**Title**
Learning Complex Functional Manipulations by Human Demonstration and Fluent Discovery



**Author**
Edmonds, Mark J

**Publication Date**
2017

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

# Learning Complex Functional Manipulations by Human Demonstration and Fluent Discovery

A thesis submitted in partial satisfaction

of the requirements for the degree

Master of Science in Computer Science

by

Mark Joseph Edmonds

2017

ABSTRACT OF THE THESIS

Learning Complex Functional Manipulations by Human Demonstration and Fluent Discovery

by

Mark Joseph Edmonds
Master of Science in Computer Science
University of California, Los Angeles, 2017
Professor Song-Chun Zhu, Chair

Learning complex robot manipulation policies for real-world objects is extremely challenging, and often requires significant tuning within controlled environments. In this thesis, we learn an And-Or graph-based model to execute tasks with highly variable structure and multiple stages, which are typically not suitable for most policy learning approaches. The model is learned from human demonstration using a tactile glove that measures both hand pose and contact forces. The tactile glove enables observation of visually latent changes in the scene, specifically the forces imposed to unlock safety mechanisms. From these observations a stochastic grammar model is learned to represent the compositional nature of the task sequence, and the compatibility of that sequence with the observed tactile feedback. We present a method for transferring this human-specific knowledge onto a robot platform, and demonstrate that the robot can perform successful manipulations of unseen objects with similar task structure.

The thesis of Mark Joseph Edmonds is approved.

Ying Nian Wu

Demetri Terzopoulos

Song-Chun Zhu, Committee Chair

University of California, Los Angeles

2017

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

# ACKNOWLEDGMENTS

This thesis consists of two conference paper submissions. The first, "Learning Complex Functional Manipulations by Human Demonstration and Fluent Discovery" constitutes most of this thesis and was submitted to IROS 2017, currently pending review. The list of co-authors is Mark Edmonds, Feng Gao, Xu Xie, Hangxin Liu, Siyuan Qi, Yixin Zhu, Brandon Rothrock, and Song-Chun Zhu.

The second submission, entitled "A Glove-based System for Studying Hand-Object Manipulation via Pose and Force Sensing," also submitted for review to IROS 2017, focuses on the tactile glove presented in this thesis. The co-author list is Hangxin Liu, Xu Xie, Matt Millar, Mark Edmonds, Feng Gao, Yixin Zhu, Veronica Santos, Brandon Rothrock, and Song-Chun Zhu.

This work was supported by a DARPA SIMPLEX grant N66001-15-C-4035, ONR MURI grant N00014-16-1-2007, and DARPA XAI grant N66001-17-2-4029. I would also like to thank Victor Zhang of the UCLA Electrical Engineering Department for his help evaluating the tactile glove, Eric Peltola, Alireza Haji Fathaliyan, and Xiaoyu Wang of the UCLA Mechanical and Aerospace Engineering Department for their help with the Vicon motion capture system, and Ruiqi Gao of the UCLA Statistics Department for her help with experiments and evaluations.

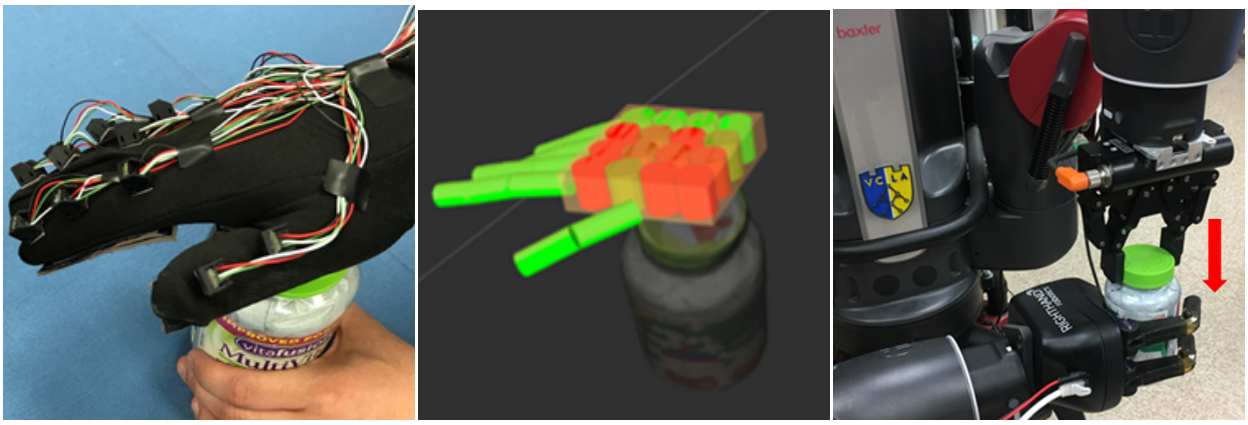
Finally, I would like to express deep gratitude toward my family for their never-ending support and encouragement during my higher education.

# CHAPTER 1

# Introduction

Manually developing policies for complex manipulation tasks is traditionally very challenging, and conventional methods are often very brittle when presented with variability in the objects or the underlying structure of the manipulation sequence. Learning from demonstration is a compelling approach to this problem, as it does not require expert knowledge of the domain dynamics. This thesis specifically focuses on using human demonstration to learn manipulation policies for objects that have similar functional properties, but exhibit very different geometries as well as internal configurations that affect how the object must be manipulated. Specifically, we choose the task of opening a collection of child-proof bottles. These bottles are available in a wide variety of shapes and sizes, and most importantly have a variety of different internal locking mechanisms. Although two bottles may appear identical, the procedure for opening them may be considerably different. The learned manipulation policy must therefore provide reasoning on how to adaptively choose the appropriate sequence of operations, as well as account for variations in geometry.

Determining the status of the manipulation task (*e.g.* if the cap is slipping because it needs to be pushed first) requires tactile feedback, which is difficult or impossible for the human to respond to when using kinesthetic teaching or tele-operation. Vision-based methods have difficulty identifying the precise pose of the hand and lack knowledge regarding the forces exerted by the demonstrator. Instead, we utilize human demonstrations using a tactile glove designed to measure both hand pose and contact forces across the surface of the hand. These demonstrations are performed naturally, and within a motion capture setup to obtain ground-truth tracking of the objects and human wrist. Our tactile glove enables the robot to observe visually latent causal changes during the demonstration, as forces exerted




(a) Data collection (b) Human modeling (c) Robot execution

Figure 1.1: Knowledge transfer sequence. (a) Collect data using tactile glove. (b) Model collected data. (c) Transfer and execute modeled data on robot platform

by the hand are difficult if not impossible to observe from vision alone.

From these demonstrations, we learn a stochastic grammar model that represents the compositional hierarchy of atomic actions. This grammar compactly captures the admissible sequence of actions for all the bottles demonstrated, as well as the compatibility of the tactile feedback for each action and between adjacent actions. The main contribution of this work is to transfer the policy model learned from human demonstration onto the robot and demonstrate successful manipulation of unseen but functionally similar objects. This policy model captures latent relationships that are imperceptible from vision alone. The key challenge is to address the correspondence problem [DH02] that relates the tactile sensing from the human experience to that of the robot. To address this, we learn a mapping function that directly relates hand pose and contact force from the human to the force-torque sensing and gripper state of the robot, such that the robot can reason about its proprioceptive measurements using the relations learned from human demonstration. These relations help determine when actions are completed successfully or have failure conditions, as well as determine the most likely choice for the subsequent action in the sequence being executed.

## 1.1 Related Work

### 1.1.1 Learning from demonstration (LfD)

LfD is a critical component to building general purpose robots. Humans quickly learn from each other and from past experiences, often only requiring a single example to learn a new task [LST15]. Teaching robots to achieve similar performance would enable robots to enter many routine human activities.

LfD can be divided into two categories: demonstration and imitation. Demonstrations are utilized through human kinesthetic teaching and teleoperation [ACV09]. Kinesthetic teaching and teleoperation both enable direct mappings between demonstrations and executions. This approach eliminates the need for an *embodiment mapping*, a function that maps states/actions in the demonstration to states/actions on the robot [ACV09]. In kinesthetic teaching, the teacher physically maneuvers the robot whereas teleoperation relies on an external controller to maneuver the robot. Controllers vary, but common methods include data gloves [CPB06, KFM14] and exoskeletons [LB04].

Imitation corresponds to a mismatch between the demonstrator's components and the robot. That is, the robot cannot directly utilize the human demonstration to maneuver; the sensors/actuators between the demonstrator and robot are different. To accommodate this mismatch, approaches seek to imitate at low-level motor functions or task-level functions. Imitating low-level motor functions attempts to mimic the demonstrator's execution using an approximation of the demonstrator's state (such as the trajectory of the demonstrator's hand). Examples of low-level imitation include maze-following [HD94], following social affordance cues from a human [SGR17], and learn to clean a whiteboard [FJ13]. Task-level imitation attempts to match the task goal of the demonstration, but does not directly consider low-level motor functions of the demonstrator. Task-level imitation can be used to learn how to pour liquid [LKG14] or assemble a structure using magnetic blocks [PJK16].

Most similar to our work, Huang *et al.* used a data glove, force torque sensors, and motion capture to learn policies to open a set of traditional bottles. However, their work

focuses largely on satisfying low-level imitation observed from the demonstrator and uses an apparatus to simplify grasping the bottle lid. Our work uses a similar experimental setup but focuses on causal and task-level knowledge transfer between the human and the robot.

Although kinesthetic teaching and teleoperation simplify knowledge transfer by eliminating the *correspondence problem* (*i.e.* learning the embodiment mapping; mapping between the demonstrator's and robot's embodiment), producing demonstrations is extremely time consuming and requires highly specialized domain knowledge, namely how to operate the robot. Imitation learning is a popular approach for applications with repetitive or simplistic motion trajectories, but becomes much more difficult with complex tasks. This work seeks to abstract much of the low-level motor control that is typical of most imitation learning work and instead places emphasis on learning task-level functions of the demonstration. This abstraction uses a collection of motion primitives that operate in task space, and allows us to focus on learning high-level policies that are semantic in nature. In contrast with previous work, our method focuses on physical measurements not observable from vision.

Other approaches to endow robots with functional knowledge include policy search, reinforcement learning, and optimization. In [LWA15], a robot learns manipulation tasks as a guided policy search problem. The approach is model- and example-free using linear-Gaussian controllers to guide robotic movement. There has also been a recent surge in reinforcement learning-based robot manipulation tasks. Such interest includes end-to-end training using deep convolutional neural networks (CNNs) [LFD16]. Mordatch *et al.* introduce an optimization-based method to unify contact activity (*i.e.* whether or not a given end-effector should be in contact during this motion) and movement [MTP12, MLT15]. While these methods are effective, we argue they are not realistic for robots expected to interact in collaborative environments. Learning from demonstration is an essential skill in environments where robots are expected to interact and learn from humans on a regular basis.

### 1.1.2 Tactile Glove

To capture visually hidden states, we use a tactile glove to record human pose and force applied at each proximal and distal phalanx and a 4-by-4 grid of sensors to detect forces exerted by the palm. Pose sensor gloves use a wide array of sensors. Most data gloves use IMUs to track finger pose [TKM13, KSR14, SLS15], though curvature sensors are also used [KMS11, KSE08]. However, the flex sensor typically decreases a user's natural motion and dexterity. Filtering is commonly used to improve pose accuracy [SLS15] and Kalman filtering is used to increase the robustness of state estimation [KSR14, LS13, KAS15]. Another designs include: an exoskeleton coupled a DC motor and a time-of-flight sensor to obtain hand pose without using a glove directly [CHC16], a Vicon motion capture sensor to track wrist, index finger, and thumb angles [BBA16].

To read force, some data gloves use a FlexiForce [GSL15, BBA16] sensor, though there has been an emerging interest in using Velostat, a piezoresistive conductive film, as a force sensor [PMT16, LKY15, JLK11]. Other approaches include a liquid-metal embedded elastomer sensor to measure forces across the palm [HMW14]. Our tactile glove takes a similar approach to [KSR14] for finger pose and utilizes Velostat sensors to measure force.

### 1.1.3 And-Or Graph

Our learning framework relies on a spatial, temporal, and causal And-Or (STC-AOG) graph, which has successfully modeled human action and attention from videos [XSX16, WZZ13, SPY11, FZ13b]. Other work uses syntactic grammars to successfully synthesize robot plans to solve the Tower of Hanoi [LSK13]. To our knowledge, this is the first work to learn a STC-AOG with a tactile glove as the demonstration data source.

This thesis focuses on the detection and utilization of visually latent states in the human demonstration. These states are crucial for our tasks, and the demonstrator must convey the necessity of unlocking the safety lock on the child-proof bottles. The tactile glove provides perceptual causality that is hidden from a visual observer. Perceptual causality attempts to infer causal relationships solely from observation [ST00, FZ13a].

We use our tactile glove and a Vicon motion capture system to observe the poses and forces exerted by the human demonstrator and infer an embodiment mapping between the human and robot to transfer spatial, temporal, and causal knowledge to the robot. Our primary contribution is the ability for the robot to utilize complex task policies by mapping its proprioceptive measurements into the human perceptual space to utilize models learned from human demonstration.

# CHAPTER 2

# Tactile Glove Design

Tactile gloves have been proposed for a wide range of applications. We use a tactile glove to capture the pose and applied forces during human demonstrations. The glove reconstructs the pose of each finger using 15 IMUs, and detects force using Velostat force sensors. An IMU is place on each phalanx of the hand and forward kinematics is used to compute the pose of each phalanx. Hand pose reconstruction relies on the 15 IMUs, each of which report an orientation. The IMUs measure the attitude of each phalanx and the palm. From these orientations, a kinematic model of the hand is used to reconstruct hand pose. The kinematic model is fixed; this introduces slight error given different phalanx lengths of each user.

To detect forces applied during demonstrations, a network of Velostat sensors used. Velostat is a conductive material which changes resistance under pressure. A 4x4 grid of Velostat sensors are placed on the palm, and each phalanx has a sensor on the distal and proximal link. Each Velostat sensor must be calibrated with known forces, as each sensor has a unique contact area and therefore a unique resistive response to pressure. This glove provides 71 degrees of freedom including all pose and force measurements. The glove provides an accurate model of state of the human hand and the forces exerted by each phalanx. The overall glove system architecture is shown in Figure 2.1.

## 2.1 Hardware Implementation

In this section, the details and capabilities of the tactile glove are outlined. The hardware system is reproducible and low-cost. The tactile glove opens the opportunity for a wildly utilized research glove for human demonstrations. Due to the glove's modularized design, it

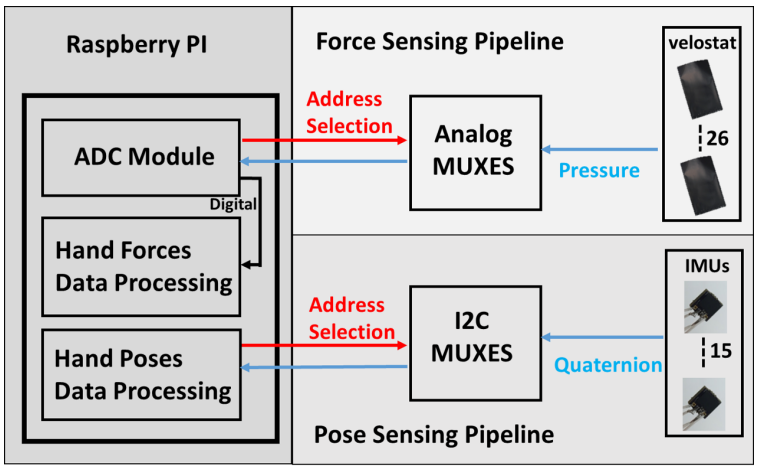
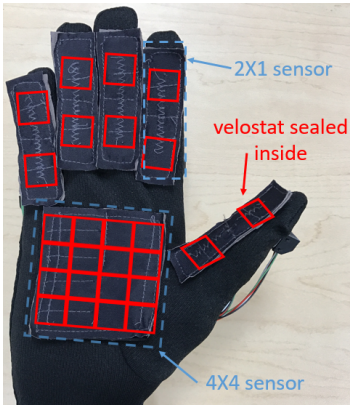
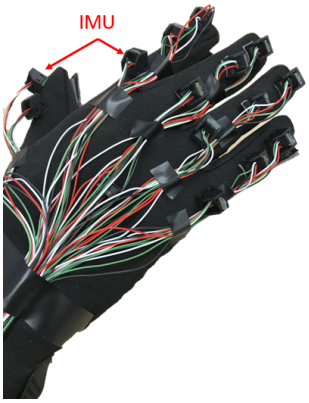



Figure 2.1: Overall system schematic




(a) Palmar view



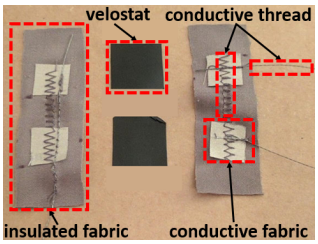

(b) Dorsal view

Figure 2.2: The tactile glove consisting 6 integrated Velostat force sensor with 26 sensors on the palmar aspects of the hand and 15 IMUs on the dorsum of the hand.

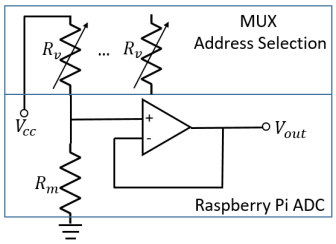
can be adapted for new research areas, such as virtual reality. A prototype of the glove is shown in Figure 2.2.

### 2.1.1 Pose Sensing Pipeline

The hand pose is reconstructed using 15 9DoF IMUs. An IMU is placed on each phalanx of the hand, with three on each finger and two on the thumb. The final IMU is mounted on the palm of the hand. Each IMU is a Bosch BNO055, and each contains a 12-bit triaxial




(a) Velostat sensor construction




(b) Velostat sensor circuit

Figure 2.3: Velostat force sensor construction and circuit layout

accelerometer, a 16-bit triaxial gyroscope, and a triaxial geomagnetometer. The sensor data is fused using a proprietary, black-box Bosch algorithm on a 32-bit microprocessor. This yields a global-frame quaternion for each IMU.

The BNO055 is mounted on a custom $250 \times 250$ mil$^2$ ($6.35 \times 6.35$ mm$^2$) breakout PCB. Each IMU has a footprint is $5 \times 4.5$ mm$^2$, making it small enough to enable natural motion. The IMUs are connected to a pair of I$^2$C buses in a star configuration. Each I$^2$C bus is multiplexed using a TCA9548A I$^2$C multiplexer. Each multiplexer is connected to a single Raspberry Pi 2 Model B through two on-board I$^2$C buses. The Raspberry Pi acts as the master controller of the system. This design is largely based on the work by Kortier *et al.* [KSR14]. In this work, a similar design yields a measurement error of approximately 1 degree. The glove components are connected using a highly flexible, silicone-coated 29-gauge wire. The IMUs are glued into small 3D-printed housings, and each housing is then sewn into the glove's Lycra fabric at each corresponding phalanx.

### 2.1.2 Force Sensing Pipeline

A network of Velostat sensors are used to detect the forces exerted by the human demonstrator on objects. Figure 2.3a shows the multi-layer structure of the Velostat sensor. The sensor consists of a contact patch, where the Velostat material is sandwiched between two layers of conductive thread and conductive fabric. The outer layer of the sensor consists of a layer insulated fabric. Lead wires connect the conductive thread and provide a connection

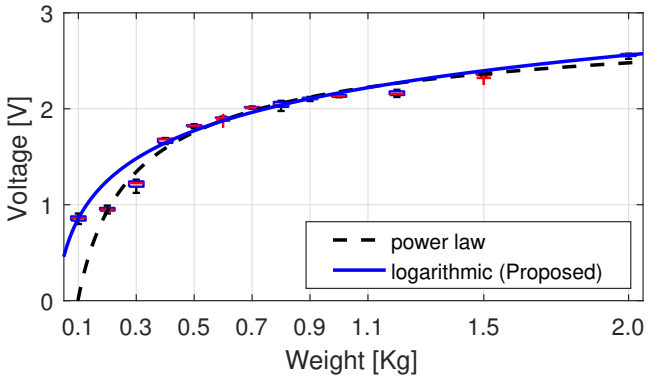



Figure 2.4: Force-voltage relation of one constructed Velostat sensing unit. A logarithmic law fit performs slightly better than a power law fit.

to the rest of the system.

The Velostat sensors on the palm are connected to a pair of 74HC4051 analog multiplexers, and the sensors on the fingers are connected to a single CD74HC4067 analog multiplexer. The multiplexers are controlled using the Raspberry Pi 2's GPIO. The Raspberry Pi 2 reads the values from a SPI-enabled ADS1256 ADC at approximately 40 samples per second.

In order to calibrate the force sensor, a series of experiments are conducted using known weights, in a setup similar to [LS16]. The weights are applied to each $2\times2$ cm$^2$ patch of Velostat. The weights range from 0.1kg to 1.0kg, with 0.1kg increments in between. Additional weights with values of 1.2kg, 1.5kg, and 2.0kg are used. The calibration circuit is identical to the circuit in Figure 2.3b, except that the Velostat sensor of interest is connected. Following a similar methodology as [LS16], a voltage divider is used to tune the sensing range of each sensor. The force-voltage relation follows a power law with $F=-1.067V^{-0.4798}+3.244$ with $R^2=0.9704$, where $F$ is the applied force and $V$ is the output voltage. However, a force-voltage relation based on a logarithmic law yields a better relation, with $F=0.569\log(44.98V)$ with $R^2=0.9902$. The force-voltage (weight-voltage) relationships are shown in Figure 2.4.

## 2.2 Software Implementation

The tactile glove software implementation consists of three major components: 1) hand pose calculation, 2) force vector calculation, 3) visualization. The software system is built on top the ROS environment to enable widespread adoption within the robotics community [QCG09].

The tactile has 71 DoF, while the human hand contains only 20 DoF. Twenty six of the tactile glove DoF arise from the force sensors, which are not considered for the human hand. Each metacarpophalangeal (MCP) joint has 2 DoF, with 1 for the proximal interphalangeal (PIP) joints, and 1 DoF for distal interphalangeal (DIP) joints. Due to these constraints, each finger can be modeled as a 4 DoF kinematic chain where the palm is the base frame and the distal phalanx is the end-effector frame. To simplify modeling, the thumb is represented by a 3 DoF kinematic chain comprised solely of its interphalangeal and carpometacarpal joints.

The hand pose reconstruction is computed using forward kinematics, with each finger representing a kinematic chain. Figure 2.5 shows the kinematic chain along the index finger. The palm is the root frame of the chain, and the proximal, middle, and distal phalanges are frame 2 through 4. The lengths $l_1$, $l_2$, $l_3$ denote the length of the proximal, middle, and distal phalanges. We denote the abduction/adduction and flexion/extension angles of the MCP joint by $\beta$ and $\theta_1$, respectively. The flexion/extension angles of the PIP and DIP joints are denoted $\theta_2$ and $\theta_3$. The offset between the palm's center and the MCP joint is denoted $d_x$ and $d_y$ for the $x$ and $y$ directions, respectively.

Using notation, we compute the standard Denavit-Hartenberg (D-H) parameters for each frame. Table 2.1 shows the D-H parameters. Also using this notation, we can compute a general homogeneous transformation matrix $\boldsymbol{T}$ from the previous frame, $i-1$ to the current frame, $i$:

$$
{}^{i-1}_{i}T = \begin{bmatrix} cos(\theta_i) & -sin(\theta_i) & 0 & a_{i-1} \\ sin(\theta_i)cos(\alpha_{i-1}) & cos(\theta_i)cos(\alpha_{i-1}) & -sin(\alpha_{i-1}) & -sin(\alpha_{i-1})d_i \\ sin(\theta_i)sin(\alpha_{i-1}) & cos(\theta_i)sin(\alpha_{i-1}) & cos(\alpha_{i-1}) & cos(\alpha_{i-1})d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.1}
$$

The pose of each phalanx relative to the palm can be computed by concatenating each homogeneous transformation. We impose motion constraints on finger joints as in [LWH00] with the joint limits in Equation 2.2. These limits restrict the possible motion to what is achievable by the human hand and eliminate unnatural motion due to sensor noise or error.

$$
\begin{aligned} 0^\circ \leq \theta_1 \leq 90^\circ \\ 0^\circ \leq \theta_2 \leq 110^\circ \\ 0^\circ \leq \theta_3 \leq 90^\circ \\ -15^\circ \leq \beta \leq 15^\circ. \end{aligned} \tag{2.2}
$$

IMUs have no global reference measurement and are subsequently subject to drift over time. To account for this error, a canonical pose is used to calibrate the glove. The hand is held in a flat, finger-extended pose during calibration. The glove's reference frame is defined by an $x$-$y$ plane with the $x$-axis extending along the user's middle finger and the $y$-axis extending to the left of the user's hand. The orientation of the palm IMU is measured with

Table 2.1: General standard Denavit-Hartenberg parameters of a finger

| Link ID | $\alpha_{i-1}$ | $a_{i-1}$ | $\theta_i$ | $d_i$ |
|---|---|---|---|---|
| 1 | 0 | $d_x$ | $\beta$ | $d_y$ |
| 2 | $\pi/2$ | $l_1$ | $\theta_1$ | 0 |
| 3 | 0 | $l_2$ | $\theta_2$ | 0 |
| 4 | 0 | $l_3$ | $\theta_3$ | 0 |

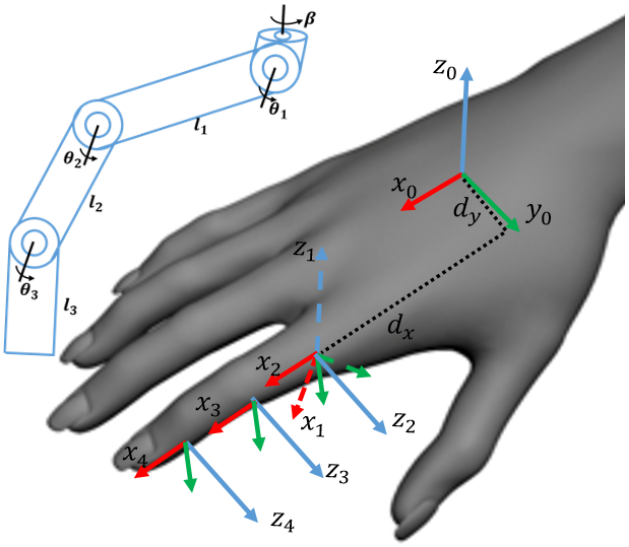



Figure 2.5: Frame attachment and the kinematic chain of the index finger.

respect to this coordinate system, and the remaining IMUs are measured with respect to the palm IMU.

From the hand pose reconstruction and Velostat force readings, the location, direction, and magnitude of the forces applied by the human demonstrator can be computed. Each force vector is comprised of the force magnitude reported by the Velostat sensor and the direction is defined by the pose of the IMU. The pose of the phalanx reported by forward kinematics translates the force vector in the frame of the glove.

The reconstructed pose is visualized in ROS using the Unified Robot Description Format (URDF). URDF is an XML-based robot model definition format that describes a robot's linkages and joint constraints. The hand is modelled similarly to a robot; the URDF defines the length of each phalanx and imposes the joint limits in Equation 2.2.

# CHAPTER 3

# Task Representation

To learn from demonstration, we represent the task using a STC-AOG. Our method relies on the following knowledge: (1) spatial knowledge to encode the poses of objects and manipulators, (2) temporal knowledge to encode action sequencing, and (3) causal knowledge to encode how changes in the spatial and temporal domains achieve the task.

## 3.1 And-Or Graph

In this section, we will introduce our representation for modeling demonstration data onto the robot platform. An AOG is a generative grammar, allowing the robot to encode compositional variability in the demonstrated task sequences. The And-Or Graph is represented by a 4-tuple:

$$\mathcal{G}_{\text{And-Or}} = \langle N = U \cup V, T, \mathcal{A} \rangle \tag{3.1}$$

In Equation 3.1, $N$ represents the set of non-terminal nodes in the graph. An And-node $u \in U$ represents a decomposition of the graph into sub-graphs. The sub-graphs are connected through a set of relations $r_1, \ldots, r_k \in \mathcal{R}$. An Or-node $v \in V$ acts as a switch among multiple alternate sub-configurations. The terminal nodes $T = \{t_1, \ldots, t_m\}$ is a set of sub-components representing the lowest level of resolution in the graph. For opening bottles, the terminal nodes correspond to *atomic actions* (Section 3.2) executed during the task. $A$ represents a set of attributes derived from the terminal nodes. In this thesis, $A$ is a set of *fluent functions* (Section 3.3) that operate on terminal nodes.

The collected data contains spatial, temporal and causal knowledge. Spatial knowledge captures the physical configuration of the robot environment, temporal knowledge encodes the sequence of atomic action to complete the task, causal knowledge expresses the status change of an object caused by an action. All three components are simultaneously modeled in a STC-AOG.

## 3.2 Atomic Actions

Both the human and robot actions are modeled using atomic actions [PSY13]. Atomic actions are equivalent to movement primitives in traditional robotics motion control literature [SPN05, PDP13]. An atomic action represents the finest resolution at which to consider planning problems. We denote the human dictionary of atomic actions to be $\Delta_h$, and each $a_h^i \in \Delta_h$ represents a human atomic action.

We endow the robot with a dictionary of atomic actions, denoted $\Delta_r$, and manually map a correspondence between human and robot action labels. This correspondence reduces the complexity of the robot planning problem; without it, the robot would search within its action and its action sequence space at the same time. This thesis addresses searching in the action sequence space alone. Examples of endowed robot actions are summarized below in Table 3.1

Table 3.1: Example robot atomic actions

| Name | Description |
|---|---|
| Approach | Moves end-effector towards a target |
| Grasp | Closes end-effector |
| Move | Moves end-effector to a position |
| Push | Applies force through translation of the arm |
| Twist | Rotates the wrist of the robot by a given degree |


(a) Consecutive atomic actions (b) Consecutive atomic actions under causal assumption

Figure 3.1: Atomic actions and their fluents

## 3.3 Fluents

We model the causal knowledge as changes in the spatial domain or temporal domain using fluent changes between the pre- and post-condition of each action. Each fluent function maps the system configuration, $s_t$, to a real value, $f(s_t) \mapsto \mathbb{R}$. A fluent change represents a transition between two system configurations, $\Delta f(s_i, s_j) = f(s_j) - f(s_i)$. We denote the action at time $t$ as $a_t$. We denote the system configuration of the pre-condition as $s_{t-1}$ and the post-condition as $s_t$. Each action can be characterized by the changes it imposes across all fluents, denoted $\Delta f = \{\Delta f_i(s_{t-1}, s_t), \quad i = 1...n\}$.

Using this representation, our learning task is to discriminate the fluent changes of each human action such that we can compute $p(a_t | f(s_{t-1}), f(s_t))$, the probability of action $a_t$, given the corresponding pre- and post-condition. We also seek to learn an embodiment mapping to translate between robot and human configurations in the fluent space. We assume the human demonstrator/robot is the only causal agent in the environment and the *inertial action* assumption [MT97]. These two assumptions imply a causal chain between the agent's previous action and the next action; i.e. the post-condition of the previous action is the pre-condition of the current action, shown in Figure 3.1.

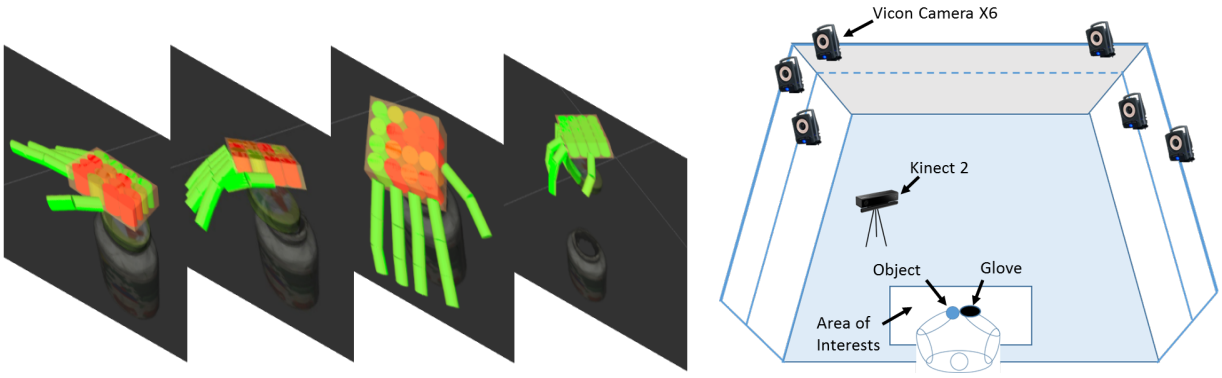
# CHAPTER 4

# Data Collection

A human demonstrator performed opening various types of bottles, shown in Figure 4.1. Some of the bottles contain child-safety locking mechanisms which require a procedure beyond simply twisting to unscrew the cap. Most child-safety locks require a particular force to be exerted on a particular part of the bottle. These forces are difficult to infer from visual observation alone. We collected human data on bottles 2, 3, and 5. The remaining bottles were reserved for testing.

To estimate object states, we use a Vicon motion capture system and attach fiducials on each bottle and its lid. We also attach fiducials to the tactile glove to capture wrist pose in world space. Vicon requires a minimum of 3 fiducials to be visible at all times. Due the closeness of the bottle, the lid, and the glove during demonstration, the fiducials are mounted


Figure 4.1: Bottles used in experiments. (1) bottle with *push-and-twist* safety mechanism. (2) bottle with *pinch-and-twist* safety mechanism. (3) bottle with *push-and-twist* safety mechanism. (4) bottle with *push-and-twist* safety mechanism. (5) bottle with no safety mechanism.




(a) Collected human data

(b) Room configuration

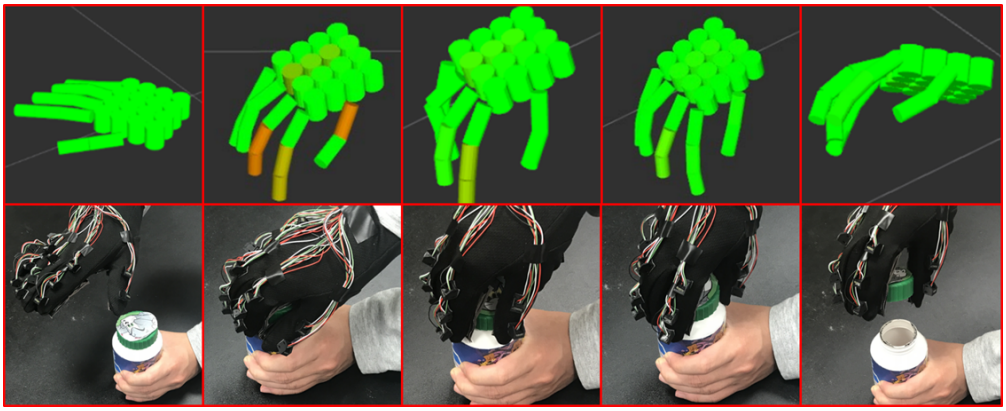
Figure 4.2: Data capture configuration. (a) modeled data from human demonstration. (b) data collection configuration consisting of demonstrator wearing the tactile glove, a motion capture system, and a RGB camera

onto 3D-printed extenders. These extenders must be position into unique configurations for proper localization within the motion capture system. The experimental setup is shown in Figure 4.2.

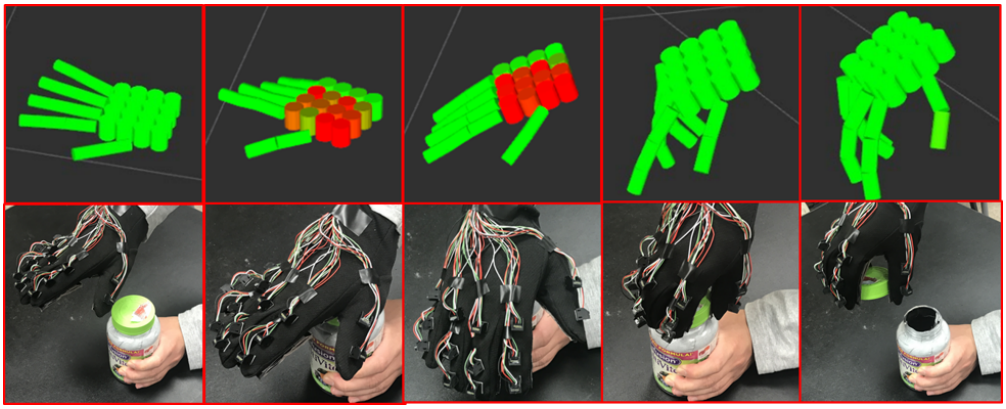
We collect approximately 10 trials for each grasping strategy for each bottle. Bottle 2 only has one grasping strategy: *pinch-and-twist*. Bottle 3 has two strategies: *push-and-twist* using the palm, or *push-and-twist* using fingers. Bottle 5 has three strategies since it lacks a safety mechanism: *twist*, *push-and-twist*, or *pinch-and-twist*. While there may be multiple ways to open each bottle, not all methods are considered equally. For instance, Bottle 5 has no safety mechanism, so while *push-and-twist* and *pinch-and-twist* may succeed in opening bottle 5, there is no reason to execute anything other than *twist*. We use this distinction during evaluation of our method. Examples of collected data are shown in Figure 4.3.

Each demonstration is manually labeled. We manually label the data to mitigate the correspondence problem between a human action and a robot action. The timestamps of the labeling provide the transition boundaries between actions; i.e. the post-condition of the labeled action and the pre-condition of the next action. This labeling procedure ensures there is no external causal change between actions, as assumed in Section 3.2.

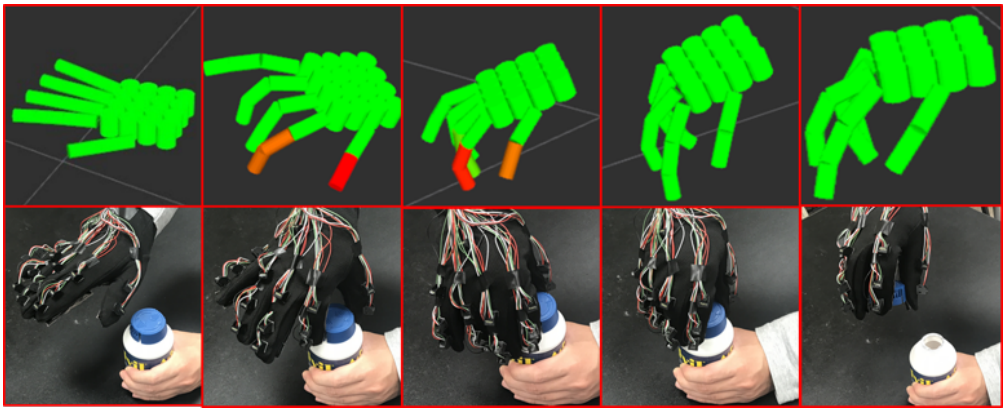
Actions such as *grasp-and-push* or *grasp-and-pinch* are labelled as a single action due to



(a) Bottle 5, regular twist to open



(b) Bottle 3. pressing the lid to open



(c) Bottle 2, pinching lock to open

Figure 4.3: Action sequences and visualization of opening three types of bottles

the small time delay between actions; the bottle is grasped simultaneously as it is pushed or pinched. This simplified labelling, however, it does make the corresponding robot atomic actions more complex.

# CHAPTER 5

# Learning from Demonstration

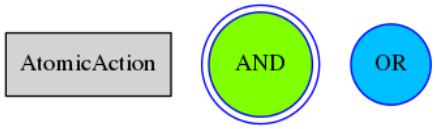
From the human data, we first learn a temporal And-Or graph (T-AOG) that encodes a grammar of valid sequences of atomic actions. We induce the T-AOG from the labeled sequences using the method presented in [TPZ13]. This method induces a stochastic context-free grammar with probabilistic Or-nodes in the graph. We denote the training set as $X$, where $x_i \in X$ represents a sequence of atomic actions, $x_i = (a_1, a_2, \ldots, a_m)$, from the demonstrator. The grammar is induced by maximizing the posterior probability of the grammar, $G$, given the data, $X$:

$$p(G|X) \propto p(G)p(X|G) = \frac{1}{Z} e^{-\alpha||G||} \prod_{x_i \in X} p(x_i|G) \tag{5.1}$$

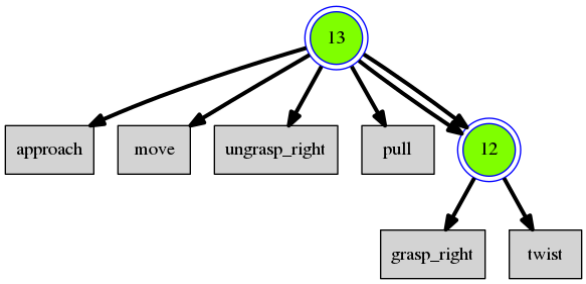
The resulting T-AOG can be used to sample valid sequences of atomic actions used by the human to achieve the task. Because the robot has a different embodiment and manipulators, the grammar is not guaranteed to produce sequences that succeed when executed by the robot. The induced grammars as additional human demonstrations are added are shown in Figure 5.1.

## 5.1 Compatibility Model

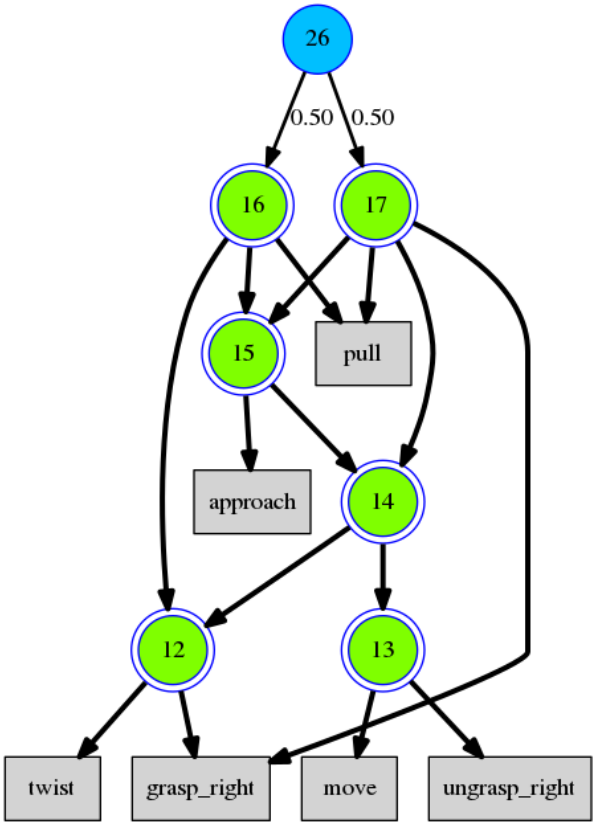
Compatibility between robot states and actions is learned discriminatively from human demonstration and robot executions using neural network architectures. During robot execution, a compatibility check consists of two stages: 1) an embodiment mapping between the robot and a low-dimensional human embedding and 2) a transition model to predict



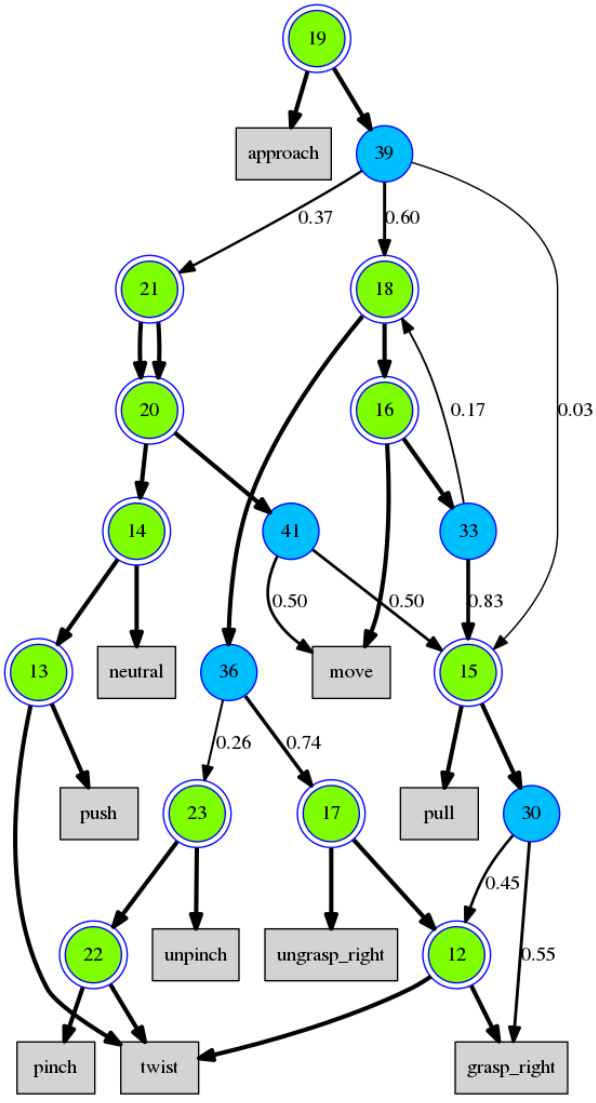

(a) Node legend




(b) T-AOG from one example




(c) T-AOG from two examples




(d) T-AOG from 65 examples

Figure 5.1: T-AOG generated from human data

the current and next action from a low-dimensional human embedding using Equation 5.5.

From the human demonstration, an auto-encoder is trained to embed the space of observed hand geometries and force distributions into a low dimensional subspace. Changes of the hand configuration within this subspace are treated as fluent changes and are used to infer the current and predicted action sequence with observed proprioceptive feedback. From this subspace, we train the transition model to predict both the current and next action from tactile observations of the pre- and post-condition of the current action.

The contact force and pose measurements from the tactile glove are reoriented to the reference frame of the wrist, and concatenated into a feature vector with 159 dimensions. An encoder-decoder architecture, illustrated in Figure 5.2, is used to learn a 6-dimensional embedding and reconstruct the full feature vector from this embedding under a criteria that minimizes the squared residuals between the original feature and the reconstruction.

The classification networks take their input from the embedding layer of the auto-encoder, and learns a multi-class classifier to predict one of the 10 atomic actions. A softmax layer is used to interpret this as a probability distribution using Equation 5.2.

$$\sigma(\mathbf{x})_i = \frac{e^{x_i}}{\sum_{k=1}^{K} e^{x_k}} \quad i = 1, \ldots, K \tag{5.2}$$

The network is trained by minimizing the normalized cross-entropy. All internal layers are linear matrix operators, and use sigmoids for their non-linearities. All elements of the network are trained simultaneously using tied weights between the auto-encoders for the pre- and post-conditions.

The probability for a sequence of actions and corresponding fluent measurements is

$$p(f_{1\ldots t-1}) = \sum_{a_{1\ldots t}} p(f_{1\ldots t-1}|a_{1\ldots t})p(a_{1\ldots t}). \tag{5.3}$$

where $p(a_{1..t})$ is the parse probability from the grammar. Because an action has to be selected at every step, we only need to choose the most likely next action given the previous observations, from which we use two time steps to capture the pre- and post-condition of

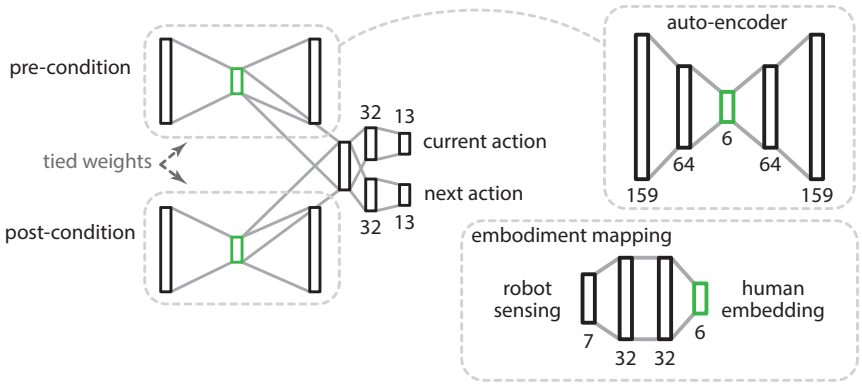



Figure 5.2: Neural network architecture for classifying current and future actions using a low-dimensional embedding of human tactile feedback.

the last executed action:

$$a_t^* = \arg\max_{a_t} p(f_{t-1}, f_{t-2} | a_{2\ldots t}) p(a_{1\ldots t}) \tag{5.4}$$

$$\propto \arg\max_{a_t} p(a_t | f_{t-1}, f_{t-2}) p(a_{1\ldots t}). \tag{5.5}$$

where $p(a_t | f_{t-1}, f_{t-2})$ is the probability learned by the network to classify the next action.

We approach the embodiment problem by learning a mapping between the proprioceptive sensing on the robot and the tactile measurements from the human demonstration. Specifically, we supervise robot executions sampled from the T-AOG using the robot's dictionary $\Delta_r$. The robot is supervised during executions to ensure only successful robot states are mapped to successful demonstrator states. We denote $s_h$ to represent the human state of the demonstration and $s_r$ to represent the robot.

Our embodiment mapping seeks a function $s_h = f(s_r)$, which maps a robot state in the robot's configuration space to the low-dimensional human perceptual space. We train a neural network to learn this function from a small number of robot examples (approximately 15 examples). The embodiment mapping network is illustrated in Figure 5.2. The robot utilizes this mapping to verify that the executed action is consistent with the tactile measurements experienced by the human.

Using the transition model and embodiment mapping, the compatibility check projects the robot's state into the low-dimensional human embedding and uses the transition model

to predict the current action and next action the robot will execute. If the actions match, the robot can infer this action execution was a success, otherwise, a failure.

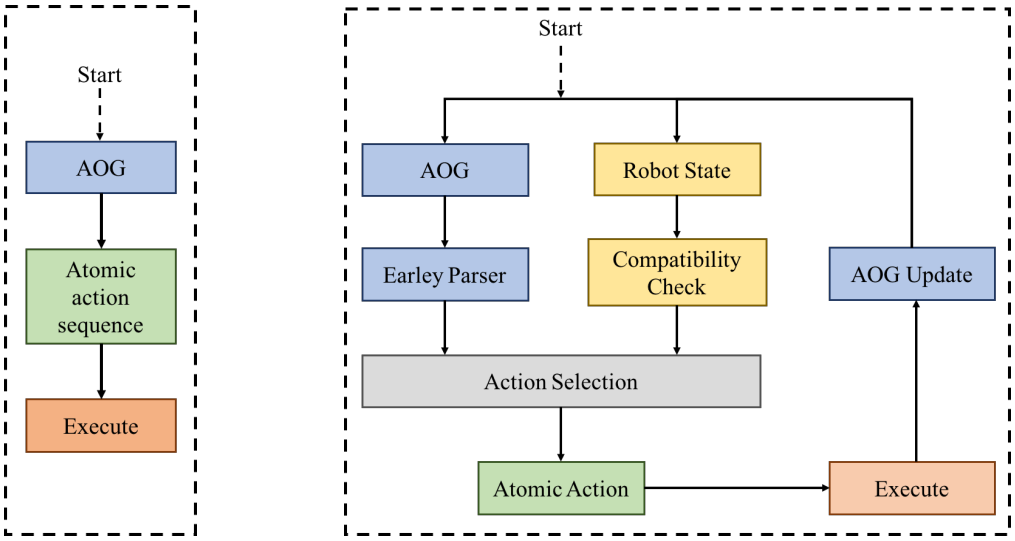
# CHAPTER 6

# Implementation

Our robot platform consists of a dual-armed 7-DoF Baxter robot from Rethink Robotics mounted on a DataSpeed Mobility Base. The robot is equipped with a ReFlex TackkTile gripper on the right wrist, and a Robotiq S85 parallel gripper on the left. In addition, we use Simtrack [PK15] for object pose estimation and tracking using a Kinect V2. The entire system runs on ROS [QCG09], and arm motion planning is computed using MoveIt! [SC13]. For object grasping, we implement a geometry based grasp planner to compute and generate grasping poses from CAD models of the objects.

In our implementation, the S-AOG stores physical knowledge of all objects in the scene. The S-AOG is updated using SimTrack and the robot state. The human T-AOG contains human actions along the time axis, including geometry transformation and physics variation of human hand pose which were retrieved from tactile glove. The robot T-AOG was constructed with the atomic actions segmented from human demonstration data. The C-AOG corresponds to the fluent changes the transition model predicts when an action is executed. Together, these form a unified STC-AOG which captures the spatial, temporal, and causal components of the task.

Figure 6.1 shows the system architectures. Figure 6.1a shows the initial system used for baseline for evaluation. This architecture is also used to supervise the robot's execution while learning the embodiment mapping. Once the mapping is learned, we use the system in Figure 6.1b to sequentially choose atomic action using the Earley parser and compatibility model. The robot STC-AOG is updated upon every successful bottle opening.




(a) Sampled sequence (b) Sampled sequence with online learning

Figure 6.1: System architectures for evaluation: (a) Each sentence is sampled directly from T-AOG and executed sequentially. (b) Each action is sampled one-by-one using the compatibility model.

## 6.1 Sampling action sequences using T-AOG

The T-AOG encodes the temporal sequence of actions and can be directly mapped between the robot and human. This describes the order in which actions were executed, but does not capture spatial or causal relationships. The And-node is generated sequentially, while the Or-node is chosen with the probability that proportional to the learned Or-node probability.

After generating the entire action sequence from the grammar, the robot executes the entire sequence without utilizing the compatibility model between two consecutive actions. Due to the fact the T-AOG is a context-free grammar, the sampled sequence may adhere to the grammar but not enforce long-range constraints of a task, leading to task failure or misuse of actions. For example, the robot may select to *pinch* bottle 3, which requires a *push*. Due to these failures, the training executions for the embodiment mapping must be supervised; we only wish to encode robot states that correlate to successful demonstrations.

## 6.2 Sampling action sequences with online learning

With online learning, the temporal sequence of actions are not generated by sampling from AOG; each action is generated one by one using a graph parser. We adopt the Earley parser [Ear70]. Once an action is generated from the Earley parser, a compatibility check is performed to predict the current and the next action. The action selected is executed, and if the action is successfully executed, the robot generates the next action.

If an action sequence was successfully executed by Baxter, the action sequence will be added to update the learned AOG. This loop enables the robot to integrate the knowledge learned from human and explored by itself. Note that while the compatibility model is using the robot state to predict the next action, it does not enforce context-sensitivity within the the action. Thus, it is still possible for the robot to select to *pinch* bottle 3, which requires a *push*. However, because the system is learning with each successful execution, it becomes increasingly less likely the robot will select an inappropriate action due to the updated values of probabalistic Or-nodes.

# CHAPTER 7

# Experiments

## 7.1 Experiment Setup

Five bottles were used in the evaluation as shown in Figure 4.1. Bottle 2, 3, and 5 were used in human data collection, while the remaining of the bottles were reserved for testing.

We classify robot executions into five different categories: 1) success, 2) success but using extra unnecessary actions, 3) success but using at least one wrong action(s), 4) failure due to using the wrong action, and 5) failure due to improper execution. While categories 2 and 3 are similar, they have a subtle difference for task completion. In category 2, the unnecessary actions do not contribute to task completion. For example, the robot may *twist* the lid an additional, unnecessary rotation. In category 3, the robot selects the wrong action, but the action still contributes to task completion. For example, the robot may *push* on a bottle with no safety lock. However, the *push* action's definition includes grasping the lid, so the action still contributes to the overall task completion.

An execution is deemed successful if the robot opens the bottle; otherwise, the execution is a failure. If the robot opens the bottle before finishing the sampled execution, we consider the action sequence that it performed is correct and discard remain actions. Each bottle was tested approximately 20 times.

## 7.2 Results

We conduct experiments opening bottles with varying or absent child lock safety mechanisms. Table 7.1 and 7.2 show the comparisons between baseline method and proposed

Table 7.1: Baseline: executions sampling sequences from T-AOG (percentage)

| **Bottle** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| Success | 0.00 | 0.05 | 0.05 | 0.00 | 0.36 |
| Success (extra actions) | 0.11 | 0.00 | 0.05 | 0.25 | 0.18 |
| Success (wrong actions) | 0.00 | 0.11 | 0.00 | 0.19 | 0.36 |
| Failure (action) | 0.00 | 0.16 | 0.15 | 0.15 | 0.05 |
| Failure (execution) | 0.89 | 0.68 | 0.075 | 0.44 | 0.05 |

Table 7.2: Proposed method: executions with online learning (percentage)

| **Bottle** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| Success | 0.50 | 1.00 | 0.4 | 0.60 | 1.00 |
| Success (extra actions) | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Success (wrong actions) | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Failure (action) | 0.50 | 0.00 | 0.00 | 0.40 | 0.00 |
| Failure (execution) | 0.00 | 0.00 | 0.6 | 0.00 | 0.00 |

method. The baseline method only consider the action sequence from the T-AOG, whereas the proposed method includes the compatibility model after executing each action. The results demonstrate that the proposed method could utilize the compatibility model to correct the action sequence generated directly from T-AOG, improving the successful rate and preventing the extra unnecessary actions or wrong actions.

Figure 7.1: Successful robot execution sequence

# CHAPTER 8

# Conclusion

In this work, we present a method to learn visually latent fluent changes from human demonstration and successfully complete the same task during robot execution. The tactile glove provides a data collection method to capture visually hidden causal changes in the scene. Using this latent encoding of the scene, we learn a model to predict the actions of the human demonstrator. The human demonstrator's action sequences are used to induce a temporal And-Or graph. The T-AOG is used to supervise successful executions of opening a bottle.

The robot states of successful executions are mapped to successful demonstrations from the human demonstrator using a low-dimensional embedding of the human tactile feedback. The mapping function provides an embodiment mapping between the human and the robot using a relatively small number of supervised robot executions. The robot utilizes this mapping to infer whether or not the current action executed successfully. This inference is then used to update the original T-AOG, enabling online learning and refinement of the model learned from human demonstration.

This system presents a novel method of capturing visually hidden states of a task and transfers them to the robot using a tactile glove, supervised learning, and online learning. The robot successfully demonstrates its ability to utilize and improve the model learned from human demonstrations.

## 8.1 Future Work

This work paves the way for additional work regarding visually latent states and corresponding embodiment mappings. We would like to investigate methods to make the system less

supervised, such as clustering the human demonstrations instead of labeling actions. The method should be expanded to utilize more actions and investigate different tasks. As this work focuses on a precision-based task, specifically opening a bottle, this method should also be examined against power-based task, such as cracking a nut. Power-based tasks involve dynamics, and our tactile glove may be useful in capturing visually latent forces exerted by the hand. This information may lead to more robust models over traditional vision-based dynamics reasoning methods. Another direction includes the examining the generality of the learned STC-AOG. In [ZZZ15], the functional equivalent of tools was examined. The framework presented here could potentially be used to attempt functionally equivalent tasks based on reasoning similar to [ZZZ15]. In this way, the robot could demonstrate understanding the which dynamics of the task need to be replicated and which can be safely ignored.

# REFERENCES


[ACV09] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. "A survey of robot learning from demonstration." *Robotics and Autonomous Systems*, **57**(5):469–483, 2009.

[BBA16] Edoardo Battaglia, Matteo Bianchi, Alessandro Altobelli, Giorgio Grioli, Manuel G Catalano, Alessandro Serio, Marco Santello, and Antonio Bicchi. "ThimbleSense: a fingertip-wearable tactile sensor for grasp analysis." *IEEE Transactions on Haptics*, **9**(1):121–133, 2016.

[CHC16] Inrak Choi, Elliot W Hawkes, David L Christensen, Christopher J Ploch, and Sean Follmer. "Wolverine: A wearable haptic interface for grasping in virtual reality." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 986–993. IEEE, 2016.

[CPB06] Christina L Campbell, Richard Alan Peters, Robert E Bodenheimer, William J Bluethmann, Eric Huber, and Robert O Ambrose. "Superpositioning of behaviors learned through teleoperation." *IEEE Transactions on Robotics*, **22**(1):79–91, 2006.

[DH02] Y Derimis and G Hayes. "Imitations as a dual-route process featuring predictive and learning components: a biologically plausible computational model." *Imitation in animals and artifacts*, pp. 327–361, 2002.

[Ear70] Jay Earley. "An efficient context-free parsing algorithm." *Communications of the ACM*, **13**(2):94–102, 1970.

[FJ13] Mohsen Falahi and Masoumeh Jannatifar. "Using orthogonal basis functions and template matching to learn whiteboard cleaning task by imitation." In *Computer and Knowledge Engineering (ICCKE), 2013 3rd International eConference on*, pp. 289–294. IEEE, 2013.

[FZ13a] Amy Sue Fire and Song-Chun Zhu. "Learning Perceptual Causality from Video." In *AAAI Workshop: Learning Rich Representations from Low-Level Sensors*, 2013.

[FZ13b] Amy Sue Fire and Song-Chun Zhu. "Using Causal Induction in Humans to Learn and Infer Causality from Video." In *CogSci*, 2013.

[GSL15] Ye Gu, Weihua Sheng, Meiqin Liu, and Yongsheng Ou. "Fine manipulative action recognition through sensor fusion." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 886–891. IEEE, 2015.

[HD94] Gillian M Hayes and John Demiris. *A robot controller using learning by imitation*. University of Edinburgh, Department of Artificial Intelligence, 1994.

[HMW14] Frank L Hammond, Yiğit Mengüç, and Robert J Wood. "Toward a modular soft sensor-embedded glove for human hand motion and tactile pressure measurement." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 4000–4007. IEEE, 2014.

[JLK11] Eunseok Jeong, Jaehong Lee, and DaeEun Kim. "Finger-gesture recognition glove using velostat (ICCAS 2011)." In *International Conference on Control, Automation and Systems (ICCAS)*, pp. 206–210. IEEE, 2011.

[KAS15] Henk G Kortier, Jacob Antonsson, H Martin Schepers, Fredrik Gustafsson, and Peter H Veltink. "Hand pose estimation by fusion of inertial and magnetic sensing aided by a permanent magnet." *Transactions on Neural Systems and Rehabilitation Engineering*, **23**(5):796–806, 2015.

[KFM14] Kamil Kukliński, Kerstin Fischer, Ilka Marhenke, Franziska Kirstein, V Maria, Norbert Krüger, Thiusius Rajeeth Savarimuthu, et al. "Teleoperation for learning by demonstration: Data glove versus object manipulation for intuitive robot control." In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2014 6th International Congress on*, pp. 346–351. IEEE, 2014.

[KMS11] Rebecca K Kramer, Carmel Majidi, Ranjana Sahai, and Robert J Wood. "Soft curvature sensors for joint angle proprioception." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 1919–1926. IEEE, 2011.

[KSE08] Nidal S Kamel, Shohel Sayeed, and Grant A Ellis. "Glove-based approach to online signature verification." *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, **30**(6):1109–1113, 2008.

[KSR14] Henk G Kortier, Victor I Sluiter, Daniel Roetenberg, and Peter H Veltink. "Assessment of hand kinematics using inertial and magnetic sensors." *Journal of Neuroengineering and Rehabilitation*, **11**(1):70, 2014.

[LB04] Jeff Lieberman and Cynthia Breazeal. "Improvements on action parsing and action interpolation for learning through demonstration." In *International Conference on Humanoid Robots*, volume 1, pp. 342–365. IEEE, 2004.

[LFD16] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. "End-to-end training of deep visuomotor policies." *Journal of Machine Learning Research*, **17**(39):1–40, 2016.

[LKG14] Joshua D Langsfeld, Krishnanand N Kaipa, Rodolphe J Gentili, James A Reggia, and Satyandra K Gupta. "Incorporating failure-to-success transitions in imitation learning for a dynamic pouring task." In *Workshop on Compliant Manipulation: Challenges and Control, Chicago, IL*, 2014.

[LKY15] JH Low, PM Khin, and CH Yeow. "A pressure-redistributing insole using soft sensors and actuators." In *International Conference on Robotics and Automation (ICRA)*, pp. 2926–2930. IEEE, 2015.

[LS13] Gabriele Ligorio and Angelo Maria Sabatini. "Extended Kalman filter-based methods for pose estimation using visual, inertial and magnetic sensors: comparative analysis and performance evaluation." *Sensors*, **13**(2):1919–1941, 2013.

[LS16] Byung Woo Lee and Hangsik Shin. "Feasibility study of sitting posture monitoring based on piezoresistive conductive film-based flexible force sensor." *Sensors*, **16**(1):15–16, 2016.

[LSK13] Kyuhwa Lee, Yanyu Su, Tae-Kyun Kim, and Yiannis Demiris. "A syntactic approach to robot imitation learning using probabilistic activity grammars." *Robotics and Autonomous Systems*, **61**(12):1323–1334, 2013.

[LST15] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. "Human-level concept learning through probabilistic program induction." *Science*, **350**(6266):1332–1338, 2015.

[LWA15] Sergey Levine, Nolan Wagener, and Pieter Abbeel. "Learning contact-rich manipulation skills with guided policy search." In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 156–163. IEEE, 2015.

[LWH00] John Lin, Ying Wu, and Thomas S Huang. "Modeling the constraints of human hand motion." In *Workshop on Human Motion*, pp. 121–126. IEEE, 2000.

[MLT15] Igor Mordatch, Kendall Lowrey, and Emanuel Todorov. "Ensemble-CIO: Full-body dynamic motion planning that transfers to physical humanoids." In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 5307–5314. IEEE, 2015.

[MT97] Norman McCain, Hudson Turner, et al. "Causal theories of action and change." In *AAAI/IAAI*, pp. 460–465, 1997.

[MTP12] Igor Mordatch, Emanuel Todorov, and Zoran Popović. "Discovery of complex behaviors through contact-invariant optimization." *ACM Transactions on Graphics (TOG)*, **31**(4):43, 2012.

[PDP13] Alexandros Paraschos, Christian Daniel, Jan R Peters, and Gerhard Neumann. "Probabilistic movement primitives." In *Advances in neural information processing systems*, pp. 2616–2624, 2013.

[PJK16] Chris Paxton, Felix Jonathan, Marin Kobilarov, and Gregory D Hager. "Do what I want, not what I did: Imitation of skills by planning sequences of actions." In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pp. 3778–3785. IEEE, 2016.

[PK15] Karl Pauwels and Danica Kragic. "Simtrack: A simulation-based framework for scalable real-time object pose detection and tracking." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 1300–1307. IEEE, 2015.

[PMT16] Ganna Pugach, Artem Melnyk, Olga Tolochko, Alexandre Pitti, and Philippe Gaussier. "Touch-based admittance control of a robotic arm using neural learning of an artificial skin." In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 3374–3380. IEEE, 2016.

[PSY13] Mingtao Pei, Zhangzhang Si, Benjamin Z Yao, and Song-Chun Zhu. "Learning and parsing video events with goal and intent prediction." *Computer Vision and Image Understanding*, **117**(10):1369–1383, 2013.

[QCG09] Morgan Quigley, Ken Conley, Brian P. Gerkey, Josh Faust, Tully Foote, Jermey Leibs, Rob Wheeler, and Andrew Y. Ng. "ROS: an open-source Robot Operating System." In *ICRA Workshop on Open Source Software*, 2009.

[SC13] Ioan A Sucan and Sachin Chitta. "Moveit!" *Online at http://moveit. ros. org*, 2013.

[SGR17] Tianmin Shu, Xiaofeng Gao, MS Ryoo, and Song-Chun Zhu. "Learning Social Affordance Grammar from Videos: Transferring Human Interactions to Human-Robot Interactions." In *ICRA 2017*, 2017.

[SLS15] Gaspare Santaera, Emanuele Luberto, Alessandro Serio, Marco Gabiccini, and Antonio Bicchi. "Low-cost, fast and accurate reconstruction of robotic and human postures via IMU measurements." In *International Conference on Robotics and Automation (ICRA)*, pp. 2728–2735. IEEE, 2015.

[SPN05] Stefan Schaal, Jan Peters, Jun Nakanishi, and Auke Ijspeert. "Learning movement primitives." In *Robotics Research. The Eleventh International Symposium*, pp. 561–572. Springer, 2005.

[SPY11] Zhangzhang Si, Mingtao Pei, Benjamin Yao, and Song-Chun Zhu. "Unsupervised learning of event and-or grammar and semantics from video." In *International Conference on Computer Vision (ICCV)*, pp. 41–48. IEEE, 2011.

[ST00] Brian J Scholl and Patrice D Tremoulet. "Perceptual causality and animacy." *Trends in cognitive sciences*, **4**(8):299–309, 2000.

[TKM13] Thomas Taylor, Seungoh Ko, Carlos Mastrangelo, and Stacy J Morris Bamberg. "Forward kinematics using IMU on-body sensor network for mobile analysis of human kinematics." In *Engineering in Medicine and Biology Society (EMBC)*, pp. 1230–1233. IEEE, 2013.

[TPZ13] Kewei Tu, Maria Pavlovskaia, and Song-Chun Zhu. "Unsupervised structure learning of stochastic and-or grammars." In *Advances in neural information processing systems*, pp. 1322–1330, 2013.

[WZZ13] Ping Wei, Yibiao Zhao, Nanning Zheng, and Song-Chun Zhu. "Modeling 4d human-object interactions for event and object recognition." In *International Conference on Computer Vision (ICCV)*, pp. 3272–3279, 2013.

[XSX16] Caiming Xiong, Nishant Shukla, Wenlong Xiong, and Song-Chun Zhu. "Robot learning with a spatial, temporal, and causal and-or graph." In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2144–2151. IEEE, 2016.

[ZZZ15] Yixin Zhu, Yibiao Zhao, and Song-Chun Zhu. "Understanding tools: Task-oriented object modeling, learning and recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2855–2864, 2015.