mathbf{x}_o, \phi_s), \Sigma(\mathbf{x}_o, \phi_s)\right) \tag{4.2}$$

A key feature of our approach is that the variational approximation is updated iteratively, and at each step the approximation is conditioned on the parameters of the previous variational content approximation, which we call the context. Specifically, given an observation $x^t \in \mathbf{x}_o$ at iteration t, the variational approximation is given by

$$q^{(t)}_\phi \left(\mathbf{c}^{(t)}_o, s^t \mid x^t, q^{(t-1)}_{\phi_c}\right), \quad \text{where } q^{(0)} = \mathcal{N}(\mathbf{0}, I) \tag{4.3}$$

To encourage the content latent variable to capture general descriptions of the object from which a sequence is sampled, our method performs inference for every observation of the sequence at each update step. At each iteration t for sequence $\mathbf{x}_o$, the marginal log-likelihood (or evidence) can be written as the sum of the evidence lower bound (ELBO), denoted $\mathcal{L}$, and the Kullback-Leibler divergence between the true posterior and the variational

[1] A sequence could potentially include every observation from a single object (e.g. over a lifetime of observations), in which case $\mathbf{c}_o$ would be shared across the full object group.

approximation:

$$\log p_\theta(\mathbf{x}_o; t) = \mathrm{KL}\left(q_\phi\left(\mathbf{c}_o^{(t)}, \mathbf{s}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}\right) \,||\, p_\theta\left(\mathbf{c}_o^{(t)}, \mathbf{s}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}\right)\right) + \mathcal{L}^{(t)}\left(\mathbf{x}_o; \theta, \phi_c, \phi_s\right) \tag{4.4}$$

$$\text{where } \mathbf{s}_o^{(t)} = \{s^u; x^u \in \mathbf{x}_o\} \text{ and } q_{\phi_s}\left(\mathbf{s}_o^{(t)} \mid \cdot\right) = \prod q_{\phi_s}\left(s^u \mid \cdot\right),$$

$$\text{with } q_{\phi_s}\left(s^u \mid x^u, q_{\phi_c}^{(u-1)}\right) = \begin{cases} q_{\phi_s}\left(s^u \mid x^u, q_{\phi_c}^{(u-1)}\right) & \text{if } u \leq t \\ \mathcal{N}\left(s^u \mid \mathbf{0}, I\right) & \text{if } u > t \end{cases}.$$

It should be noted that before an observation has been seen by the network, its style is sampled from a standard normal distribution. Because the KL divergence is always non-negative, the ELBO is a lower bound on the marginal log-likelihood. The ELBO for sequence $\mathbf{x}_o$ at step t can itself be written as the negative of the sum of the negative log-likelihood ($\mathcal{L}_{\text{NLL}}$) and the KL divergences between the variational approximations and their corresponding priors:

$$\begin{aligned} \mathcal{L}^{(t)}(\mathbf{x}_o; \theta, \phi_c, \phi_s) = \\ \mathbb{E}_{q_{\phi_c}^{(t)}\left(\mathbf{c}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}\right)} \mathbb{E}_{q_{\phi_s}^{(t)}\left(\mathbf{s}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}\right)} \log p_\theta\left(\mathbf{x}_o \mid \mathbf{c}_o^{(t)}, \mathbf{s}_o^{(t)}, \mathbf{a}_o\right) \\ - \mathrm{KL}\left(q_{\phi_c}^{(t)}(\mathbf{c}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}) \,||\, p_\theta(\mathbf{c}_o)\right) \\ - \mathrm{KL}\left(q_{\phi_s}^{(t)}(\mathbf{s}_o^{(t)} \mid x^t, q_{\phi_c}^{(t-1)}) \,||\, p_\theta(\mathbf{s}_o)\right), \end{aligned} \tag{4.5}$$

where $\mathbf{a}_o$ is the vector of actions that corresponds to the observations in $\mathbf{x}_o$. In practice, we use the β-NLL formulation [STAM22] of the negative log-likelihood. This ELBO loss $\mathcal{L}^{(t)}$ can be summed over all iterations. The full training procedure is described in Algorithm 1.

4.3.3 Demonstration on MNIST

To demonstrate our method on a simple example, we first apply it to the MNIST dataset [LBBH98]. Here, the set of objects $\mathcal{O}$ is comprised of the ten

Algorithm 1: Sequential Group VAE training algorithm.

Given $\mathbf{X} = \{(x^i, a^i, o^i)\}$ with $a^i \in A = \{a_1, \ldots, a_m\}$ and $o^i \in \mathcal{O}$
for each epoch **do**
    `// Create random sequences for training`
    $\mathbf{x}_s \leftarrow \{\}$
    for $(x, a, o) \in \mathbf{X}$ **do**
        $\mathbf{a}_o = \text{permutation}(A \setminus \{a\})$
        $\mathbf{x}_o = \{x$, random set of inputs from object o with actions $\mathbf{a}_o\}$
        $\mathbf{X}_s \leftarrow \{\mathbf{X}_s \cup \mathbf{x}_o\}$
    end
    `// Train model`
    while All sequences not seen **do**
        $\mathbf{X}_{s,b} \leftarrow$ Sample batch of sequences $\mathbf{X}_s$
        for $\mathbf{x} \in \mathbf{X}_{s,b}$ **do**
            $q(c) \leftarrow \mathcal{N}(0, 1)$
            $q(\mathbf{s}) \leftarrow \mathcal{N}(0, 1)^{|\mathbf{x}|}$
            for $t = 1 \ldots | \mathbf{x} |$ **do**
                $x^t \leftarrow \mathbf{x}[t]$
                $q(c) \leftarrow q_{\phi_c}\left(\mathbf{c}^{(t)} \mid x^t,\ q_{\phi_c}^{(t-1)}\right)$ `// Encoding from Eq. 4.3`
                $q(\mathbf{s})[\ t\ ] \leftarrow q_{\phi_s}\left(s^t \mid x^t,\ q_{\phi_c}^{(t-1)}\right)$ `// Update style for action` t
                `// Reconstruct for all actions`
                for $u = 1 \ldots | \mathbf{x} |$ **do**
                    Sample $c^u \sim q(c)$
                    Sample $s^u \sim q(\mathbf{s})[\ u\ ]$
                    $p(x^t) \leftarrow p_\theta(x^t \mid c^u,\ s^u,\ a^u)$ `// Decode` $c^u,\ s^u$
                end
                Compute $\mathcal{L}^{(t)}(\mathbf{x}, p(\mathbf{x});\ \theta,\ \phi_c,\ \phi_s)$ `// From Eq. 4.5`
            end
            $\mathcal{L}_\mathbf{x} = \sum_t \mathcal{L}^{(t)}$
        end
        Update parameters θ, ϕ_c, ϕ_s by back-propagating gradient $\nabla_{\theta,\phi_c,\phi_s} \sum_\mathbf{x} \mathcal{L}_\mathbf{x}$.
    end
end

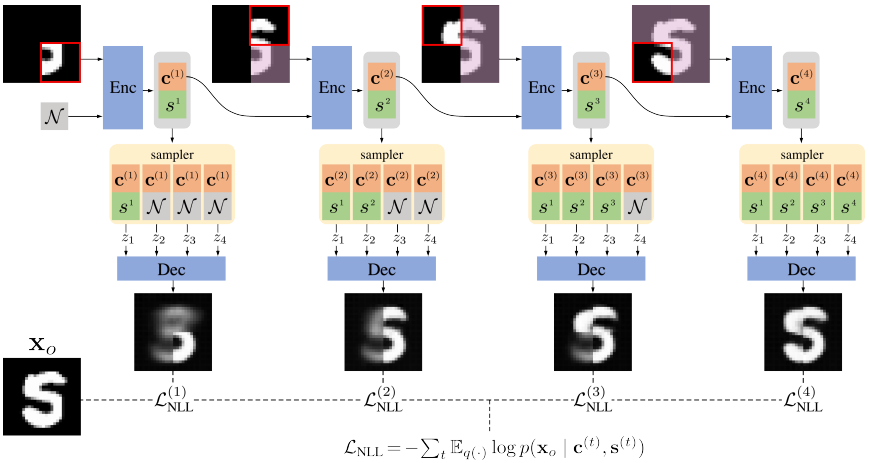


Figure 4.1: Model architecture on an example MNIST digit. At each step t a single crop x^t is fed into the encoder, which outputs probabilistic content and style encodings $\mathbf{c}^{(t)}$ and s^t. The content encoding $q\left(\mathbf{c}^{(t)} \mid x^t, \mathbf{c}^{(t-1)}\right)$ is passed as context into the encoder for the next update. The decoding step attempts to reconstruct all four crops of the MNIST digit. For each crop, the content latent representation $q\left(\mathbf{c}^{(t)} \mid x^t, \mathbf{c}^{(t-1)}\right)$ is sampled. If a crop x^u has already been seen, then the style latent representation $q\left(s^u \mid x^u, \mathbf{c}^{(t-1)}\right)$ is sampled. If a crop hasn't been seen, then $\mathcal{N} = \mathcal{N}(0, 1)$ is sampled. As more crops are observed, the representation improves and becomes more certain.

digits $d = \{0, ..., 9\}$. We define four actions, each of which crops a different quadrant (top right, ..., bottom left) of a traditional MNIST image sample; like one haptic EP, one crop usually does not fully identify the object. For model training, we select a digit d and perform one of each cropping action in random order, generating a sequence four crops long. Then we follow the iterative training procedure described above. Figure 4.1 shows an overview of the full approximation and inference procedure on an example MNIST digit. As individual crops are observed, the content representation is refined, and the style for each crop is inferred.

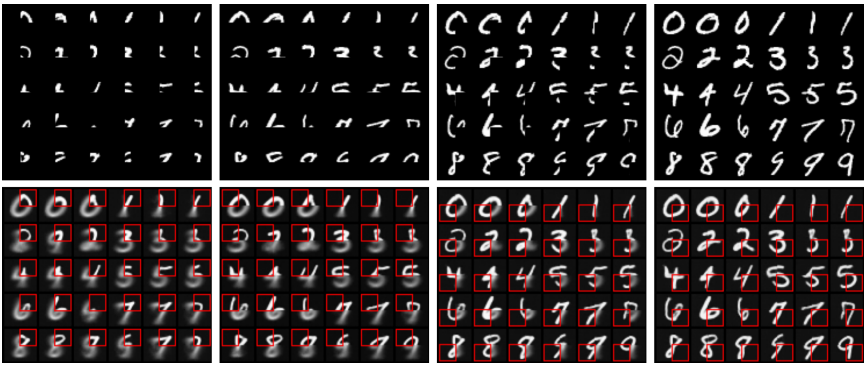


Figure 4.2: Available crops (top row) and corresponding reconstructions (bottom row) over four progressive iterations for thirty sample MNIST digits. The digits at $(\text{row}, \text{col}) = (2, 1)$, $(2, 2)$, $(4, 3)$ and $(5, 3)$ show the content refinement most clearly over time.

Figure 4.3: Progressive crops for six sample digits (top row) with their style-independent (middle row) and context-independent (bottom row) reconstructions.

After training the models, we can qualitatively evaluate our model behavior by performing inference over iterative context updates under multiple conditions: using context updates and style, using context updates but ignoring style during inference, and using content and style for inference but bypassing context update. Full reconstructions of test digits are shown in Figure 4.2. Moving from left to right, as more of the digits are seen, the content representation improves, and the reconstructions of both the unseen and seen crops improves.

Selected style-independent and context-independent reconstructions are shown in Figure 4.3. With no style information, the inference relies solely on the content. As more of each digit is seen, the content typically remains stable or improves and more accurately captures general shapes of the digits (middle row). Conversely, when the content is not updated, the reconstructions are much messier, and the general digit representation is unstable (bottom row).

With qualitative evidence supporting our method's behavior and performance, we now apply it to the problem of learning haptic representations from sequential EPs by a haptically sensitive robot. This cropped MNIST setting has similarities with our real-world data, as only partial information is perceived with every action. A difference is that in the synthetic example, we are interested only in digit identity, whereas in the haptic application we care about different object properties.

4.4 Experimental Setup

Our platform consists of a robotic arm and hand equipped with kinesthetic and tactile sensing. It can perform four programmed EPs (*drag*, *press*, *shake*, and *squeeze*) on a set of spherical objects that vary widely in size, stiffness, and mass. Some of the objects are also hollow and contain various media.

4.4.1 Hardware

We use a six-degree-of-freedom (dof) robot arm (Universal Robots UR5) with a wrist-mounted force-torque sensor (Weiss KMS 40) and a modified

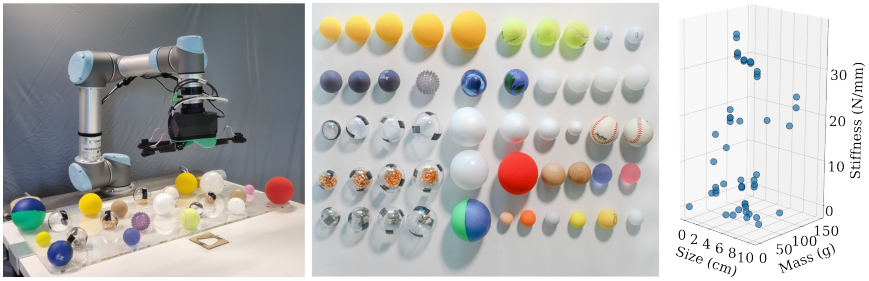


Figure 4.4: Robot with UR5 arm, customized Reflex Takktile 2 hand, and wrist-mounted KMS 40 six-axis force-torque sensor (left). The set of 52 objects (including empty space) used for the study (center). The distribution of manually measured object properties: size, mass, and stiffness (right).

three-fingered gripper (Right Hand Labs Reflex Takktile 2), as shown in Figure 4.4. The UR5 is position controlled and records position, velocity, and effort at 125 Hz for each of its six joints. The KMS 40 records three-axis force and three-axis torque at 500 Hz. The modified Reflex Takktile 2 gripper has three under-actuated fingers each with a customized inertial measurement unit (IMU) for estimating the distal joint angles, and one of the fingers has 14 tactile pressure sensors. The proximal joint positions are measured by encoders. Two of the fingers are coupled and can be rotated in opposite directions (called the preshape). Finally, we record the angular position and load from all four actuating motors in the hand. All hand data is sampled at 25 Hz. All control and data collection were run via ROS in Python and C++, building on the existing libraries for the UR5, KMS 40, and ReflexTakktile 2.

4.4.2 Exploratory Procedures (EPs)

The robot is able to perform four exploratory procedures (EPs). To diversify the robot's perceptual experiences, it begins each EP by dropping the objects into a central cardboard well that consists of a circle cut out of a sheet of cardboard that is adhered to the horizontal table surface. The ball can freely roll within the well, creating some randomness in its initial position.

- *Drag* (Figure 4.5a): The robot moves directly above the well where the object is held at one of three heights selected based on the size of the object. It widens the preshape joint to 0.7 radians and then slowly closes the fingers at 0.5 radians per second until each finger reaches a motor load of 150 (arbitrary digital units on a ten-bit scale), after which the finger position is fixed. The object is moved to a fixed location above the table and lowered at 1 cm/s until the force-torque sensor measures an increase of 1 N in the z-axis. The robot then begins recording data, and the object is dragged 5 cm horizontally across the surface at 3 cm/s in the direction of the single finger, such that the two preshape fingers are behind the object.
- *Press* (Figure 4.5b): To perform press, the robot moves into a fixed position above the well containing the object such that the two preshape fingers can be positioned directly over the center of the well. These two fingers are moved to predefined proximal angles, and the preshape is reduced such that the fingertips just touch. The robot begins recording data, and the gripper is lowered at 1 cm/s until the force-torque sensor measures an increase of 10 N in the z-axis, at which point the arm returns to the starting position at 2 cm/s. The predefined proximal pressing angles are determined by a calibration procedure where the robot moves the hand to a fixed position above the table and slowly closes each parallel finger until its motor load reaches 50.
- *Shake* (Figure 4.5c): The robot moves to a fixed position over the object and widens the preshape joint to 0.6 radians. It then slowly closes the fingers at 0.5 radians per second until each finger reaches a motor load of 120, after which the finger position is fixed. The robot then lifts the object to a fixed position such that the two preshape fingers are under the object. The robot begins recording data, and it then rapidly shakes the object four times at approximately 2 Hz by actuating the elbow and first wrist joints with sinusoidal acceleration with an amplitude of 10 rad/s^2.
- *Squeeze* (Figure 4.5d): The robot moves to a fixed position over the

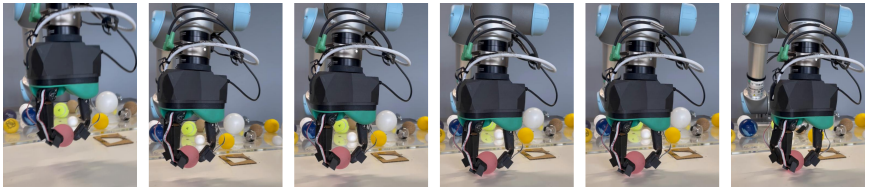
object, widens the preshape joint to 0.7 radians, starts recording data, and then slowly closes the fingers at 0.2 radians per second. Each finger closes until it reaches a motor load threshold of 150 and then maintains that motor load. After every finger has reached the threshold, the fingers open at 0.5 radians per second.

The EPs squeeze and press are standard human EPs [LK87], whereas shake has been used as an EP for robots [SWS09]. Because our robot does not have highly sensitive fingertips, we decided to drag the objects across the table instead of sliding a finger across the objects.

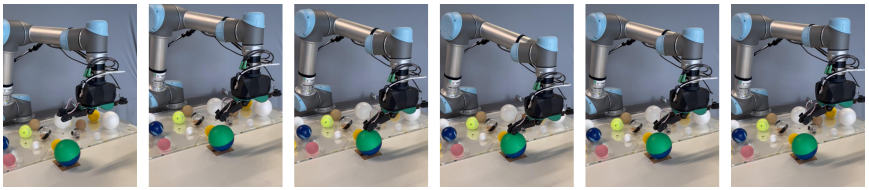
For all force thresholding performed on the data from the KMS 40 force-torque sensor, the threshold was applied to a low-pass-filtered version of the measured force to reduce the influence of high-frequency noise. Specifically, this filtered force was calculated by averaging the most recent 10 measurements, which were collected at 500 Hz. Before each EP, the tactile sensors were all recalibrated by subtracting their present readings, so that they all begin at zero. Furthermore, the IMU sensors were all recalibrated by multiplying by the inverse of their present quaternions.

4.4.3 Signal Processing

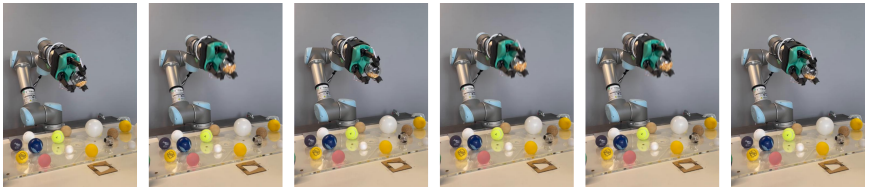
To prepare our data for the convolutional architecture, all signals were reduced to 25 Hz. The UR5 data were downsampled using the scipy `decimate` function. To capture both the transient, high-frequency vibrations and the signal magnitude, force and torque data were separated into AC and DC components using the scipy `spectrogram` (window size $= 40$ points, overlap $= 20$ points) and `decimate` functions. Each interaction in the dataset was then either cut or padded to the same duration; the standard duration of 400 points (16 seconds) was determined by subtracting the earliest moment of contact across all *presses* from the latest moment of contact across all *presses*. All *presses* and *squeezes* were cropped to these time points, and the shorter *shakes* and *slides* were padded on the end by repeating the values of the last recorded point.



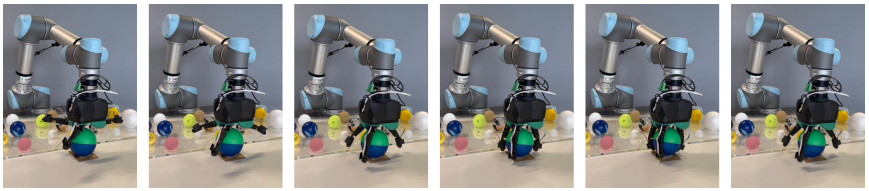
(a) *Drag*



(b) *Press*



(c) *Shake*



(d) *Squeeze*

Figure 4.5: Sequences of still images captured during each of the robot's four EPs.

4.4.4 Objects

We designed a set of 52 objects (Figure 4.4) to test our approach on learning comprehensive latent representations of haptic properties from sequences of interactions. All objects needed to be graspable and liftable by our selected robot platform. To minimize the influence of object orientation on this investigation, we decided to use only spherical objects. Most are purchased sports balls, and some of the others are either hollow or filled spherical shells. Though uniform in shape, the 51 selected items have a range of physical properties, differing significantly in size, stiffness, mass, and filling. The final object is empty space, giving a full set of 52 things the robot can drag, press, shake, and squeeze.

4.5 Experiments

Our experiments were designed to evaluate the behavior and performance of our method. Specifically, how does the latent content representation change over iterations? How well are the haptic properties of size, stiffness, mass, and filling encoded in the latent content space? And how is information distributed between the content and style latent spaces?

4.5.1 Training Procedure

In total, the robot performed each EP 50 times on each object, for a total of 10 400 trials. We trained and validated our models on 60% of the data and tested on the remaining 40%. Our encoder and decoder networks are composed of four 1D convolutional layers and two dense layers, and we use ten dimensions for both the content and style vectors. When training the models, we generate new random sequences of four EPs from the same object (as in the MNIST example) at the beginning of every epoch.

4.5.2 Evaluation

Qualitative Haptic Property Representation To determine how well haptic properties are captured in the latent content representation, we generated random sequences of four EPs from our test data, passed them through the full iterative modeling process, and then compressed the ten-dimensional content space into two dimensions using t-distributed stochastic neighbor embedding (t-SNE). We then visualize the distribution of each haptic property in the compressed latent space.

Object and Property Classification To determine how the content representation improves over iterations and how well the properties and objects are represented in the latent representation, we train models on content and style representations to classify the objects and their four haptic properties. We generate random four-EP sequences from the training dataset and use the final-iteration content and style representations as our training and validation data for the classifiers. We similarly generate random four-EP sequences from the test dataset and evaluate the classification performance on content and style *after each iteration* through the Group VAE. For each of the five labels, we train a total of 50 classification models. To label the objects by size, stiffness, and mass, the objects are grouped into 10 clusters via k-means clustering.

4.5.3 Implementation Details

4.5.3.1 Sequential Group VAE Implementation

The Group VAE was implemented using the PyTorch library. The models were all trained using the Adam optimizer [KB15] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and learning rate $= 0.001$.

Instead of using the standard VAE for our architecture, we use the β-VAE, which introduces the regularization parameter β into Equation 4.5 that allows for tuning the trade-off between the negative loglikelihood and the KL-divergence [HMP+17]. We implement separate regularization parameters

β_C and β_S for content and style. The Group VAE was trained with the following parameters:

- MNIST: batch size$= 256$, $\beta_C = 0.01$, $\beta_S = 0.01$, 5 latent dimensions each for content and style.
- Robot Data: batch size$= 32$, $\beta_C = 0.033$, $\beta_S = 0.066$, 10 latent dimensions each for content and style.

Robot Data The encoder network consists of four 1D convolutional layers followed by two dense layers, and it outputs the content and style mean and variance, which parameterize $q_{\phi_c}\left(\mathbf{c}^{(t)} \mid x^t\right)$ and $q_{\phi_s}\left(s^t \mid x^t\right)$. The four convolutional layers have kernel sizes and strides of $\{6, 6, 5, 4\}$ and $\{4, 4, 2, 2\}$, with $\{128, 64, 64, 32\}$ filters. Each layer is followed by batch normalization and leaky rectified linear unit (Leaky ReLU) activation functions. The output of the convolution layers is flattened and concatenated with the previous content mean and variance. This concatenated vector of size e_d is the input to the first dense layer, which has size $e_d \times 100$ and is followed by the Leaky ReLU activation function. The output of this layer is fed into four separate dense layers, one for each of the content mean and variance and style mean and variance. The outputs of these four layers parameterize $q_{\phi_c}\left(\mathbf{c}^{(t)} \mid x^t,\ q_{\phi_c}^{(t-1)}\right)$ and $q_{\phi_s}\left(s^t \mid x^t,\ q_{\phi_c}^{(t-1)}\right)$.

We then sample a content vector c and a style vector s from these distributions and concatenate them with a one-hot encoding a of the corresponding EP. This vector is then fed into the decoder. The decoder is the reverse of the encoder except for the slightly larger input to the first layer (to accommodate a) and the final deconvolutional layer, which is split into two equivalent layers whose outputs parameterize the mean and variance of the normal distribution $p_\theta(x \mid c,\ s,\ a)$.

Cropped MNIST [LBBH98] The encoder and decoder architectures for modeling the cropped MNIST data are very similar to those used for the robot data. The differences are that the convolutions are 2D, all four convolutional layers use 32 filters with a kernel size of 3 and stride of 2, and the dense layers have 50 hidden nodes instead of 100.

Additional Implementation Notes We slightly alter the loss function from Equation 4.5. Instead of computing the style-KL term for all the style vectors $\mathbf{s}^{(t)}$, we compute it only for the most recent style vector s^t. Additionally, the loss is back-propagated through s^t only at iteration t. For all future iterations it is used as a constant, detached from the back-propagation graph, to condition the generative model $p_\theta(x \mid \mathbf{c}, s^t, a)$

4.5.3.2 Classifier Implementation

We implemented relatively small neural networks to perform classification of the objects and their properties. For the object properties we used networks with one hidden layer with 10 nodes and a Leaky ReLU activation function. To classify the objects, the hidden layer had 50 nodes.

The models were all trained with a batch size of 32 using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate $= 0.0001$, and weight decay $= 1$.

4.6 Results

4.6.1 Latent Embedding Visualization

Figure 4.6 shows the results of the qualitative latent embedding. For visualization purposes, we plotted only 10 random embeddings per object. There is clearly structure encoded in the latent space for all haptic properties. Size and mass are highly separable in just these two dimensions. Interestingly, most of the filled objects are clustered together but are indistinguishable.

4.6.2 Object and Property Classification in Content and Style

The results of the classification are shown in Figure 4.7. The high classification accuracies achieved on the content representations indicate that general properties of the objects are indeed encoded in content. Conversely, the style contains very little content information and performs barely better

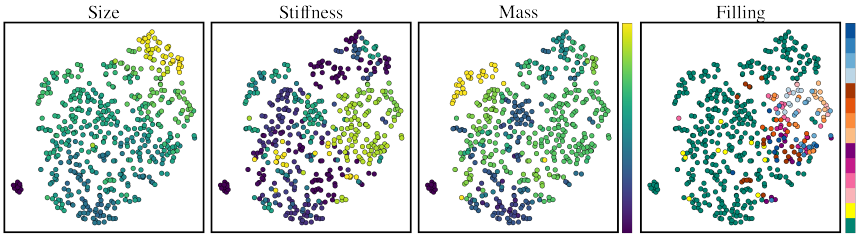


Figure 4.6: Final-step content latent-space embeddings of 10 sequences per object, as visualized in 2D using t-SNE. The points in each plot are colored to show the respective property value: size, stiffness, mass, and the filling of the object. For the three continuous properties, higher values are indicated by the colors at the top of the color bar. Similar fillings are indicated by shades of the same color: stones are blue, corn kernels are orange, and salt grains are purple. Here, green represents objects that have no loose filling, and yellow indicates a single object containing a loose solid mass. The dense cluster at the lower left is empty space.

than chance; this result is expected because style is designed to capture only trial-specific variations.

Additionally, the classification on the test set content improves dramatically as a function of sequence iteration, indicating that information about different properties is accumulated over sequences of EPs that each elicit incomplete information. This strong improvement occurs with content while the style classification performance improves only incrementally, again implying that most general group information is contained in the content. The rapid improvement over a small number of observations is also consistent with the low number of samples that the ML-VAE needs to perform accurate MNIST classification [BTN18].

To better understand why classification performance on the objects ("Ball ID" in Figure 4.7) was relatively poor compared to the properties, we used a confusion matrix to visualize which objects were difficult to classify (Figure 4.8). Unsurprisingly, objects of the same type (e.g., baseballs 1 and 2 and the three tennis balls) are confused by the classifier. Additionally, it

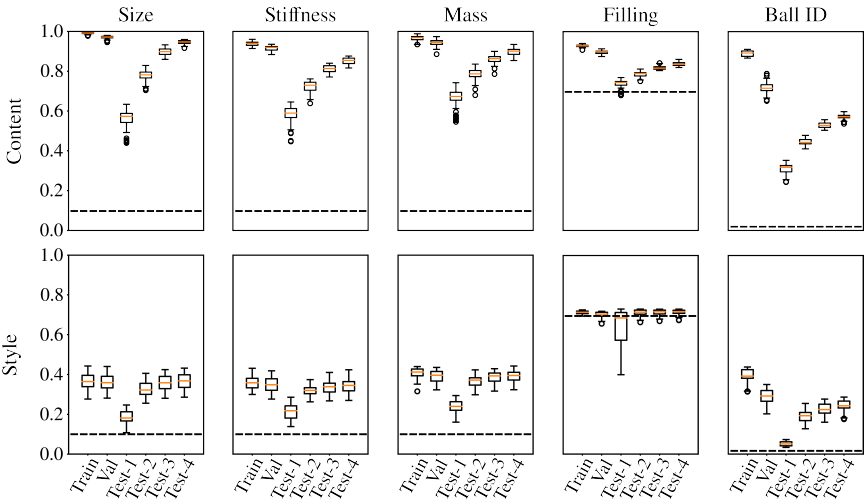


Figure 4.7: Distributions of classification results for individual properties using the content (top) and style (bottom) representations. As expected, object properties are strongly represented in the content latent space and only weakly represented in the style latent space. Chance performance is indicated by the dashed lines.

appears that the classifier has difficulty detecting the difference between the filler materials, as the objects of the same size that are filled with different materials are frequently confused. Perhaps this confusion could be resolved with high-frequency tactile sensing.

4.6.3 Latent Representation Variance

An interesting result of the evidence accumulation strategy of the Multi-Level VAE is that as more evidence is accumulated, the variance of the group content distribution decreases [BTN18]. While that variance reduction is not analytically derived from our formulation, our method seems to nonetheless demonstrate a similar property. We generated multiple random sequences for each trial and measured the variances of the latent content and style representations for each of those encoded sequences at each iteration. Fig-

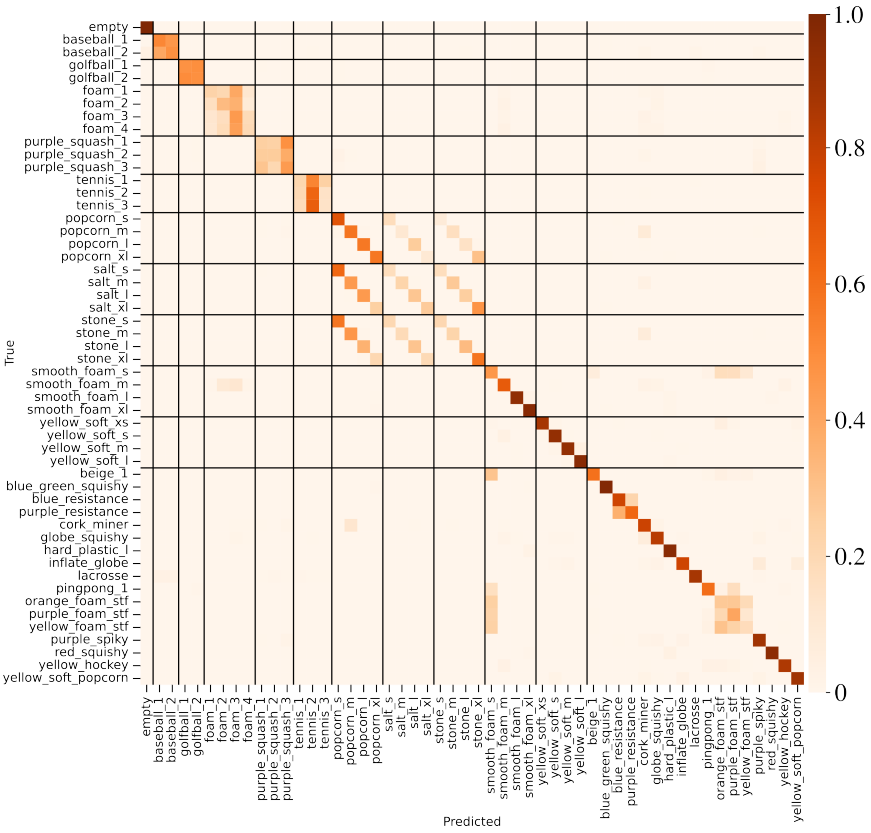


Figure 4.8: Confusion matrix of classification results. Objects of a similar type are grouped together. There is some confusion between objects of the same type, and there is confusion between the three filling materials (popcorn, salt, and stone).

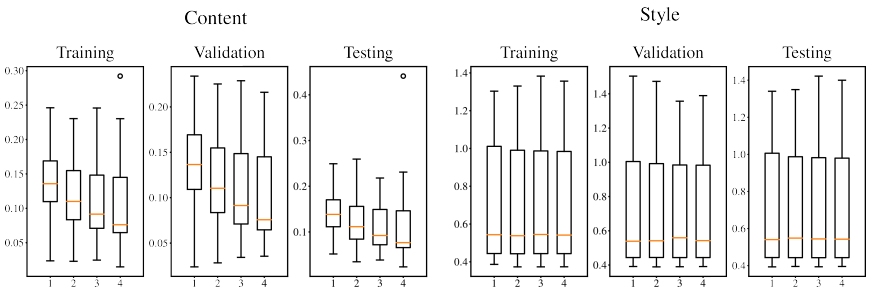


Figure 4.9: Distributions of the average standard deviations across the ten dimensions of content and style latent representations as a function of sequence iteration. Standard deviation (and therefore variance) in the content representation decreases as a function of sequence iteration. Style variance doesn't change at all.

ure 4.9 shows the resulting distributions over thousands of sequences of the average standard deviations across all ten content and style dimensions as a function of sequence iteration. For the content representations, the average standard deviation decreases as a function of sequence iteration for all three of the training, validation, and testing sets. Conversely, there is no increase or decrease in standard deviation in the style representations. These findings reinforce our conclusion that the content representations are capturing meaningful information about the essential properties of the objects being touched, and that this information is being accumulated over multiple interactions. Additionally, the style representations encode mainly trial-to-trial variability and do not accumulate information over interactions.

4.7 Summary

In this chapter we presented an iterative Group VAE for learning and updating latent representations of haptic data as a robot sequentially explores and physically interacts with objects. After validating our method on a modified MNIST dataset, we used a tactilely sensitive robot platform interacting with

52 spherical objects to demonstrate that our method accumulates general group information as it iterates through sequences of exploratory procedures (EPs).

We investigated the learned representations and found that they are predictive of relevant object properties, such as size, stiffness, mass, and filling. Thus, we believe our method is a valuable addition to unsupervised robot learning because it finds representations that are important for future downstream tasks from sequential multimodal haptic sensations.

Though promising, this work has some limitations. The set of 52 objects is quite simple and differs along only a small set of physical and semantic properties. Additionally, the robot's set of four EPs should be expanded by randomly parameterizing the EPs or by allowing the robot to optimize across parameterizations of the EPs. Finally, the robot's sensing capabilities were limited because only one of its fingers includes functional tactile sensors. Furthermore, none of the robot's current fingertip sensors can capture the rich high-frequency vibrations that are likely to be important for perceiving object texture during *drag* actions or loose filling during *shake*.

Although our method could be extended to longer sequences with repeated EPs, we trained only on sequences of length four with one of each EP. We did not test how training repeatedly on the same action influences the learned model; we expect repetition should reduce uncertainty in relevant object properties, but not along dimensions for which no information is gained. We also did not evaluate how each EP influences the update of the latent content representation. These directions should be explored in future work to build on and test the limits of the promising initial results.

Chapter 5

Conclusions

The work presented in this thesis is directed towards developing a deeper understanding of how humans perceive complex haptic interactions and leveraging that understanding to endow robots with similar capabilities. Such robots would be able to explore objects, gather information over time, and develop general haptic representations that inherently capture properties also relevant to human perception. The methods are designed to avoid some of the most common assumptions made in traditional perceptual studies and robot haptic learning and instead provide new approaches to exploring fundamental questions in haptics research. This chapter summarizes the motivating problems and contributions of the individual chapters and provides an outlook on future work.

Chapter 2: *Modeling Human Haptic Perception from Unconstrained Surface Exploration*

The work in Chapter 2 focuses on understanding how individual people make perceptual judgments of specific haptic interactions. The traditional approach to modeling human perception has been to ask people to rank or

cluster various sample objects or surfaces along specific dimensions such as roughness. Various physical properties of the samples can be measured separately and then correlated with the given ranks, or the samples can be embedded in property space and transformed to most closely resemble the clusters or ranks provided by the human participants. However, these methods inherently do not take any interaction-specific details into account even though people are often free to explore samples however they wish. To address this gap, we proposed a method that learns to model individual and average group perception and predict perceived surface similarity directly from physical interaction data. The primary contributions of this work are as follows:

- Parsing interactions into successive exploratory procedures allows us to consider the contribution of each EP separately in our modeling process and assess how each one impacts perception.
- By parsing the signals into short time windows, we can treat the interactions as probability distributions over the property space of surfaces. This conceptualization provides a versatile tool for both measuring perception of specific interactions and potentially integrating more complex perceptual weighting schemes.
- We introduced a pipeline that can embed and estimate perception of individual interactions while training on only rank-ordered noisy labels provided by human study participants. Given that many haptic perception studies use discrete subjective ratings or ordering methods, we believe that other researchers can use our method as a bridge between modeling of measured signals and human perceptual measurements.
- By using simple embedding functions, we are able to study how individual surface properties are scaled and aligned in the perceptual representations of individual participants.

We demonstrated our approach by learning to predict perceptual representations for multiple participants, generalizing to unseen participants, and

fitting to individual participants. Additionally, we made progress toward interpreting how individual people perceptually represent haptic interactions.

Although promising, the work was limited in ways that we believe negatively impacted the overall performance of the method. These limitations include a small number of participants, surfaces, and chosen features. For the sake of interpretability, we also avoided full end-to-end learning. Additionally, it is possible that perceptual representations can shift between individual explorations. Addressing these different problems can lead to a more comprehensive understanding of the relationship between physical interactions and perception. Nonetheless, our method provides a novel framework with which to explore human haptic perception.

Chapter 3: *Unsupervised Feature Learning for Predicting Human Perception of Haptic Properties*

Chapter 3 focuses on how to provide robots with haptic perception that is more comparable to human perception. Haptic robotics research has primarily been informed by human studies, in which certain properties of recorded physical signals are correlated with semantic attributes. When a robot interacts with an object, these same properties can be used to predict semantic properties of that object. However, this approach limits a robot's perception in two important ways: potentially useful information in the signal is simply discarded, and the robot has to judge objects according to predefined semantic properties. We proposed methods to address both of these limitations, one which fully addresses the issue inherent in using handcrafted features and one that, while not allowing a robot to define its own properties, provides a robot more flexibility in determining the intensity of a particular attribute. The primary contributions of this work are as follows:

- We introduced a full pipeline for unsupervised feature extraction from diverse haptic data streams, using K-SVD dictionary learning for 1D and 2D data. These features demonstrated great effectiveness across multiple different classification and regression tasks.

- We presented and analyzed ordinal adjective labels that human study participants applied to various objects that they touched. Our analyses provide unique evidence for the relationships between various adjectives and demonstrate a great subjectivity in human perception that supports the need for more robust and adaptable robot perception algorithms.
- Our modified ordinal regression method provides a simple way to encapsulate diverse and scaled perceptual experiences into a single model which can then be used to predict intensity distributions for various semantic object properties.

We demonstrated our approaches on many different tasks for a variety of robotic exploratory procedures. First, the learned features were evaluated against and outperformed the baseline achieved with a large set of expertly hand-crafted features on 19 binary classification tasks, demonstrating that the assumptions made by using hand-crafted features sacrifice relevant information contained in haptic signals. Second, these features were used in ten regression tasks to demonstrate the ordinal regression method. Although the learned features were successful across all tasks, the performance did vary by exploratory procedure. These results demonstrate that using unsupervised feature detection can capture relevant physical data, and that robots can be trained on crowd-sourced perceptual ratings to develop more robust haptic representations that are more adaptable to real-world situations. However, to develop comprehensive models of individual objects, robots needs to be able to accumulate information across a multitude of exploratory procedures.

Chapter 4: *Implicit Robot Learning of Haptic Properties from Sequential Interactions*

In Chapters 2 and 3, we focused on modeling sensor and robot haptic data from human descriptions of various objects and surfaces. While the human surface similarity modeling included two different exploratory procedures, it did not create explicit surface representations that changed with new

information from additional exploration. Additionally, it relies on human comparisons to form a model and is thus not particularly suited for a robot to perform autonomous exploration. Predicting human adjective descriptions from robot exploration is a step in the right direction, but it still requires human labeling and provides no way to efficiently combine data from multiple exploratory procedures. The work presented in Chapter 4 seeks to address these limitations with a novel learning method to provide a robot with all the necessary tools to learn robust and comprehensive object representations from scratch as it explores the world. The primary contributions of this work are as follows:

- We developed a novel algorithm based on Multi-Level VAEs [BTN18] that leverages object permanence to accumulate data gathered over any number of exploratory procedures to build comprehensive object representations, updating the representations as more information is gathered.
- We collected a new dataset using a haptically sensitive robot that used four diverse exploratory procedures to probe objects. The robot performed many explorations of approximately 50 spherical objects that differ substantially across several salient dimensions. Our method was evaluated on this new dataset.
- The multi-level latent structure naturally learned to use one of the latent spaces to represent general object properties and the other to capture random variations within single exploratory procedures.
- The accumulation of evidence naturally caused a decrease in the variance of the posterior estimate, in particular with regards to the latent space representing object properties.

Our proposed method was able to quickly accumulate information about a variety of object properties that it was never explicitly instructed to learn. We demonstrated that our method naturally exhibits desirable properties, is able to build representations over time in an efficient recursive manner, and

doesn't need to make strong assumptions like previously proposed haptic evidence-accumulation methods.

Outlook

The gap between the current, general understanding of human haptic perception and the development of highly specialized robotic haptic applications offers fertile ground on which to explore fundamental properties of haptic perception from both the human and robot perspective. Although the work presented in this thesis is an initial effort into exploring this gap, it leaves many unanswered and new questions that can be explored in future work.

To understand human haptic perception, is it sufficient to measure general surface properties and correlate those with noisy human ratings, or do we need a more fundamental approach that captures the individual particularities of how humans extract relevant haptic information? Our work into human perception offers this alternative latter approach, and we believe it opens the door into a deep new research direction. Continuing work could begin by formulating more controlled experiments that encourage or force participants to employ particular judgment strategies when they explore surfaces. For example, one could design surfaces or textures that differ on a very fine scale that can be distinguished only by gentle and slow exploration. Alternatively, participants could be made to explore in different fixed patterns, and various models of rating strategies could be tested against the perceptual results. Another direction could be to try to measure how similarly surfaces are perceived at different contact forces and velocities, and test whether those similarities or dissimilarities can predict which type of rating strategy is used. This is just a sampling of what can be explored using this type of framework.

In this thesis, we also worked on designing a robot perceptual system that can achieve some aspects of human perception, mainly object continuity and the ability to accumulate information over time. However, our implementation is still limited and our overall approach under-explored.

For example, how many explorations are sufficient to develop a comprehensive understanding of an object? As the complexity and number of objects increases, will this method scale? We know that humans are capable of this generalization, so what additional modeling capability does a robot need to approach our abilities? As it currently stands, our method could be incorporated into a more general autonomous framework where a robot can intelligently explore objects to learn necessary properties to perform particular tasks. How might those learned representations differ if they are learned independently of or jointly with any real-world tasks? Additionally, the hand sensors used in our experiments were quite simplistic. By using a more comprehensive set of sensors, more dynamic and transient data could be gathered and used to learn new object properties. Continuing to improve on this research could lead to exciting new improvements in the ability of robots to operate autonomously in the world.

Bibliography

[ADT+17] A. Abdouni, M. Dhaghloul, C. Thieulin, R. Vargiolu, C. Pailler-Mattei, H. Zhaouani. 'Biophysical properties of the human finger for touch comprehension: influences of ageing and gender'. In: 4.8 (2017) (cit. on p. 21).

[AEB06] M. Aharon, M. Elad, A. Bruckstein. 'K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation'. In: *IEEE Transactions on Signal Processing* 54.11 (Nov. 2006), pp. 4311–4322 (cit. on pp. 58, 59, 66).

[AGCC18] Z. Abderrahmane, G. Ganesh, A. Crosnier, A. Cherubini. 'Haptic Zero-Shot Learning: Recognition of objects never touched before'. In: *Robotics and Autonomous Systems* 105 (Mar. 2018), pp. 11–25 (cit. on pp. 59, 102).

[Agr02] A. Agresti. *Categorical Data Analysis*. Vol. 482. John Wiley & Sons, 2002 (cit. on p. 80).

[ARSD17] M. Arvidsson, L. Ringstad, L. Skedung, K. Duvefelt. 'Feeling fine - the effect of topography and friction on perceived roughness and slipperiness'. In: *Biotribology* 11 (2017), pp. 92–101 (cit. on p. 27).

[Ben12] Y. Bengio. 'Deep learning of representations for unsupervised and transfer learning'. In: *Proceedings of the ICML Workshop on Unsupervised and Transfer Learning*. PMLR, June 2012, pp. 17–36 (cit. on pp. 57, 59).

[BES09] S. Baccianella, A. Esuli, F. Sebastiani. 'Evaluation Measures for Ordinal Regression'. In: *Ninth International Conference on Intelligent Systems Design and Applications*. 2009, pp. 283–287 (cit. on p. 85).

[BK06] W. M. Bergmann Tiest, A. M. L. Kappers. 'Analysis of haptic perception of materials by multidimensional scaling and physical measurements of roughness and compressibility'. In: *Acta Psychologica* 121.1 (2006), pp. 1–20 (cit. on pp. 22, 23, 27, 49, 60).

[BK07] W. M. Bergmann Tiest, A. M. .-. Kappers. 'Haptic and visual perception of roughness'. In: *Acta Psychologica* 124.2 (2007), pp. 177–189 (cit. on p. 47).

[BK17] A. Burka, K. J. Kuchenbecker. 'Handling Scan-time Parameters in Haptic Surface Classification'. In: *Proceedings of the IEEE World Haptics Conference*. June 2017, pp. 424–429 (cit. on p. 56).

[BRF11] L. Bo, X. Ren, D. Fox. 'Hierarchical matching pursuit for image classification: Architecture and fast algorithms'. In: *Proceedings of the 24th International Conference on Neural Information Processing Systems*. Dec. 2011, pp. 2115–2123 (cit. on p. 59).

[BRK18a] T. Bhattacharjee, J. M. Rehg, C. C. Kemp. 'Inferring object properties with a tactile-sensing array given varying joint stiffness and velocity'. In: *International Journal of Humanoid Robotics* 15.01 (2018), p. 1750024 (cit. on p. 102).

[BRK18b] T. Bhattacharjee, J. M. Rehg, C. C. Kemp. 'Inferring Object Properties with a Tactile-Sensing Array Given Varying Joint Stiffness and Velocity'. In: *International Journal of Humanoid Robotics* 15.01 (2018), p. 1750024 (cit. on p. 78).

[BSB93] A. Bicchi, K. Salisbury, L. Brock. 'Contact sensing from force measurements'. In: *International Journal of Robotics Research* 12.3 (1993), pp. 249–262 (cit. on p. 27).

[BTBD20] M. Blondel, O. Teboul, Q. Berthet, J. Djolonga. 'Fast differentiable sorting and ranking'. In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2020, pp. 950–959 (cit. on p. 35).

[BTN18] D. Bouchacourt, R. Tomioka, S. Nowozin. 'Multi-level variational autoencoder: Learning disentangled representations from grouped observations'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018 (cit. on pp. 101, 103–105, 120, 121, 129).

[CHS+16] Y. Chebotar, K. Hausman, Z. Su, G. S. Sukhatme, S. Schaal. 'Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning'. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Oct. 2016, pp. 1960–1966 (cit. on p. 70).

[CK17a] H. Culbertson, K. J. Kuchenbecker. 'Importance of Matching Physical Friction, Hardness, and Texture in Creating Realistic Haptic Virtual Surfaces'. In: *IEEE Transactions on Haptics* 10.1 (2017), pp. 63–74 (cit. on pp. 22, 23, 27, 29).

[CK17b] H. Culbertson, K. J. Kuchenbecker. 'Ungrounded Haptic Augmented Reality System for Displaying Roughness and Friction'. In: *IEEE/ASME Transactions on Mechatronics* 22.4 (2017), pp. 1839–1849 (cit. on p. 27).

[CKS+16] L. Cao, R. Kotagiri, F. Sun, H. Li, W. Huang, Z. M. M. Aye. 'Efficient spatio-temporal tactile object recognition with randomized tiling convolutional networks in a hierarchical fusion strategy'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2016 (cit. on p. 103).

[CLK14] H. Culbertson, J. J. López Delgado, K. J. Kuchenbecker. 'One Hundred Data-Driven Haptic Texture Models and Open-Source Methods for Rendering on 3D Objects'. In: *Proc. IEEE Haptics Symposium*. Feb. 2014, pp. 319–325 (cit. on p. 24).

[CMR+13] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell, K. J. Kuchenbecker. 'Using robotic exploratory procedures to learn the meaning of haptic adjectives'. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. 2013, pp. 3048–3055 (cit. on pp. 62, 102).

[CMR+15] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, T. Darrell, K. J. Kuchenbecker. 'Robotic learning of haptic adjectives through physical interaction'. In: *Robotics and*

Autonomous Systems 63 (Jan. 2015), pp. 279–292 (cit. on pp. 57–60, 62, 64, 72, 74, 75, 77, 95, 102).

[CMR+16] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, T. Darrell, K. J. Kuchenbecker. 'Corrigendum to "Robotic learning of haptic adjectives through physical interaction" [Robot. Auton. Syst. 63 (P3) (2015) 279–292]'. In: *Robotics and Autonomous Systems* 83 (Sept. 2016), p. 349 (cit. on p. 74).

[CSDB15] T. Callier, H. P. Saal, E. C. Davis-Berg, S. J. Bensmaia. 'Kinematics of unconstrained tactile texture exploration'. In: *Journal of Neurophysiology* 113.7 (2015), pp. 3013–3020 (cit. on p. 23).

[Cut13] M. Cuturi. 'Sinkhorn distances: Lightspeed computation of optimal transport'. In: *Advances in Neural Information Processing Systems* 26 (2013), pp. 2292–2300 (cit. on p. 33).

[CZHL15] Z. Chen, W. Zuo, Q. Hu, L. Lin. 'Kernel sparse representation for time series classification'. In: *Information Sciences* 292 (Jan. 2015), pp. 15–26 (cit. on p. 69).

[DBE+16] B. Delhaye, A. Barrae, B. B. Edin, P. Lefèvre, J. L. Thonnard. 'Surface strain measurements of fingertip skin under shearing'. In: *Journal of the Royal Society Interface* 13.115 (2016), p. 20150874 (cit. on p. 21).

[DDS+09] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei. 'ImageNet: A large-scale hierarchical image database'. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255 (cit. on p. 56).

[DGÉC14] P. Dallaire, P. Giguère, D. Émond, B. Chaib-Draa. 'Autonomous tactile perception: A combined improved sensing and Bayesian nonparametric approach'. In: *Robotics and Autonomous Systems* 62.4 (2014), pp. 422–435 (cit. on pp. 100, 103).

[DWCK18] K. Drewing, C. Weyel, H. Celebi, D. Kaya. 'Systematic Relations between Affective and Sensory Material Dimensions in Touch'. In: *IEEE Transactions on Haptics* 11.4 (2018), pp. 611–622 (cit. on p. 22).

[ETA+13] G. Elkharraz, S. Thumfart, D. Akay, C. Eitzinger, B. Henson. 'Making tactile textures with predefined affective properties'. In: *IEEE Transactions on Affective Computing* 5.1 (2013), pp. 57–70 (cit. on p. 22).

[FKPC21] R. F. Friesen, R. L. Klatzky, M. A. Peshkin, J. E. Colgate. 'Building a navigable fine texture design space'. In: *IEEE Transactions on Haptics* 14.4 (2021), pp. 897–906 (cit. on pp. 22, 49).

[FL12] J. Fishel, G. Loeb. 'Bayesian Exploration for Intelligent Identification of Textures'. In: *Frontiers in Neurorobotics* 6 (June 2012) (cit. on pp. 22, 28, 56, 100, 102, 103).

[FMS19] C. Frogner, F. Mirzazadeh, J. Solomon. 'Learning Embeddings into Entropic Wasserstein Spaces'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2019 (cit. on p. 33).

[FS19] R. W. Fleming, K. R. Storrs. 'Learning to see stuff'. In: *Current Opinion in Behavioral Sciences* 30 (2019), pp. 100–108 (cit. on pp. 47, 52).

[GBC16] I. Goodfellow, Y. Bengio, A. Courville. *Deep Learning (Adaptive Computation and Machine Learning series)*. Cambridge, MA: The MIT Press, 2016 (cit. on pp. 56, 57).

[GBGB05] G. A. Gescheider, S. J. Bolanowski, T. G. Greenfield, K. E. Brunette. 'Perception of the Tactile Texture of Raised-Dot Patterns: A Multidimensional Analysis'. In: *Somatosensory and Motor Research* 22.3 (2005), pp. 127–140 (cit. on p. 28).

[GDM+10] S. Guest, J. M. Dessirier, A. Mehrabyan, F. McGlone, G. Essick, G. Gescheider, A. Fontana, R. Xiong, R. Ackerley, K. Blot. 'The development and validation of sensory and emotional scales of touch perception'. In: *Attention, Perception, & Psychophysics* 73.2 (2010), pp. 531–550 (cit. on pp. 64, 93).

[Gib62] J. J. Gibson. 'Observations on active touch.' In: *Psychological Review* 69.6 (1962), p. 477 (cit. on p. 100).

[GPS+16] P. A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernández-Navarro, C. Hervás-Martínez. 'Ordinal Regression Methods: Survey and Experimental Study'. In: *IEEE Transactions on Knowledge and Data Engineering* 28.1 (2016), pp. 127–146 (cit. on pp. 78, 83).

[GS14] M. C. Gemici, A. Saxena. 'Learning haptic representation for manipulating deformable food objects'. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2014, pp. 638–645 (cit. on pp. 100, 102, 103).

[GTH14] P. A. Gutiérrez, P. Tiňo, C. Hervás-Martínez. 'Ordinal regression neural networks based on concentric hyperspheres'. In: *Neural Networks* 59 (2014), pp. 51–60 (cit. on p. 82).

[HBKY00] M. Hollins, S. Bensmaïa, K. Karlof, F. Young. 'Individual differences in perceptual space for tactile textures: Evidence from multidimensional scaling'. In: *Perception & Psychophysics* 62.8 (2000), pp. 1534–1544 (cit. on pp. 22, 60, 66).

[HFRY93] M. Hollins, R. Faldowski, S. Rao, F. Young. 'Perceptual dimensions of tactile surface texture: A multidimensional scaling analysis'. In: *Perception & Psychophysics* 54.6 (1993), pp. 697–705 (cit. on pp. 22, 28, 60).

[HJ06] H.-N. Ho, L. Jones. 'Contribution of thermal cues to material discrimination and localization'. In: *Perception & Psychophysics* 68.1 (2006), pp. 118–128 (cit. on p. 48).

[HMP+17] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, A. Lerchner. 'beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017 (cit. on p. 117).

[HPH15] J. Hoelscher, J. Peters, T. Hermans. 'Evaluation of tactile feature extraction for interactive object recognition'. In: *IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. Nov. 2015, pp. 310–317 (cit. on p. 57).

[HS06] G. E. Hinton, R. R. Salakhutdinov. 'Reducing the Dimensionality of Data with Neural Networks'. In: *Science* 313.5786 (2006), pp. 504–507 (cit. on pp. 57, 59).

[IVB19] A. Isleyen, Y. Vardar, C. Basdogan. 'Tactile Roughness Perception of Virtual Gratings by Electrovibration'. In: *IEEE Transactions on Haptics* 13.3 (2019), pp. 562–570 (cit. on p. 22).

[JF09] R. S. Johansson, J. R. Flanagan. 'Coding and use of tactile signals from the fingertips in object manipulation tasks'. In: *Nature Reviews Neuroscience* 10.5 (Apr. 2009), pp. 345–359 (cit. on p. 21).

[KB15] D. P. Kingma, J. Ba. 'Adam: A Method for Stochastic Optimization'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. Ed. by Y. Bengio, Y. LeCun. 2015. URL: http://arxiv.org/abs/1412.6980 (cit. on p. 117).

[KLM85] R. L. Klatzky, S. J. Lederman, V. A. Metzger. 'Identifying objects by touch: An "expert system"'. In: *Perception & Psychophysics* 37.4 (1985), pp. 299–302 (cit. on pp. 56, 100).

[KLR87] R. L. Klatzky, S. J. Lederman, C. Reed. 'There's more to touch than meets the eye: The salience of object attributes for haptics with and without vision.' In: *Journal of Experimental Psychology: General* 116.4 (1987), pp. 356–369 (cit. on p. 57).

[KPMR18] S. Kolouri, P. E. Pope, C. E. Martin, G. K. Rohde. 'Sliced Wasserstein auto-encoders'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018 (cit. on p. 33).

[KSG+19] M. Kerzel, E. Strahl, C. Gaede, E. Gasanov, S. Wermter. 'Neuro-robotic haptic object classification by active exploration on a novel dataset'. In: *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. 2019, pp. 1–8 (cit. on pp. 101, 102).

[KW14] D. P. Kingma, M. Welling. 'Auto-encoding variational bayes'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2014 (cit. on p. 104).

[KWT+13] A. Klocker, M. Wiertlewski, V. Theate, V. Hayward, J. L. Thonnard. 'Physical factors influencing pleasant touch during tactile exploration'. In: *Plos One* 8.11 (2013), pp. 1–8 (cit. on p. 27).

[LAC17] N. F. Lepora, K. Aquilina, L. Cramphorn. 'Exploratory tactile servoing with active touch'. In: *IEEE Robotics and Automation Letters* 2.2 (2017), pp. 1156–1163 (cit. on pp. 100, 103).

[LaM00] R. H. LaMotte. 'Softness discrimination with a tool'. In: *Journal of Neurophysiology* 83.4 (2000), pp. 1777–1786 (cit. on pp. 29, 102).

[LBBH98] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner. 'Gradient-based learning applied to document recognition'. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324 (cit. on pp. 107, 118).

[LBDL17] S. Luo, J. Bimbo, R. Dahiya, H. Liu. 'Robotic tactile perception of object properties: A review'. In: *Mechatronics* 48 (2017), pp. 54–67 (cit. on p. 16).

[LK09] S. J. Lederman, R. L. Klatzky. 'Haptic perception: A tutorial'. In: *Attention, Perception, & Psychophysics* 71.7 (2009), pp. 1439–1459 (cit. on p. 16).

[LK87] S. J. Lederman, R. L. Klatzky. 'Hand movements: A window into haptic object recognition'. In: *Cognitive Psychology* 19.3 (1987), pp. 342–368 (cit. on pp. 100, 114).

[LK93] S. J. Lederman, R. L. Klatzky. 'Extracting object properties through haptic exploration'. In: *Acta Psychologica* 84.1 (1993), pp. 29–40 (cit. on pp. 15, 61).

[LRMK10] N. Landin, J. M. Romano, W. McMahan, K. J. Kuchenbecker. 'Dimensional Reduction of High-Frequency Accelerations for Haptic Rendering'. In: *Haptics: Generating and Perceiving Tangible Sensations: Part II (Proceedings of EuroHaptics)*. Vol. 6192. Lecture Notes in Computer Science. Springer, 2010, pp. 79–86 (cit. on p. 29).

[LSG+17] H. Liu, F. Sun, D. Guo, B. Fang, Z. Peng. 'Structured output-associated dictionary learning for haptic understanding'. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47.7 (2017), pp. 1564–1574 (cit. on pp. 101, 102).

[MBKF14a] M. Madry, L. Bo, D. Kragic, D. Fox. 'ST-HMP: Unsupervised Spatio-Temporal feature learning for tactile data'. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. May 2014, pp. 2262–2269 (cit. on pp. 57–59, 68, 71).

[MBKF14b] M. Madry, L. Bo, D. Kragic, D. Fox. 'ST-HMP: Unsupervised Spatio-Temporal feature learning for tactile data'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2014, pp. 2262–2269 (cit. on pp. 102, 103).

[McC80] P. McCullagh. 'Regression Models for Ordinal Data'. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 42.2 (1980), pp. 109–142 (cit. on p. 82).

[MPC16] D. J. Meyer, M. A. Peshkin, J. E. Colgate. 'Tactile Paintbrush: A Procedural Method for Generating Spatial Haptic Texture'. In: *Proceedings of the IEEE Haptics Symposium*. Apr. 2016, pp. 259–264 (cit. on p. 22).

[MS11] J. H. McDermott, E. P. Simoncelli. 'Sound texture perception via statistics of the auditory periphery: Evidence from sound synthesis'. In: *Neuron* 71.5 (2011), pp. 926–940 (cit. on p. 47).

[MSB+14] L. R. Manfredi, H. P. Saal, K. J. Brown, M. C. Zielinski, J. F. Dammann III, V. S. Polashock, S. J. Bensmaia. 'Natural scenes in tactile texture'. In: 111 (2014), pp. 1792–1802 (cit. on p. 21).

[MT22] A. Metzger, M. Toscani. 'Unsupervised learning of haptic material properties'. In: *eLife* 11.e64876 (2022) (cit. on pp. 47, 52).

[NGW+12] S. E. Navarro, N. Gorges, H. Wörn, J. Schill, T. Asfour, R. Dillmann. 'Haptic object recognition for multi-fingered robot hands'. In: *Proceedings of the IEEE Haptics Symposium*. 2012, pp. 497–502 (cit. on pp. 101, 102).

[ONe17] T. A. O'Neill. 'An Overview of Interrater Agreement on Likert Scales for Researchers and Practitioners'. In: *Frontiers in Psychology* 8.777 (2017) (cit. on pp. 85, 86).

[ONY13] S. Okamoto, H. Nagano, Y. Yamada. 'Psychophysical Dimensions of Tactile Perception of Textures'. In: *IEEE Transactions on Haptics* 6.1 (2013), pp. 81–93 (cit. on pp. 16, 22, 66).

[PCC19] K. Priyadarshini, S. Chaudhuri, S. Chaudhuri. 'PerceptNet: Learning Perceptual Similarity of Haptic Textures in Presence of Unorderable Triplets'. In: *Proceedings of the IEEE World Haptics Conference (WHC)*. July 2019, pp. 163–168 (cit. on p. 23).

[PDVG03] D. Picard, C. Dacremont, D. Valentin, A. Giboreau. 'Perceptual dimensions of tactile textures'. In: *Acta Psychologica* 114.2 (2003), pp. 165–184 (cit. on pp. 28, 64).

[PGV77] J. Piaget, H. Gruber, J. Vonèche. 'The Essential Piaget'. In: *Educational Researcher* 8 (Dec. 1977) (cit. on p. 103).

[PRK93] Y. C. Pati, R. Rezaiifar, P. S. Krishnaprasad. 'Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition'. In: *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*. 1993, pp. 40–44 (cit. on p. 67).

[PS00] J. Portilla, E. P. Simoncelli. 'A parametric texture model based on joint statistics of complex wavelet coefficients'. In: *International Journal of Computer Vision* 40 (2000), pp. 49–70 (cit. on p. 47).

[PY10] S. J. Pan, Q. Yang. 'A Survey on Transfer Learning'. In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359 (cit. on p. 57).

[RDE16] R. Rosas-Romero, A. Díaz-Torres, G. Etcheverry. 'Forecasting of stock return prices with sparse representation of financial time series over redundant dictionaries'. In: *Expert Systems with Applications* 57 (Sept. 2016), pp. 37–48 (cit. on p. 69).

[RK19] B. A. Richardson, K. J. Kuchenbecker. 'Improving Haptic Adjective Recognition with Unsupervised Feature Learning'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2019, pp. 3804–3810 (cit. on pp. 19, 56, 102, 103).

[RK20] B. A. Richardson, K. J. Kuchenbecker. 'Learning to Predict Perceptual Distributions of Haptic Adjectives'. In: *Frontiers in Neurorobotics* 13 (2020), pp. 1–16 (cit. on pp. 19, 22, 56, 101, 102).

[RKM23] B. A. Richardson, K. J. Kuchenbecker, G. Martius. *A Sequential Group VAE for Robot Learning of Haptic Representations*. 2023. In preparation for submission to Robotics: Science and Systems (cit. on pp. 19, 100).

[RVWK22] B. A. Richardson*, Y. Vardar*, C. Wallraven, K. J. Kuchenbecker. 'Learning to Feel Textures: Predicting Perceptual Similarities from Unconstrained Finger-Surface Interactions'. In: *IEEE Transactions on Haptics* (2022). *Equal contribution. Accepted (cit. on pp. 18, 24).

[SBKS20] M. Strese, L. Brudermueller, J. Kirsch, E. Steinbach. 'Haptic Material Analysis and Classification Inspired by Human Exploratory Procedures'. In: *IEEE Transactions on Haptics* 13.2 (2020), pp. 404–424 (cit. on pp. 22, 52).

[SD14] H. Soh, Y. Demiris. 'Incrementally Learning Objects by Touch: Online Discriminative and Generative Models for Tactile-Based Recognition'. In: *IEEE Transactions on Haptics* 7.4 (2014), pp. 512–525 (cit. on p. 103).

[SF05] A. Streri, J. Féron. 'The development of haptic abilities in very young infants: From perception to cognition'. In: *Infant Behavior and Development* 28.3 (2005), pp. 290–304 (cit. on p. 15).

[SHC+15] Z. Su, K. Hausman, Y. Chebotar, A. Molchanov, G. E. Loeb, G. S. Sukhatme, S. Schaal. 'Force estimation and slip detection/classification for grip control using a biomimetic tactile sensor'. In: *IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. Nov. 2015, pp. 297–303 (cit. on p. 58).

[SHCR20] L. Skedung, K. L. Harris, E. S. Collier, M. W. Rutland. 'The finishing touches: the role of friction and roughness in haptic perception of surface coatings'. In: *Experimental Brain Research* 238.238 (2020), pp. 1511–1524 (cit. on pp. 22, 23, 27).

[SK21] G. Serhat, K. J. Kuchenbecker. 'Free and Forced Vibration Modes of the Human Fingertip.' In: *Applied Sciences* 11.12 (2021) (cit. on p. 50).

[SL10] A. Stevenson, C. A. Lindberg, eds. *New Oxford American Dictionary*. 3rd ed. Oxford University Press, 2010 (cit. on pp. 93, 94).

[SL95] M. A. Srinivasan, R. H. LaMotte. 'Tactual discrimination of softness'. In: *Journal of Neurophysiology* 73.1 (1995), pp. 88–101 (cit. on p. 95).

[SLCD16] A. J. Spiers, M. V. Liarokapis, B. Calli, A. M. Dollar. 'Single-grasp object classification and feature extraction with simple robot hands and tactile sensors'. In: *IEEE Transactions on Haptics* 9.2 (Apr. 2016), pp. 207–220 (cit. on pp. 56, 102).

[SNMU22] K. Sato, S. Nakata, T. Matsubara, K. Uehara. 'Few-Shot Anomaly Detection Using Deep Generative Models for Grouped Data'. In: *IEICE Transactions on Information and Systems* 105 (2022), pp. 436–440 (cit. on p. 104).

[SS13] J. Sinapov, A. Stoytchev. 'Grounded object individuation by a humanoid robot'. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. May 2013, pp. 4981–4988 (cit. on pp. 57, 58).

[SSIS17] M. Strese, C. Schuwerk, A. Iepure, E. Steinbach. 'Multimodal feature-based surface material classification'. In: *IEEE Transactions on Haptics* 10.2 (Apr. 2017), pp. 226–239 (cit. on p. 102).

[SSS+09] A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, W. Burgard. 'Object identification with tactile sensors using bag-of-features'. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2009, pp. 243–248 (cit. on pp. 98, 102).

[SSS15] M. Strese, C. Schuwerk, E. Steinbach. 'Surface classification using acceleration signals recorded during human freehand movement'. In: *Proceedings of the IEEE World Haptics Conference*. June 2015, pp. 214–219 (cit. on p. 57).

[STAM22] M. Seitzer, A. Tavakoli, D. Antic, G. Martius. 'On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. Apr. 2022 (cit. on p. 107).

[SWS09] J. Sinapov, M. Wiemer, A. Stoytchev. 'Interactive learning of the acoustic properties of household objects'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2009, pp. 2518–2524 (cit. on p. 114).

[THHS20] G. Tatiya, R. Hosseini, M. C. Hughes, J. Sinapov. 'A Framework for Sensorimotor Cross-Perception and Cross-Behavior Knowledge Transfer for Object Categorization'. In: *Frontiers in Robotics and AI* 7 (2020) (cit. on pp. 101, 102).

[Twi65] T. E. Twitchell. 'The automatic grasping responses of infants'. In: *Neuropsychologia* 3.3 (1965), pp. 247–259 (cit. on p. 15).

[Vil08] C. Villani. *Optimal transport: old and new*. Vol. 338. Springer Science & Business Media, 2008 (cit. on pp. 32, 33).

[VLBM08] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol. 'Extracting and Composing Robust Features with Denoising Autoencoders'. In: *Proceedings of the 25th International Conference on Machine Learning*. July 2008, pp. 1096–1103 (cit. on p. 59).

[VVPH15] F. Veiga, H. Van Hoof, J. Peters, T. Hermans. 'Stabilizing novel objects by learning to predict tactile slip'. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Sept. 2015, pp. 5065–5072 (cit. on p. 58).

[VWK19] Y. Vardar, C. Wallraven, K. J. Kuchenbecker. 'Fingertip Interaction Metrics Correlate with Visual and Haptic Perception of Real Surfaces'. In: *Proceedings of the IEEE World Haptics Conference (WHC)*. July 2019, pp. 395–400 (cit. on pp. 22, 24, 27, 46).

[WKWD06] J. Z. Wu, K. Krajnak, D. E. Welcome, R. G. Dong. 'Analysis of the dynamic strains in a fingertip exposed to vibrations: Correlation to the mechanical stimuli on mechanoreceptors'. In: *Journal of Biomechanics* 39.13 (2006), pp. 2445–2456 (cit. on p. 50).

[WLH11] M. Wiertlewski, J. Lozada, V. Hayward. 'The Spatial Spectrum of Tangential Skin Displacement Can Encode Tactual Texture'. In: *IEEE Transactions on Robotics* 27.3 (2011), pp. 461–472 (cit. on p. 52).

[WSL+13] A. I. Weber, H. P. Saal, J. D. Lieber, J. W. Cheng, L. R. Manfredi, J. F. Dammann III, S. J. Bensmaia. 'Spatial and temporal codes mediate the tactile perception of natural surfaces'. In: 110.42 (2013), pp. 17107–17112 (cit. on pp. 21, 52).

[XLF13] D. Xu, G. E. Loeb, J. A. Fishel. 'Tactile identification of objects using Bayesian exploration'. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. May 2013, pp. 3056–3061 (cit. on p. 56).

[YBCH07] T. Yoshioka, S. J. Bensmaia, J. C. Craig, S. S. Hsiao. 'Texture perception through direct and indirect touch: An analysis of perceptual space for tactile textures in two modes of exploration'. In: *Somatosensory Motor Research* 24.1-2 (2007), pp. 53–70 (cit. on pp. 22, 27, 28, 49).

[YMWA18] W. Yuan, Y. Mo, S. Wang, E. H. Adelson. 'Active Clothing Material Perception Using Tactile Sensing and Deep Learning'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4842–4849 (cit. on p. 102).

[YZO+17] W. Yuan, C. Zhu, A. Owens, M. A. Srinivasan, E. H. Adelson. 'Shape-independent hardness estimation using deep learning and a GelSight tactile sensor'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 951–958 (cit. on p. 102).

List of Figures

List of Tables

List of Algorithms