Probabilistic Representation of Perceived Tool Affordances for Robot Tool Use

Raghvendra Jain

Adviser: Assoc. Prof. Tetsunari Inamura

April 28, 2015

Abstract

The tool-use model to perform a task depends on: properties and configurations of multiple objects and surfaces that interact with each other, perceptual and motor skills of the robot, its capabilities to predict the consequences of its motor and cognitive behaviors while handling the uncertainties and plan the sequence of behaviors to realize the desired effect. However, due to variety of tasks and several variations within them, it is not feasible for a robot to be pre-programmed with a tool use model. The goal of this research is to enable autonomous learning of tool use model to perform a task. Three sub-goals are presented in this thesis which require developing functions that enable a robot to 1) select a suitable tool among a set of available external objects to achieve the desired effects 2) perform bi-directional inferences to infer the function of tool, action (and its parameters), predict effects, plan etc; and 3) evaluate the likelihood of the success of its inferences and prediction to reformulate its plan and update the model.

The three problems in realization of the corresponding sub-goals are: 1) determining a tool representation generalizable to different objects is difficult because robot does not have the notion of relevancy between the perceptual features of available objects and the required function of the tool 2) one-way learning between action and effects does not enable bi-directional inference and it is difficult to deal with uncertainties of action-perception when learning is deterministic 3) determining a quantitative measure of its own capabilities is difficult because in deterministic tool-use model the success is expressed in binary terms.

This thesis presents an integrated approach termed as *tool affordances* to solve these problems via robot learning of bi-directional probabilistic relationship between action, *functional features* of the tool and effects of the manipulation of target object. The functional features represent the features that remain distinctive and invariant across several tools that can be used for same functionality. They can be used to transfer the learned tool-use model to unseen tools. To acquire causal and diagnostic reasoning while dealing with the uncertainties in the domain knowledge, perceptual and motor skills of the robot, learning and inference process, the probabilistic semantics of Bayesian network(BN) is used to model the tool affordances. The modeling of affordances as a probabilistic function also enables robot to get a quantitative measure of its performance for different situations, thus allowing the robot to determine when a corrective measures is required, particularly for novel situations e.g. unseen tools, action, environment, effect etc. A robot can then make probabilistic queries to the human user based on its internal state, incorporate his feedback and learn the tool use model in an online, incremental and interactive manner. Results show that probabilistic representation of tool-use model enables autonomous learning of tool-use model.

Keywords: Tool Manipulation, Probabilistic Graphical Models, Bayesian Network, Affordances

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Tool use is fundamental activity in our daily lives which we perform to extend our reach, amplify the physical strength, to transfer objects and liquids etc. For example, we use a stick to retrieve an object kept far from our reachable workspace, hammer to insert a nail into the wood, knife to cut the fruit and spoon to pour the oil while cooking etc. Thus a domestic or household robot who is supposed to help us in human environments should be able to solve tool using tasks with human competence. An intelligent robot is expected to determine what change is required in the world (e.g. setting up a table after the dinner is over); and in order to realize that required change it should be able to reason about the suitable action and tool. And since human environments are quite complex, dynamic and un-predictable, the capability to solve a tool-use task is considered as the hallmark of intelligent behavior (Amant et al.[6]).

1.1 Definitions of Tool Use

There has been several studies of tool use in animal cognition research (Parker et al.[7], Seed et al.[8]) with the objective of understanding their behavioral and cognitive capabilities (e.g. Hunt[9] studied manufacturing and usage of hook-tools in crows, Inoue et al.[10] studied stone tool use by wild chimpanzees etc.), however a universally accepted definition of tool use is still lacking. Thus, it makes sense that I provide my own definition of tool use along with the conceptual foundations, in the context of this thesis. But before that, lets look at the existing definitions.

Lawick-Goodall[11] focused on abstract properties of the behavior of tool-using agent such as functionality and goals. He described tool use as:

> *"the use of an external object as a functional extension of mouth or beak, hand or claw, in the attainment of an immediate goal"*

However this definition does not specify the required alterations in the environment that can be considered as goals for a tool-use behavior. Alcock[12] is more precise in this regard:

> *"Tool-using involves the manipulation of an inanimate object, not internally manufactured, with the effect of improving the animals efficiency in altering the form or position of some separate object."*

However, in the literature for animal cognition and tool use, a more general definition provided by Benjamin Beck[13] is now widely accepted, which states:

> *"[Tool use is] the external employment of an unattached environmental object to alter more efficiently the form, position, or condition of an- other object, another organism, or the user itself when the user holds or carries the tool during or just prior to use and is responsible for the proper and effective orientation of the tool."*

The original definition by Beck has gone through several refinements since then and many alternative definitions that fit different contexts (e.g tool-user, environment, actions, forms of engagement with the tool) has been proposed (Chevalier-Skolnikoff[14], Pretson[15], Lestel and Grundmann[16], Matsuzawa[17], Baber[2], Holmes and

Spence[18]). Please refer Bentley-Condit et al.[19] for current definitions and an updated comprehensive catalog of animal tool use. However, the original definition covers the examples of tool-use cases that I intend to address in this thesis and hence I shall adapt it in this work with slight modifications that concerns with the semantics of tools and their usage.

In this research, I am concerned with both semantics of the tools (i.e. the features that are the relevant to the functionality of the tools) and effective the employment of the tool to generate desired impact on the target. In my view, tool use is a goal directed behavior and a robot should have an understanding of the implicit goal of the task. That is to say, that robot must be responsible for its tool-using behavior (knowledge of causality) and hence accidental or incidental achievement of goals are not considered the tool-use cases.

I define tool use as:

> *Learning to utilize the functionally relevant features of the tool and generate the suitable spatial relations between tool, target, surfaces and agent's body with respect to three orthogonal properties as (a) specificity i.e. simple or precise (with respect to position, orientation or location), (b) temporal order of control i.e. static or dynamic (with respect to time) (c) temporal order of production i.e. sequential vs concurrent; to bring the required alteration in the form, position or condition of the target.*

1.2 Requirement of a Tool Use Model

To achieve the desired effects of a task, an autonomous tool-using robot should be able to perform dynamic mechanical interaction with the target object via a suitable tool. A tool-use model typically consists of sequence of several motor and cognitive behaviors i.e. (i) determination of low-level effect which is required to satisfy the task requirements (ii) selection of suitable tool based on some features relevant to the functionality of the task, (iii) selection of suitable target object (if not already given) that can be manipulated to satisfy the task requirements, (iv) determination of suitable position and orientation in which a tool should be placed relative to the target and (v) determination and generation of suitable action (and its parameters) to use the tool.

The tool use model should be robust to the contextual demands e.g. different effects, tools, environment and the robot's capabilities. For instance, to realize the low-level effect of bringing a remotely placed object (e.g. a toy car) closer, a robot should be able to reason about which of the available external objects can serve as suitable tool (e.g. it should be able to "hook" the object) and how it should be placed relative to the target object. To achieve the desired effect, it should also be able to determine and generate the suitable action e.g. pull, tap etc. (and its parameters e.g force on the tool, angle of tool movement on 2D surface etc). The action here is defined as the movement of tool that changes one or more features of the target e.g. position, orientation, shape etc. and; the effect is defined as the quantitative and qualitative change in some feature of target measured during or after the manipulation.

The model of tool-use (as considered in this thesis) depends on the requirements of the task(and its variations), environment, capabilities, knowledge and prior experiences of the tool-using robot. Other dependencies such as environmental constraints like surfaces, presence of obstacles, task constraints like time and accuracy, morphological constraints like limitation of agent's capabilities etc. are not beyond the scope of this thesis.

The development of tool-using robots designed for very specific tasks e.g. to work in space (Lovchik et al.[20], Bluethmann et al[21]) and industrial work (Whitney et al.[22], Hara et al.[23], Takeuchi et al.[24]), medical applications (Dario et al.[25], Guthart et al.[26], Tierney et al.[27]) suggests that with careful programming and background knowledge, it is possible to create tool using behaviors into an artificial agent.

However, even for a relatively simple task of pulling a distant object closer, there are different possibilities e.g., change in the initial position of box, size, shape, material etc. of the target object and/or the tool, agent's capabilities, surface frictions etc; the changes which a programmer had not anticipated in advance. Thus in real world situations, often an agent does not know the all the knowledge about the target object, tools, actions etc; which it is required to reason about the missing information based on prior experiences of performing the same task. Often, when a new tool-using task appears, a tool-using agent may not have any prior experience and is required to learn the usage of tool. Thus, the problem space of tool-use is quite large and it is a near-impossible task for the designers to embed all the required knowledge representations and intelligent behaviors within the robot.

1.3 The goal of my research

The **goal of my research** is to develop a function for robot learning of tool-use model. For, the sake of structuring and organizing my work, the goal of my research is sub-divided into following three sub-goals:

1. to develop a function that enables a robot to select objects available in its environment, to be used as the tool for performing some specific task.

2. to develop a function that enables the robot to perform causal and diagnostic reasoning about the unknown components of tool use model, while handing the uncertainties of action-perception, background knowledge and the environment. Using this function, a robot can determine suitable tools and actions to realize the desired effects and predict the probable effects that could be generated using its actions and available tools in an environment. This functions also enables robot to handle various uncertainties, redundancy and irrelevancy of the data observed by the robot during target manipulation. Thus, this function

is required for a robot to plan its motor and cognitive behaviors such that desired effects can be achieved.

3. to develop a function that enables robot to assess the plausibility of success of its tool-use plan beforehand, during or after the manipulation and reformulate it accordingly such that desired effects can be achieved.

1.4 The problems to realize the goal

The following are the three problems in related works (discussed in detail in Chapter 2) that correspond to the three sub-goals of this research:

1. The problem to realize sub-goal 1 is that robot can not determine the *relevance* of objects as a *suitable* tool because the causal probabilistic dependency between the perceptual features of objects and the desired effects of the task is not established.

2. The problem to realize sub-goal 2 is that one-way learning of tool use model i.e. role of actions and tools in obtaining the effects, does not enable inverse estimation within the tool-use model; and the deterministic mode of learning does not enable handing of uncertainties, redundancy and irrelevancy of the action-perception, background knowledge and the environment.

3. The problem to realize sub-goal 3 is that the inference mechanisms based on deterministic model of learning do not provide the quantitative measure of the *plausibility* of obtained inferences because the success and failure are judged in binary terms (i.e. with a qualitative measure). Thus, since the robot can not ascertain the confidence value of the robot in a particular situation in quantitative terms, it can not reformulate a suitable plan accordingly.

1.5 Contributions of this thesis

I have solved the three problems mentioned in this chapter by proposing three corresponding novel approaches (discussed in detail in Chapter 3). Thus, my contributions are:

1. I propose a novel approach of learning the perceptual features of the tool causally relevant to realize the desired effects of the task. These features are termed as *functional features* of the tool. This approach enables a robot to select available objects that share those functional features to be used as tools. It also enables predicting the effects of different tools on the basis of their functional features.

2. I propose an approach of bayesian learning of tool-use model termed as *tool affordances* to enable robot perform causal and diagnostic reasoning. The function of tool affordances is modeled using the probabilistic graphical model of **Bayesian Networks** (BN) . The probabilistic semantics of BN enables a robot to handle various uncertainties in its action-perception, learning, inference, environment etc.

3. I propose a novel approach to determine the quantitative measure of the *plausibility* of inferences made in different situations. The gap between the desired outcome of the task with the probable outcome predicted using the inferred value as an input is calculated. The gap value is used to compare their match, the smaller the gap the better is the match and hence more is the plausibility of the result of inference. Thus, based on the different values of gap measure a robot can reformulated its plan e.g. it may request human feedback or further explore the environment.

Outline of the Thesis

The contributions, limitations and the problems of related works in context of realizing the goal of this thesis are discussed in Chapter 2. My proposed approach is discussed in detail in Chapter 3. The proposed method, experiments and results are discussed in Chapter 4. The conclusion and future works are discussed in Chapter 5.

Chapter 2

Related Work

In this chapter, the previous work related to the manipulation of target object with and without the tool-use is reviewed. I start by examining the literature on single target object manipulation by the robot and then discuss its manipulation via a secondary object (i.e. the tool). I will also discuss about the tool representations and computational models for representing sensori-motor experiences of tool use.

2.1 The manipulation of target object without tool use

An early attempt to learn manipulation strategies by Christiansen et al.[28] allowed a robot to acquire the object manipulation model through exploration, practice and observation. The robot starts with no initial model of the consequences of its actions and learns a non-deterministic model which is incrementally refined to achieve its goals. To learn this model, the manipulation task consists of moving target objects to goal locations by tilting the tray on which they are placed. The presented system is shown to represent and successfully reason about non-deterministic effects of the actions. It also shows the benefit of using the experimental strategies for collecting the training data such that a robot's learning rate can be increased.

Zrimec[29] , Lynch[30], Yoshikawa and Kurisu[31] proposed to plan object manipulation by pushing it. The latter proposed an approach to learn the friction parameters using experimental pushes and resultant observation of the target displacement. The friction parameters and mechanical model is estimated using vision. Yoshikawa and Kurisu[31] additionally deployed a force sensor on the pusher and pushed the object via multiple contacts. The obtained mechanical analysis of the target manipulation can be used to determine the traversability of the target in specific directions when pushed with multiple pushing contacts. They also studied how target manipulation can be explored to recognize objects.

A novel supervised on-line method for the pushing manipulation based on vision is proposed by Salganicoff et al.[32]. The robot learns to manipulate objects into new locations on a plane by learning a predictive model. This prediction model is formed by observing the rotation of the object when it is pushed into the plane through a single point contact. The advantage of the method is that the prediction model being on-line in nature, keeps improving with the increase in number of target manipulations. The most informative and appropriate action is selected using *stochastic action selection* method to correct the deviations from the desired object path.

To enable autonomous target manipulation outside the controller environments, Salganicoff et al.[33] proposed a learning method that enables robot to adapt its perceptual and motor system according to the task variations. To deal with the higher costs of generating the data for learning, *active learning* algorithms based on a version of ID3 decision trees are proposed to select appropriate directions for object grasping from the visual information. For learning, the shapes of target objects is approximated as superquadric, thus modeling different objects in a uniform manner. Results show that robot rapidly learns to pick up new objects. The assumption of superquadricity of shapes also enable to generalize its learning to other similar shapes for a simulated grasping task when size of objects is larger.

The above works considered precision as the main requirement for the target manipulation. But for the situations when a robot must perform a wide variety of manipulation tasks e.g. in a kitchen, space, prosthetic etc; versatility is also the requirement. But for such cases, a robotic hand with higher degrees of freedom is needed which in turn makes it difficult to be programmed or even perform learning in a short period of time. To deal with this, Fuentes and Nelson[34] presented a method for learning dexterous manipulation of objects using multi-sensory information with multifingered robot hands having 16-degrees of freedom. The approach is to learn a few basic manipulation primitives using a genetic algorithm for a few prototypical objects and store those primitives in a database. These manipulation primitives translate or rotate a given object in a set of orthogonal directions. With the elements of the database and learned primitives, an associate memory is formed that enables determining suitable parameters for manipulating new objects e.g. by scaling and adding or subtracting the primitives. The learning algorithm is robust to the noise present in the sensors and effectors while performing a complex task in real world situations; also primitives can be learned within a short duration of few minutes.

Fitzpatrick et al.[35] addressed the problem of learning about the effects and consequences of self-generated actions. To determine how a robot learns to move an object in a particular fashion (e.g. whether object rolls or slides) and direction (towards the robot and away from it). Simple poking and prodding actions are employed to learn the resultant effects on novel objects of unknown shape and properties. Using the obtained displacement of target object and action parameters (type of action i.e. pull or push, initial hand position etc.), a mapping is learned to predict the required actions for goal directed behaviors of target object (i.e. different target displacements).

Since many objects e.g. scissors have articulations, Katz et al.[36] proposed a relational approach of learning the kinematic structure of articulated objects through manipulation. The articulated objects are represented as set of links connected by rotational or prismatic joints. The goal of the robot is to learn the structure of such objects and apply the learned policy in discovering the kinematic structure of related objects. Robot uses relational reinforcement learning to discover a new link or joint in the object in which a reward is given to it on each such discovery. The discovery is the based on observing the outcome of pushing action via one of the link of articulated objects. Robot learns a policy, which discovers the kinematic structure quickly and generalizes it to realize the goal of the their research.

2.2 The manipulation of target object via tool use

Bogoni[37] conducted the first study in robot learning of tool use. His view is that the representation for a tool must include not only its intrinsic (material) properties but also its functional (how it is used) properties. In his study, the robot uses a variety of tools made up of different shapes to perform piercing and chopping tasks. The target of learning was their suitability to perform these aforementioned tasks. The experiments involved robot manipulating target objects made from different tools materials (styrofoam, sponge, pine and balsa wood) using the tools.

With this propositional approach, the features of the tool are represented as a set of attribute-value pairs. Such a tool representation can be used to determine the suitability of different available objects to be used as tools for the similar tasks. For example, by checking the availability of same shape related features in the available objects, a robot can select them to manipulate the target object and achieve the similar effect. Thus, this approach is significant in addressing sub-goal 1. However, this approach is limited for the tools which can clearly represented as a set of attribute-values. Also, spatial relations between tool, target object, agent's body and the environment can not be suitably expressed modeled using this approach.

Stoytchev[38,39]'s work on learning of affordances (based on the concept of Affordances, originally proposed by Gibson) is most cited work in robotic tool-use. To understand his work, its benefits and drawbacks, it is imperative to first understand what Gibson meant by *Affordances*. James Gibson[40], perceptual psychologist who coined the concept of affordance defined it as "*perceptual invariants that are directly perceived by an organism*" and enable it to perform tasks. In his seminal book *Ecological Approach to Visual Perception* (page 127), Gibson[41] wrote:

> *The affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill. The verb to afford is found in the dictionary, but the noun affordance is not. I have made it up. I mean by it something that refers to both the environment and the animal in a way that no existing term does. It implies the complementarity of the animal and the environment.*

Gibson[41] also wrote on page 129:

> *An important fact about the affordances of the environment is that they are in a sense objective, real, and physical, unlike values and meanings, which are often supposed to be subjective, phenomenal, and mental. But actually, an affordance is neither an objective property nor a subjective property; or it is both if you like. An affordance cuts across the dichotomy of subjective-objective and helps us to understand its inadequacy. It is equally a fact of the environment and a fact of behavior. It is both physical and psychical, yet neither. An affordance points both ways, to the environment and to the observer.*

According to him, there are three fundamental properties of an affordance:

1. An affordance exists relative to the action capabilities of a particular actor.
2. The existence of an affordance is independent of the actors ability to perceive it.
3. An affordance does not change as the needs and goals of the actor change.

Stoytchev[38,39] proposed to learn tool affordances, solely grounded in behavioral repertoire of the robot. The robot observes and investigates the outcome of performing user-defined primitive behaviors (pushing, pulling, or sideways arm movements) whilst using different tools to the target object (an orange puck). The experimental condition is shown in Figure 2.1 in which different colored tools (stick, L-shape, T-shape, L-hook, T-hook) are used to manipulate the target object. Using different combinations of tools and behaviors the target object is displaced and its displacement is stored in a look-up table. This table is later used to solve simple manipulation tasks such as the task of *object movement* into the brown goal zone on the table shown in the Figure 2.1.

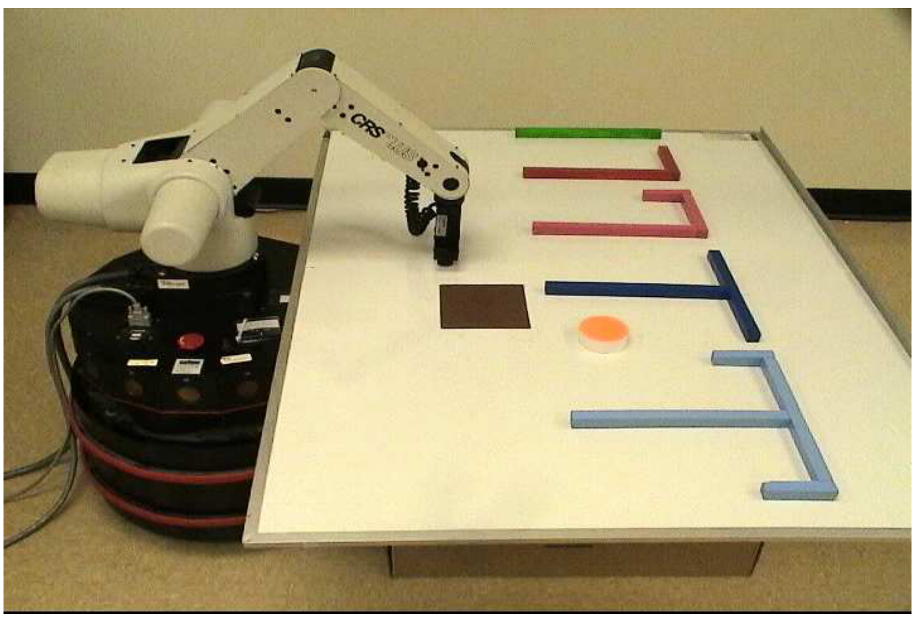


Figure 2.1: Experimental apparatus used in Stoychev[3]. The goal of the robot is push the orange puck on to the brown square using the color-coded tools and user-defined primitive behaviors (pushing, pulling, or sideways arm movements). The geometric model of the tool is not known to the robot.

Robot learns tool affordances by associating probabilities with the success of particular tool/behavior combinations which are used to *adapt* when tool was modified e.g. broken or changed. For example, one of the branches of a T-shaped stick was intentionally broken to show that the agent was able to learn that the tool affordances it was using was no longer reliable. The robot was also then able to adapt its tool behavior by choosing an alternative affordances so that it could still move the puck into the goal location.

The Gibsonian Affordances has certain limitations. As Gibson view's affordances as the action possibilities available to an individual in a particular environment i.e. these possibilities are considered relative to the capabilities of that individual agent. They are also considered independent of agent's prior experience, knowledge, culture and ability to perceive. Thus, Gibson's affordances lack the power of prediction; are invariant to the needs and goals of the the agent; and have binary existence.

Stoychev's work carries some of the limitations of Gibsonian affordances. For example, for a new tool, the affordances has to be learnt from scratch even if its similar to the previously learned tools. Since, the tool representation is done using color of the tool which is casually irrelevant feature for the outcome of target manipulation, sub-goal 1 remains unsolved. The learning of affordances enable robot to to estimate suitable action and its parameters to realize some given desired effects with a given tool. The outcome of exploratory behaviors made with a tool are stored in a affordance representation table. Stoytchev[3] showed that such affordance representation can be used as a *predictive* model of the results of target manipulation. However, the "one-way" learning does enable robot to perform both causal and diagnostic reasoning, thus sub-goal 2 remains unsolved.

Gibsonian notion suggests that affordances are independent of agent's prior experience. Thus, robot can not perform the estimation when action parameters and tool representation are not used previously (i.e. only "memory" based look-up is used to predict the outcome of affordances). Since the affordances are binary, robot can not evaluate the quantitative measure of likelihood of success/ failure and hence sub-goal 3 is also not solved.

Sinavpov and Stoytchev[4] extended the approach of Stoychev[3] to learn the functional taxonomies of tool so that it can detect the similarity between tools based on the outcomes of target manipulation performed with each tool. In these experiments, a simulated robot arm manipulates the target object (a puck) using six different tools T-Stick, L-Stick, Stick, L-Hook, Y-Stick, Arrow and six exploratory behaviors push, pull, slide-left, slide-right, rotate-left and rotate-right. The robot observes the different ways of target displacement. The visualization of the trajectories of target object movement for different ways of target manipulation is shown is Figure 2.2.

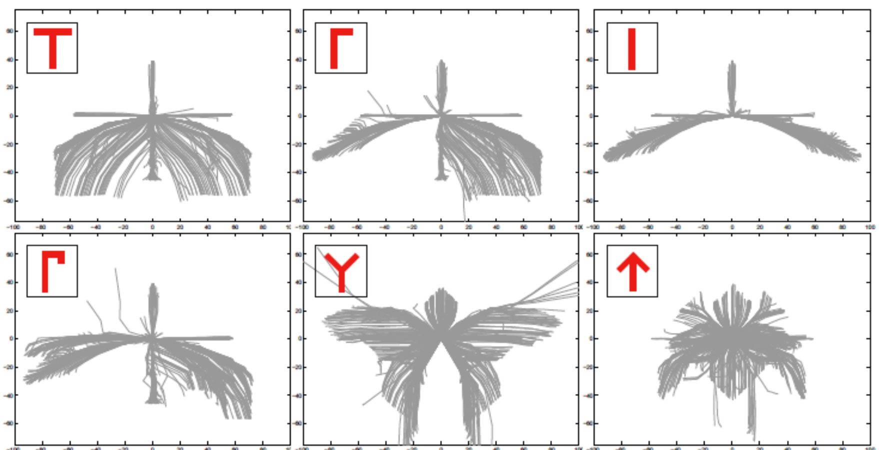


Figure 2.2: A visualization of the trajectories of target object movement manipulated using each of six user-defined behaviors with each of the six tools. All trajectories are plotted relative to the target object's initial position, which is decided randomly for each trial.

A hierarchical clustering algorithm was then used to find clusters of similar displacements where each cluster is treated as a discrete prototype of outcomes. The set of these target displacement prototypes for each tool were used to calculate a similarity measure which is used to compare one tool to another. In this way the robot is able to classify different types of tools based upon their functional abilities. However, a large number of trials (1200) were needed to generate the clusters of outcomes and detect similarity between the tools.

The benefit of this approach is that functional equivalence enables robot to determine the "suitability" of different tools to solve the similar task of object movement. Thus, robot can select available objects to be used as tools. However, the drawback is that functional taxonomies of each tool has to be learnt separately even if its similar to the previously learned tools, because the determination of functional equivalence is based on blind exploration rather than actually understanding of causally relevant features. Thus, sub-goal 1 is only partially solved.

Nishide et al.[42] proposed a neuro-scientific representation of tool in which all the multi-modal information is abstracted in two dimensional phase space of parametric bias. The drawback is that it is difficult to extract the tool representation and generalize it to unseen tools, thus sub-goal 1 is not solved. Nabeshima et al.[43,44] investigated the concept of alterable body schema and proposed temporal integration of multisensory information for robotic tool use.

Kemp and Edsinger[45] focused on learning tool manipulation from a human demonstration. Their approach is to detect and control the tip of an unknown tool. The tip of the tool is automatically extracted from image data by using a multi-scale spatio-temporal interest point operator which selects fast-moving convex shapes in the image. Human demonstration of tool-using behavior is provided to the robot and then robot attempts to mimic the tracked trajectory by using a form of feed-forward control on the arm and wrist joints. Using this method the humanoid robot is able to learn to clean a flexible hose with a brush without having any prior models of the object or tool in its knowledge base.

The interesting feature of the approach by Kemp and Edsinger[45] is that it can be used for wide variety of tools for which tool representation can be encoded in the tip of the tool. The study of Radwin et al.[46] is detailed analysis of different types of tools for which tip of the tool is considered most relevant in its functionality. Thus, the benefit of using this approach is that the knowledge of previously used tool

Figure 2.3: Kemp and Edsinger[45] demonstrated the tip detection on these tools (hotglue gun, screwdriver, bottle, electrical plug, paint brush, robot finger, pen, pliers, hammer, and scissors). The method performed best on the tools with sharp tips.

can be transferred to a new tool to solve similar tasks, thus addressing sub-goal 1. But the drawback is that it is restricted for limited category of tools for which tool representation can abstracted to the tip of the tool.

Brown et al.[47] proposed a *relational approach* to tool representation which requires modeling an explicit representation of structure and relationships between tool parts and/or the tool, target object, and environment. Regarding modeling of tools with his relational approach, Brown wrote on page 32:

> *A hammer, for example should be modeled as having a flat head which lies at a right-angle to the handle; the handle to be held near the base; and the flat heavy head of the hammer to be brought into contact with the blunt end of the nail; and the nail being held in at right-angles to the piece of wood [in which it is being inserted]*

Brown suggests expressing such relational concepts about the tools, environment and background knowledge in minute detail using inductive logic (see Muggleton and Feng[48]). The agent learns abstract actions by observing human demonstrator and executes them by providing the goals of action to a motion planner like RRT as input.

The relational approach of Brown et al.[47] can richly express tools and the situations and approach can be used across different objects and situations in a flexible manner. This work addresses sub-goal 1 since the knowledge of previously learnt tool affordances can be transferred to different tools as far as the structure of those tools is similar to the previous one. But the problem is that a tool with different structure or having different geometrical features can not be used because the casual relationship between functionally relevant features of the tool i.e. geometrical features, used action and the outcome of target manipulation is not established. Another drawback of this work is the lack of robustness due to its inability to handle noise and uncertainty of the sensors,actuators and environment, thus sub-goal 2 is only partially solved.

Nakamura and Nagai[49] proposed a novel approach to model object concepts by encoding the relationship between its appearance, usage and function. The appearances is modeled using SIFT (Lowe[50]) visual descriptors, the function is modeled by Gaussian distributions of visual changes in the tool and usage by the multinomial distribution of various features comprising of hand shape for grasping the tool, parts of the tool that support the grasp and contact points of the tool on the target object. To learn the parameters of the model, Expectation Maximization (EM) algorithm and Variational Bayesian method (see Attias[51]) is used. The learned model is used to make inferences about functions and usage from the visual appearances. A large set of tools having different functions were used e.g. scissors, pen, pliers, tweezers, cutter, stapler, glue, scotch tape, vinyl tape etc.; thus enabling robot to determine the usage of different tools.

The variational methods, bayesian in nature, depend on hyper-parameters i.e. the parameters of the prior distribution. These parameters are chosen without looking at the data and require tuning based on model validation. This "experimental tuning" requires several iterations of trial and validation, thus necessitating the need for a human developer who can actually evaluate the results, tune the parameters and re-evaluate the consequences. Thus for a autonomous robot, self-estimation of hyper-parameters, based on model validations and cross-validations is required; however, it remains a challenge. My approach is avoid such experimental tuning of hyper-parameters so that a robot can learn from its experiences without the need of an expert developer.

2.3 Conclusion

Thus, based on the literature survey, I notice the following three problems that correspond to the three sub-goals of this research (mentioned in Section 1.3):

1. The problem to realize sub-goal 1 is that robot can not determine the *relevance* of objects as a *suitable* tool because the causal probabilistic dependency between the perceptual features of objects and the desired effects of the task is not established. The robot thus can not determine which of the features of the tool (e.g. shape, size, color, material etc) are causally relevant to its functionality for the given task and what is the quantitative measure of this relevancy. For example, for hammer, the material of the tool is highly relevant, while its color is irrelevant. Thus, if robot can determine the causal relevance of the features of the tool then they can be used to define the *suitable* tool representation for solving a particular task. Such a "tool representation" may enable the robot to select unseen available objects as the tools by determining their functional equivalence with the tools previously used.

2. The problem to realize sub-goal 2 is that one-way learning of tool use model i.e. role of actions and tools in obtaining the effects, does not enable inverse estimation within the tool-use model; and the deterministic mode of learning does not enable handing of uncertainties, redundancy and irrelevancy of the action-perception, background knowledge and the environment.

3. The problem to realize sub-goal 3 is that the inference mechanisms based on deterministic model of learning do not provide the quantitative measure of the

plausibility of obtained inferences because the success and failure are judged in binary terms (i.e. with a qualitative measure). Thus, since the robot can not ascertain the confidence value of the robot in a particular situation in quantitative terms, it can not reformulate a suitable plan accordingly. Thus, if the plausibility can be assigned a probabilistic value, then when it is below a certain threshold, then robot can determine that the result of inference is less likely to achieve the desired effects with required accuracy. Then in such case, robot can take some corrective measure e.g. asking some expert tool-user, further exploration of the environment etc.

To summarize the differences between previous works with mine, the info-graphic is presented below:

X: unsolved or out of scope

△ : partially solved

◯: solved

Table 2.1: A visual description of related works and my work with respect to the problems mentioned in Chapter 1

Author	**Problem 1**	**Problem 2**	**Problem 3**
Stoytchev[38,39]	X	△	X
Sinavpov and Stoytchev[4]	X	△	X
Kemp and Edsinger[45]	△	X	◯
Brown et al.[47]	◯	△	◯
Nishide et al.[42]	△	△	X
Nakamura and Nagai[49]	△	◯	X
Jain (this study)	◯	◯	◯

Guerin et al[52] suggests that the fundamental unit to learn tool-use is to learn the sensori-motor representations that capture the essential world and object properties in terms of the actions that a robot is able to perform. These sensori-motor representations enable a robot to predict the consequences of its own behaviors and also plan them in order to achieve some effect. That is why, the concept of Affordances (despite its limitations), hold such a significant appeal in the robotics research. I shall discuss these limitations in the next chapter and adapt a different notion of affordances to present an integrated framework to solve the problems mentioned above.

Chapter 3

Approaches Used in this Thesis

The aim of this chapter is to present an integrated solution of Tool Affordances based on *functionally relevant features* of the tool to acquire the inference capabilities mentioned in Table 3.4. The acquired inference capabilities shall enable robot to emulate different tasks using various actions and tools. Also, generalization of functional features shall enable robot to estimate the effects of unseen tools.

3.1 Tool Representation Using Functional Features

In Section 1.3, I addressed the requirements of a tool representation that can be generalized to a wide variety of available objects which can be used as tools. I propose a novel approach to tool representation grounded in a robot's determination of casually relevant features of the tool that influence its functionality for the given task (e.g., shape, material, etc.) and discard the irrelevant features (e.g., color, texture, etc). I term these the *functional features* of the tool for the given task. Table 3.1 lists examples of functional features of some tools and required functionalities of the tasks. In this study, I focus only on the peculiar *geometry* of the constituent part of the tool that is used to manipulate the target object to fulfill the functionality of the task.

Table 3.1: Functional features of various tools required for different tasks.

Task Objective	Functionality of Tool	Candidate Tools	functional features
1. Extend Reach	Move object	Stick , L-shaped tool	flat surface, corners
2. Transfer liquids	Hold liquid	Spoon, Glass	non-convex shape
3. Deform object shapes	Cut fruits	Knife, Blade	sharp edge
4. Amplify physical strength	Insert nail	Stone, Hammer	mass, material

I argue that the functional features remain invariant across different types of tools that offer similar functionality and that generalization of functional features enables the robot to estimate the effects of unknown tools. For instance, as shown in Table 3.1, both knife and blade have the functionality of cutting, since both have sharp edges. My hypothesis is that a tool use model learnt using functional features can be generalized to a wide variety of objects that share similar functional features. Thus, when a robot is asked to realize an effect which it had previously realized using a different tool, it can search for a suitable new tool on the basis of the similarity of previously used functional features. This approach is likely to address the Problem 1 mentioned in Section 1.3.

3.2 Representation of Perceived Tool Affordances

Informally defined as the "action possibilities", there is a clear lack of consensus within the ecological psychology community on what accordance means. Due to the lack of a consensus over the concept of affordances, it is not clear that whether affordances are the properties of an individual agent (its "effectiveness" or its complete body) or of the environment or of combined agent-environment system. Chemero[53] discusses in details the multiple interpretations of affordances and the major points of disagreement over them; also from the perspective of robotics (Chemero et al.[54]). Sahil et al.[55] recently proposed three different perspectives: the agent, the environment, or an observer to view the affordances.

Since there are many possibilities for applying the concept of affordances to the design of artificial agents many affordance-based approaches have been developed. Affordances have been used to develop a wide range of AI behaviors for artificial agents. These behaviors include controlling the robot for its traversal and obstacle avoidance (Murphy et al.[56], Cakmak et al.[57], Sahil et al.[55], Paleta et al.[58], Brock et al.[59], Ugur et al.[60], Erdemir et al.[61,62]), grasping behaviors (Montesano et al.[63], Detry et al.[64–66], Cos-Aguilera et al.[67–69], Kraft et al.[70], Sweeney at al.[71]), manipulation of target object to determine its properties using primitive actions like pushing, poking, pulling, lifting and rotating (Fitzpatrick et al.[35], Mondoval et al.[72], Sun et al.[73], Dag et al.[74], Fritz et al.[75]). Affordances are also proposed to learn target manipulation with the use of hand-held tools e.g with the use of sticks of various shapes (Wood et al.[76], Stoytchev[38,39], Sinapov and Stoytchev[77]) where target is manpulated in 2D surfaces using simple pre-programmed actions and also for learning more complex behaviors e.g. driving a bolt into a slot using a screwdriver tool (Hortol et al.[78,79]).

According to McGrenere and Ho[1], Gibsonian affordances mean "*an action possibility available in the environment to an individual, independent of the individual's ability to perceive this possibility*". In context of tool use, a tool can have certain affordance but the information specifying its affordance are not available to the robot. For example, a power tool has an affordance to drill a hole into the wall, but the button to operate but can be hidden, camouflaged or likewise. Thus, Gibson's affordance do not take into account the perceived properies of an object to define its affordance. It exists relative to the action capabilities of a particular agent, e.g. the affordance of hammering a nail exists for a robot with manipulator arms capable enough to handle the hammer but does not exist for the robot with such action possibility. Also, it does not change according to the needs and goals of the agent.

On the contrary, according to Norman[80] an affordance as a combination of both actual and perceived properties of an object. Norman wrote:

> *"...the term affordance refers to the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used. [...] Affordances provide strong clues to the operations of things. Plates are for pushing. Knobs are for turning. Slots are for inserting things into. Balls are for throwing or bouncing. When affordances are taken advantage of, the user knows what to do just by looking: no picture, label, or instruction needed."*

In context of tool use, Norman's affordance should suggest how it should be used; and give a visual clue to its function and use. For example, the affordance of a hammer is both its elongated handle, heavy head, its material etc. (its actual properties) as well as suggestion as to how it should be used e.g. attributes relevant for its grasping at the end of its elongated handle (its perceived properties). Thus, Norman's affordances assume that information in the sensory receptors of an agent is good enough to perceive anything and do not need any higher-level cognitive processes to mediate between it's sensory experience and the perception (Sternberg[81]). Also Norman, unlike Gibson, does not disregard the knowledge, prior experience and expectation of the agent. The reader is suggested to refer Table 3.2 for comparison of the concept of affordances as defined by Gibson[41] and Norman[80].

Thus, in this thesis, I am using the notion of ***perceived affordances*** , the concept proposed by Norman[82–84], rather than Gibsonian's affordances. In the words on Norman:

> *Its very important to distinguish real from perceived affordances. Design is about both, but the perceived affordances are what determine usability. I*

didnt make this point sufficiently clear in my book and I have spent much time trying to clarify the now widespread misuse of the term.

Table 3.2: Comparison of affordances as defined by Gibson and Norman (McGrenere and Ho[1]).

Gibson's Affordances	**Norman's Affordances**
• Action possibilities in the environment in relation to the action capabilities of an actor • Independent of the actor's experience, knowledge, cultre, or ability to perceive • Existence is binary - an affordance exists or it does not exist.	• Perceived properties that may not actually exist • Suggestions or clues as to how to use the properties • Can be dependent on the experience, knowledge, or culture of the actor • Can make an action difficult or easy

In my work, I assume the existence of several tool affordances to solve the similar task and also assume that a robot can generate the actions difficult or easy and exploit the perceivable function of the tool. This is in accordance with the Norman's view of affordances. When Norman's concept of affordances is applied to the everyday artifacts e.g. in doors of a house, thin vertical doorhandles afford pulling, while flat horizontal plates afford pushing (Gaver[85]). Both these affordances result in same effect i.e. opening a closed door. Similarly, for a tool using task of pulling an object, L-shape tool affords *hooking* an object to pull it, while a tweezer tool affords clamping of object through the application of inward pressure with its sides and lifting the object. Using both of the afforances, task of moving the object can be accomplished.

Thus, my view of tool affordances is implicitly linked with its outcome. Two tool affordances are considered *equal* as far as they produce the *similar* effect, irrespective of which tool is used and how it is used. An effect is defined as both quantitative and qualitative change in some property of target object. An action is defined as the movement of tool that changes one or more properties of the target object. Two tasks are also considered *similar* when their effects match qualitatively rather than quantitatively. For example, in case of the task of *object movement*, the desired effect of the task is to move a target object from place to other in a specific direction. Thus the movement of object for 5 cm or 50 cm or any distance are considered the same task as far as their angular movement directions are within certain threshold. Table 3.3 defines the quantitative measure of similarity of different tasks performed with the same objective.

Table 3.3: Two different tasks with the same objective are considered similar when their effects match in qualitative terms.

Task Objective	**Candidate Tools**	**Notion of similarity with the effects of other tasks**
1. Object Movement	Stick, L shaped Tool, Rake	angular movement directions of target object are within 30 degree of each other
2. Cutting an object into pieces	Knife, blade, Scissor	number of connected components of the target object change after the manipulation
3. Inserting nail into the wood	Stone, Hammer	contour of target object changes after the manipulation

However since in real world situations, it may not have access to previously used tools and/or the knowledge of previously used actions may be insufficient, I intend to enable robot learn *different* tool affordances in such a way that they can be used *alternatively* by the robot to solve *similar* tool using tasks.

For example, if robot acquires the tool affordances (through learning, pre-programming etc) of pulling the object with a L-shape tool that ”hooks” the box and changes its position by pulling it using some specific force. Now, consider the case when the initial position of target object is a different from the situation for which tool affordances Ire initially acquired, while the goal position remains the same. In this case, to perform the similar task of *object movement* using the same L-shaped tool, the robot should be able to reason about parameters of actions in

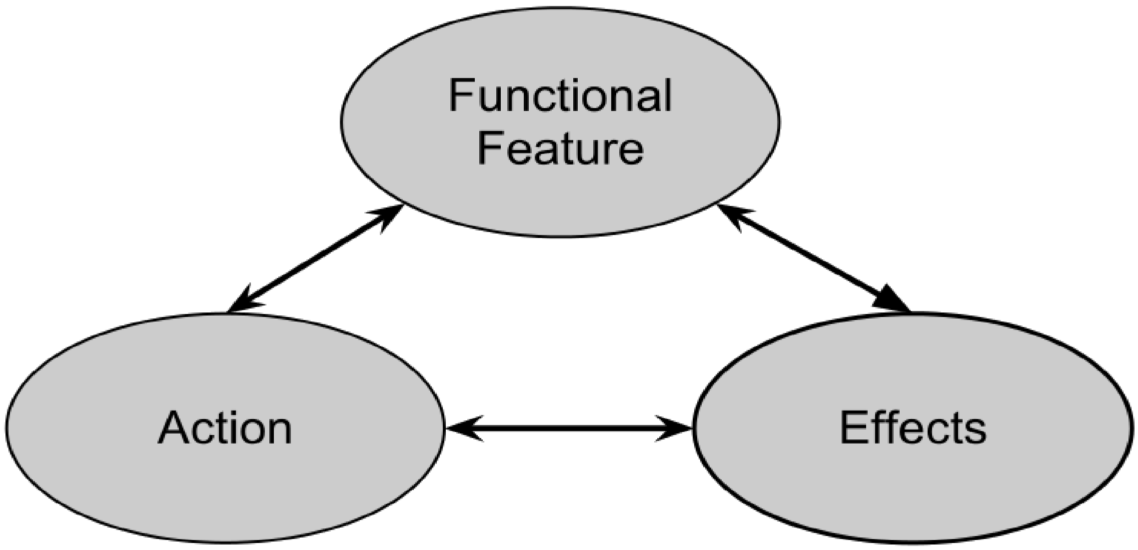


Figure 3.1: Framework of Tool Affordances

terms of abstract properties such as linear forces or torques and generate the action that can move target to the goal position. Now consider a more complex case when robot does not even have access to L-shaped tool, then it must find a different tool that has similar function. Thus, upon availability of different tools and action schemas, an intelligent robot is expected to estimate their tool affordances using its prior experience so that an alternative solution to solve a task can be explored when required.

It means that robot should be able to determine the different ways of solving a *similar* task using different actions and different tools. This also requires that robot should be able to make predictions and inferences based on previous experiences of target manipulation. To address the Problem 2, I propose learning causal bi-directional relationships between the tool-using action, functional features of the tool, and effect of target manipulation. I propose the framework of *Tool Affordances* (Fig 3.1) to perform such learning using the probabilistic framework of Bayesian Networks (BN).

The Bayesian network (BN) is chosen as the implementation of learning and evaluation of Tool Affordances to acquire the inference capabilities listed in Tabl 3.4. My

Table 3.4: Inference capabilities required for an autonomous tool user to solve a tool-using task in different situations.

Inference	**Input**	**Target of Inference**	**Objective**
Inference 1	Tool Representation ,Effects	Action	To select a suitable action and estimate its parameters.
Inference 2	Action,Effects	Tool Representation	To select a suitable tool by determining its function.
Inference 3	Effects	Action,Tool Representation	To select a suitable action and estimate its parameters along with selecting a suitable tool by determining its function.
Inference 4	Action,Tool Representation	Effects	Estimation the outcome of tool affordances i.e. probable effects

argument about robot requiring a large set of inference capabilities listed in Table 3.4 is also supported by recent studies in neuroscience. Grafton and Hamilton[86] argued about the existence of hierarchical topology in human brain for generating goal directed actions. They show from the experiments the three levels of this hierarchy. The desired *effects* of an action is at top level, followed by the tool which is to be manipulated and tool using action at lowest level. This hierarchical topology enables inference 1. Studies also suggest that recognition of tools is done in the same brain region for action word generation(Grafton and Hamilton[86], Martin et al.[87], Martin et al.[88]). This indicates that tool representation is closely correlated with the actions that can be generated from these tools (Koechlin and Jubault[89]). Such correlation enables the inference of tools though the knowledge of realized effects and used action (inference 2). Tool use skill should enable robot to recognize tools(and its functions) by observing the used action and effects on target object. For example, a person performing a nailing action can be understood to have a hammer in his hand. Such learning shall also enable robot to select suitable tools to realize the required effects using some given action. Thus inference 3 is required for robots capability of tool recognition and selection.

Representing tool affordances with the BN enables robot to use a rich class of inference and learning algorithms that comes with graphical models. I chose BN because of its ability to model probabilistic dependencies between the data. It enables us to handle incomplete and irrelevant data as well as allows us to input domain and background knowledge with a degree of confidence and relevance (see Pearl[90,91]). The probabilistic semantics of the BN can be used to compute the probability distribution over the possible outcomes by using a limited number of learning samples. In addition, when there is uncertainty about the actions, tool representations and effects, the robot can make queries to its human user by referring to uncertainties expressed in terms of probability. Hence, BN enables the robot to learn how to use tools by making probabilistic queries for requesting knowledge, instructions, and demonstrations from the user. Thus, modeling tool affordances using the probabilistic semantics of BN gives us an edge over standard statistical methods (linear regression, logistic regression) as well as over the non-probabilistic machine learning methods (e.g., support vector machines, classification and regression trees, random forests, neural networks, nearest neighbor algorithms).

3.3 The proposed approaches

Thus, my proposed three approaches are:

1. To solve Problem 1 in order to realize sub-Goal 1, I propose a novel approach of learning the perceptual features of the tool causally relevant to realize the desired effects of the task. These features are termed as *functional features* of the tool. This approach enables a robot to select available objects that share those functional features to be used as tools. It also enables predicting the effects of different tools on the basis of their functional features. To determine

the causal probabilistic dependency between features of tool and the outcome of target manipulation, robot performs multiple repetitions of the manipulation of target object via the tool e.g., geometry, color, size, material, etc. For instance, to bring a distant object closer, the tool should have a "hook" to pull the target object. Thus, a peculiar geometry in this case is causally relevant while color and texture of the tool is not. These causally relevant features are termed as *functional features* of the tool for the given task. Thus, with the knowledge of "hook" shaped geometry as the functional feature, a robot can determine that *stick* is clearly inappropriate tool while a *rake* is appropriate. Table 3.1 lists examples of functional features of some tools and required functionality of the tasks.

2. To solve Problem 2 in order to realize sub-Goal 2, I propose an approach of bayesian learning of tool-use model termed as *tool affordances* to enable robot perform causal and diagnostic reasoning. The function of tool affordances is modeled using the probabilistic graphical model of **Bayesian Networks** (BN) . The probabilistic semantics of BN enables a robot to handle various uncertainties in its action-perception, learning, inference, environment etc. To obtain the sensory-motor experiences of target manipulation, a robot performs a random exploration of the environment i.e. manipulate target object by randomly varying actions (and their parameters) and the functional features of the tool (i.e. tool representation). Using the plethora of obtained sensory-motor experiences, it should then learn the causal relationships between the tool-using action, functional features of the tool, and effect of target manipulation (Fig 3.1) and acquire the inferences capabilities (listed in Table 3.4). BN is chosen because it comes with rich class of learning algorithms to model probabilistic dependencies between the data and mechanisms of inferences capable of dealing with partial observations, incomplete and irrelevant data (see Pearl[90,91]).

3. To solve Problem 3 in order to realize sub-Goal 3, I propose a novel approach to determine the quantitative measure of the *plausibility* of inferences made in different situations. The gap between the desired outcome of the task with the probable outcome predicted using the inferred value as an input is calculated. The gap value is used to compare their match, the smaller the gap the better is the match and hence more is the plausibility of the result of inference. Thus, based on the different values of gap measure a robot can reformulated its plan e.g. it may request human feedback or further explore the environment.

In the next chapter, I will discuss the experiments performed to acquire the inference capabilities mentioned in this chapter, the used method and the obtained results.

Chapter 4

Learning of Tool Affordances

The aim of this chapter is to present the experiments, method and results for acquiring the inference capabilities mentioned in Table 3.4.

4.1 Method

I implemented the framework shown in Figure 3.1 using the *Bayesian Network* (BN) shown in Figure 4.4. BN is a directed probabilistic graphical model in which nodes represent random variables, and the arcs (or lack of) represent conditional independence assumptions. The conditional independence relationship encoded in a Bayesian network can be stated as follows: a node is independent of its ancestors given its parents, where the ancestor/parent relationship is with respect to some fixed topological ordering of the nodes. The conditional independence provides a compact representation of joint probability distributions. One can regard an arc from action to target placement as indicating that action "causes" the target to be displaced. This notion of causality is used as a guide to construct the graph structure shown in Figure 4.4. The oval nodes represent continuous data stored using Gaussian nodes with higher granularity, while the rectangular boxes are representations of discrete data. The discrete variables are represented as a conditional probability table (CPT), which lists the probability that the child node takes on each of its different values for each combination of values of its parents.

Two things are required to describe a BN: the graph topology (structure) and the parameters of each conditional probability distribution (CPD). It is possible to learn both of these from data (Heckerman[92]). The aim of the experiments presented in this chapter is to confirm the hypothesis that functional feature based bayesian learning of tool affordances enable a robot acquite bi-directional inference capabilities. Thus, the problem of learning the structure was not considered. A brief discussion on the categories of structure learning algorithm and references is presented in Section 6.1. I provided the robot with the graph structure of the BN, and CPD was specified at each node. The parameters of each CPD of the model were learned using maximum likelihood parameter estimation (Bishop[93]). The algorithm is discussed in Section 6.1.1.

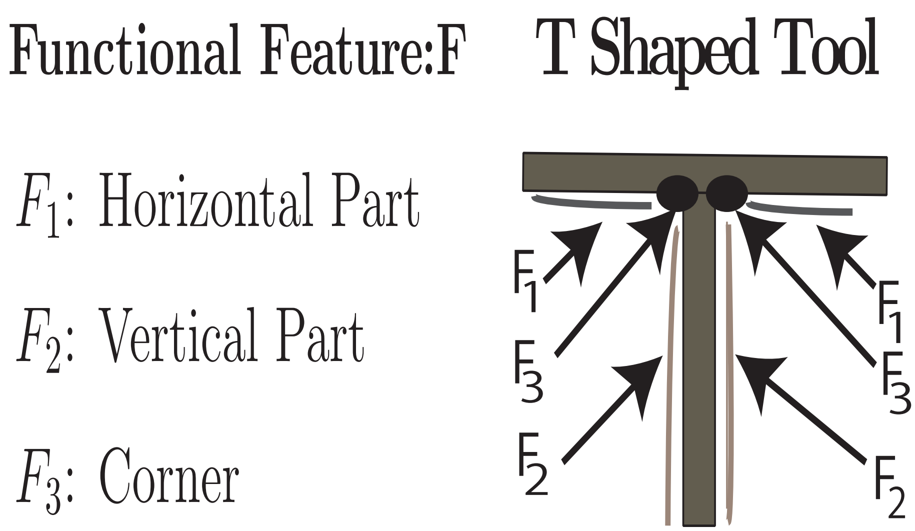


Figure 4.1: Functional features of T-shaped tool are shown. The brown lines indicate that for functional features F_1 and F_2 entire flat surface of the tool can be used to manipulate the target object. Please note that the the notion of functional feature in this chapter is based on the geometrical feature of the tool and is independent of the location of that geometrical feature. That is why there is a single category for each available horizontal and vertical surfaces and corner respectively.

In my experiments, the required effect of the task is to *move* the target object in different directions. To do that, robot is equipped with actions and a tool. The used T-shaped tool has 3 geometrical features to be used as functional features, as shown in Figure 4.1. They are 1) the horizontal part, 2) vertical part, and 3) corner. Note that both the *horizontal part* and *vertical part* have flat surfaces, but different orientations relative to the handle of the tool. A cubic target object o is placed in proximity of each functional feature F_j of the T-shaped tool, as shown in Figure 4.2, where $j \in [1, 3]$.

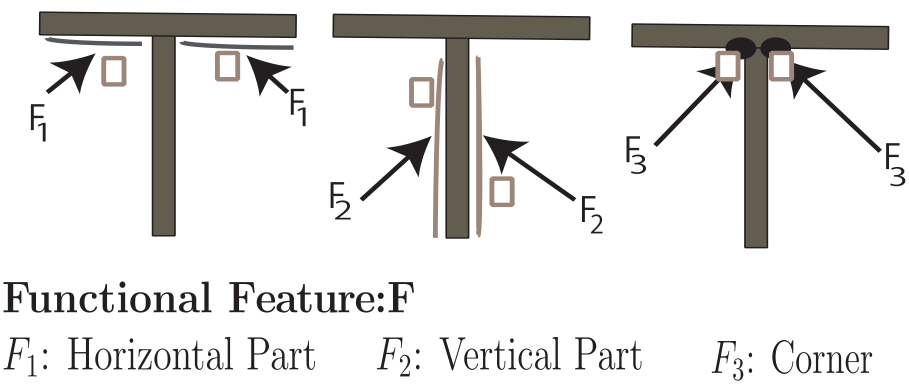


Figure 4.2: The placement of target object relative to the functional features. The square shape denotes the cubic target object. The lines closes to the functional features F_1 and F_2 indicate that a target object is placed randomly near the surface in different manipulation trials.

For target manipulation, a force is applied to the tool. There are 5 discrete directions in which robot is pre-programmed to move the tool. The the magnitude of velocity is continuous. The tool velocities in each of the 5 discrete directions V_i (for $i \in [1, 5]$) are indexed with a specific action label A_i. The A_i thus refers to specific direction of the movement of the tool. The discrete action is parameterized with continuous magnitude of the velocity of tool in that particular direction (refer Figure 4.3).

The pre-designed action A_i and functional feature F_j is used to manipulate the tool, which in turn manipulates the object, where $i \in [1, 5]$, $j \in [1, 3]$. The action labeling and its semantic meaning are:

- A_1: Contract Arm
- A_2: Slide Arm Left
- A_3: Pull Diagonally-1
- A_4: Slide Arm Right
- A_5: Pull Diagonally-2

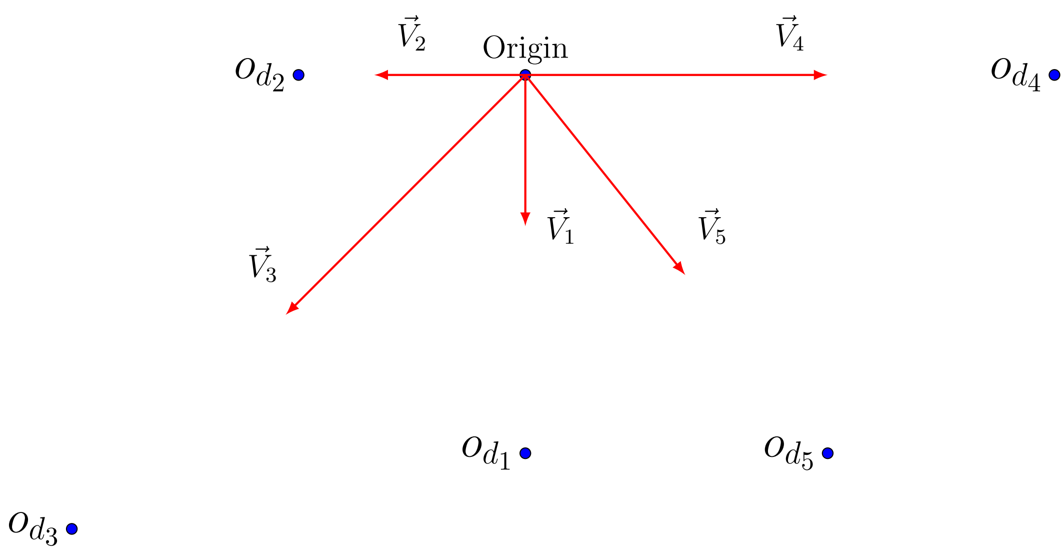


Figure 4.3: The tool is applied at once with a force that moves it in specific direction and initial velocity $V_i \in \mathbb{R}^2$ and $i \in [1, 5]$, where i denotes the direction of tool velocity. In total, tool moves in 5 directions. The magnitude of tool velocity changes for each manipulation trial, thus the observed target displacement also changes according to the tool's velocity. Here one such case for target manipulation is shown for each of the 5 cases of tool movement. The length of arrows (in red) denote the magnitude of tool velocity and o_{d_i} is the resulting target displacement.

Robot uses the five pre-designed actions as mentioned above and their corresponding movements are shown in Figure 4.3. The tool has the three functional features as shown in Figure 4.1 using which the target is manipulated. The robot records the observations during the manipulation, as shown in Table 4.1. Thus, in total, there are $5 \cdot 3 = 15$ combinations of *Action* and *functional features* that can be used to manipulate the object and create the effect. All experiments on manipulation and data acquisition were performed with a simulated T shaped Tool using *Webots* [1] simulation software.

The following two effects were measured:

1. The final displacement of the target object $D \in \mathbb{R}^2$ between the end position and initial position.

2. The velocity of the target object measured immediately after its impact with the tool $W \in \mathbb{R}^2$.

Table 4.1: The notation of robot's observations during a manipulation trial.

Node	Description	Random Variable	Value
A	Set of actions	A	A_i , where $i \in [1, 5]$
F	Set of functional features	F	F_j , where $j \in [1, 3]$
D	Final target displacement	$D \in \mathbb{R}^2$	$d_{ij} \in \mathbb{R}^2$
X	Initial position of object relative to used F_j	$X_o^F \in \mathbb{R}^2$	$X_o^{F_j} \in \mathbb{R}^2$
V	Initial velocity of tool before it hits the target	$V \in \mathbb{R}^2$	$v_i \in \mathbb{R}^2$
W	Velocity of object after being hit by the tool	$W \in \mathbb{R}^2$	$w_{ij} \in \mathbb{R}^2$

Two sets of experiments are performed:

1. For each combination of action and functional feature, the target object is manipulated by the tool only once. This experiment and its result is detailed in Section 4.2.

2. For each combination of action and functional feature, the target object is manipulated by the tool multiple times. For each such manipulation trial (in which

[1] http://www.cyberbotics.com/

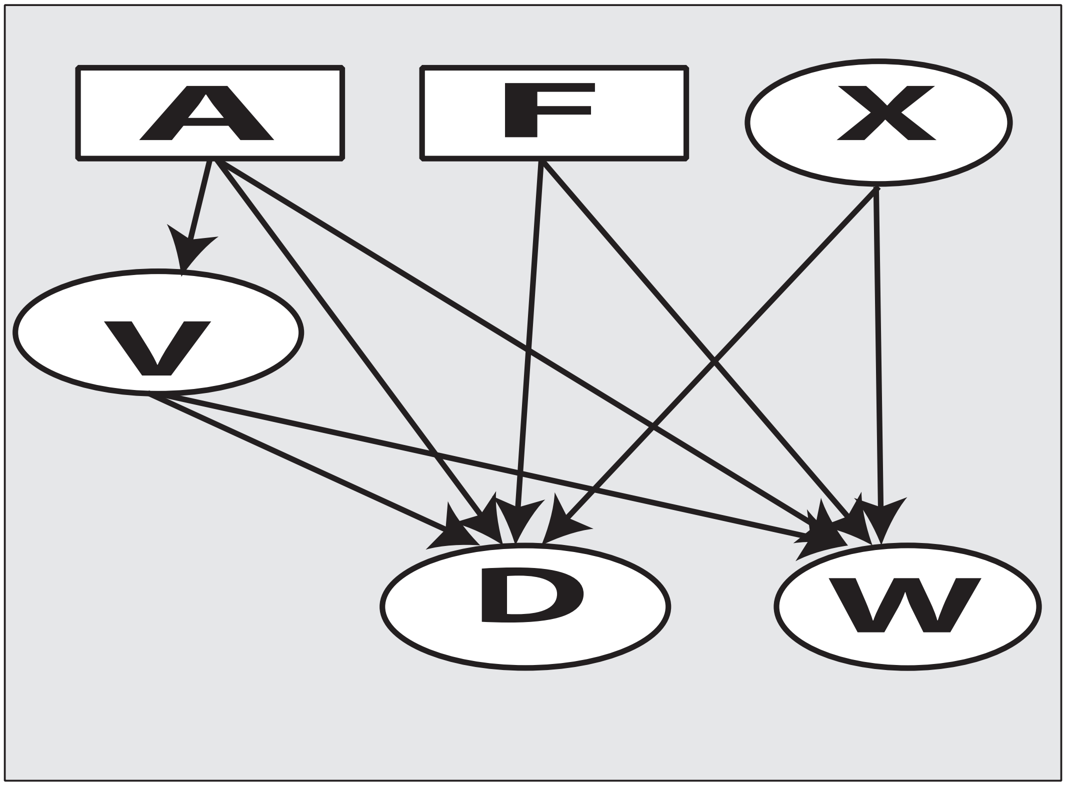


Figure 4.4: Structure of Bayesian network to learn the proposed tool affordances proposed in Figure 3.1. The notation is described in Table 4.1.

an action is executed with some functional feature), some random variation in the force applied on the tool takes place. Thus, a different target velocity and target displacement is observed for each manipulation trial. This experiment and its result is detailed in Section 4.3.

4.2 Experiment 1: Manipulation of target object when a fixed linear force is applied on the tool.

One manipulation trial was performed for each pair (A_i, F_j) and their observations were recorded shown in Table 4.1 (a total 15 pairs). The effects observed during the manipulation trial of one pair of (A_i, F_j) is called as one sample. Only one sample is taken for each pair of pair of (A_i, F_j) to learn the tool affordances. However, one such sample is also taken to make inferences from the learned tool affordances and one additional sample is taken to cross-validate the plausibility of the obtained result of inference. The observation data from all the samples except (A_5, F_3) was used in the learning process. The reason, I do not use the data from (A_5, F_3) for learning is that, I am interested in evaluating the capability of learned affordances for the case when the observations are novel or unseen to the robot. Thus the evaluation sample from (A_5, F_3) is mentioned as *novel* or *unseen* effect. Hence, in total, there were 14 samples for learning the tool affordances and all 15 samples are used for the evaluation i.e. making the inferences.

After the robot has learned the tool affordances, I evaluated the robot's capability to emulate the desired effects and test the inference capabilities mentioned in Table 3.4. The effects created for the specific purpose of evaluation are given by the human user to the robot. Hence, the data corresponding to that purpose is denoted using superscript (h), while the data used for learning is denoted using superscript (r).

As mentioned above, one manipulation trial was performed for each pair of (A_i, F_j) $(i \in [1,5],\ j \in [1,3])$. The robot observed the effects of the manipulation on the target object $(d_{ij}^{(h)}, w_{ij}^{(h)})$ given by the human demonstrator (denoted by superscript (h)) using action A_i and functional feature F_j, where the $i \in [1,5]$ and $(j) \in [1,3]$. The task of the robot was to emulate the effects presented to it by estimating the inference target for the given inputs as shown in Table 4.2.

Table 4.2: Input and output of tool affordances for inference tests mentioned in Table 3.4.

Test	**Input**	**Target Inference**
1	$(d_{ij}^{(h)}, w_{ij}^{(h)})$,$v_k^{(h)}$, A_k	F_m
2	$(d_{ij}^{(h)}, w_{ij}^{(h)})$, F_m	A_k
3	$(d_{ij}^{(h)}, w_{ij}^{(h)})$	A_k,F_m
4	A_i,F_j ,$v_i^{(h)}$	$(d_{ij}^{(r)}, w_{ij}^{(r)})$

Using the learned tool affordances, probabilistic inference was performed with the junction-tree inference engine (Huang and Darwiche[94]). The engine computes the marginal probability distribution of members of a set of variables, given the evidence. The engine was supplied with a Bayesian network of tool affordances (see Figure 4.4) along with the evidence and weight for each variable in the network. The results for acquiring the inference capability 1 are presented in Figure 4.5. The objective of inference 1 is to select a suitable tool to realize the target effects using an arbitrarily given action. Since the tool in my approach is represented using functional features, to select a suitable tool, the robot inferred the functional features to realize the desired effects using arbitrarily given actions.

4.2.1 Results of the Experiment 1 and their Discussion

The probability of a robot estimating a suitable functional feature F_m to realize the effects given during the evaluation process $d_{ij}^{(h)} \in \mathbb{R}^2$ and $w_{ij}^{(h)} \in \mathbb{R}^2$ using the given arbitrarily action A_k, where h and r denote the human user and robot, respectively, and $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$. The inference calculated using the following equation is shown in Figure 4.5.

$$P_F = P(F = F_m | A_k, v_k^{(h)}, d_{ij}^{(h)}, w_{ij}^{(h)}) \tag{4.1}$$

For example, to realize $(d_{11}^{(h)}, w_{11}^{(h)})$ using action A_1, robot estimates F_1 as a suitable functional feature. To determine whether the inferred F_m is suitable, the robot calculates the gap between the effects given during evaluation process i.e. $(d_{ij}^{(h)}, w_{ij}^{(h)})$ and the actual effects $(d_{km}^{(r)}, w_{km}^{(r)})$ resulting from the manipulation of action A_k and functional feature F_m by the robot(denoted by the superscript (r)). The gap between these effects in $D \in \mathbb{R}^2$ and $W \in \mathbb{R}^2$ is defined as follows:

$$\epsilon(d_{ij}^{(r)}) \overset{def}{=} d_{km}^{(r)} - d_{ij}^{(h)} \tag{4.2}$$

$$\epsilon(w_{ij}^{(r)}) \stackrel{def}{=} w_{km}^{(r)} - w_{ij}^{(h)} \tag{4.3}$$

$$\frac{\sum\limits_{n=1} \Bigg(\sum\limits_{i=1}^{5} \sum\limits_{j=1}^{3} \Big[\sum\limits_{k=1}^{5} \Big(\sum\limits_{m} (d_{km}^{(r)} - d_{ij}^{(h)}) \Big) \Big] \Bigg)}{\sum\limits_{n=1} \Big[\sum\limits_{i=1}^{5} \sum\limits_{j=1}^{3} \Big(\sum\limits_{k=1}^{5} \sum\limits_{j=1}^{3} (\delta_{j\,m}) \Big) \Big]} \tag{4.4}$$

$$\frac{\sum\limits_{n=1} \Bigg(\sum\limits_{i=1}^{5} \sum\limits_{j=1}^{3} \Big[\sum\limits_{k=1}^{5} \Big(\sum\limits_{m} (w_{km}^{(r)} - w_{ij}^{(h)}) \Big) \Big] \Bigg)}{\sum\limits_{n=1} \Big[\sum\limits_{i=1}^{5} \sum\limits_{j=1}^{3} \Big(\sum\limits_{k=1}^{5} \sum\limits_{j=1}^{3} (\delta_{j\,m}) \Big) \Big]} \tag{4.5}$$

Discussions on the result of inference

The learned tool affordances enable robot to emulate desired effects (including the novel effects) using several different combinations of action and functional features. Figure 4.6 and Figure 4.7 show the the gap matrices $\epsilon(D)$ and $\epsilon(W)$ with greyscale shading corresponding to the gap values of $\epsilon(d_{ij}^{(r)})$ and $\epsilon(w_{ij}^{(r)})$, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$.

For any suitable inferred F_m, the gap should be smaller than gaps that correspond to non-inferred F_m. I analysed the gap calculated using Equation 4.6 for the values shown in Figure 4.6 and the gaps calculated using Equation 4.7 for the values shown in Figure 4.7. The analysis shows that gaps corresponding to the inferred functional feature F_m (shown using '*') are much narrower than the gaps corresponding to non-inferred functional feature F_m. Figure 4.8 compares gap values corresponding to the inferred F_m with those corresponding to non-inferred F_m. The bars clearly show the inferred F_m for any arbitrarily given A_k is most suitable tool affordance to emulate both learned and novel desired effects.

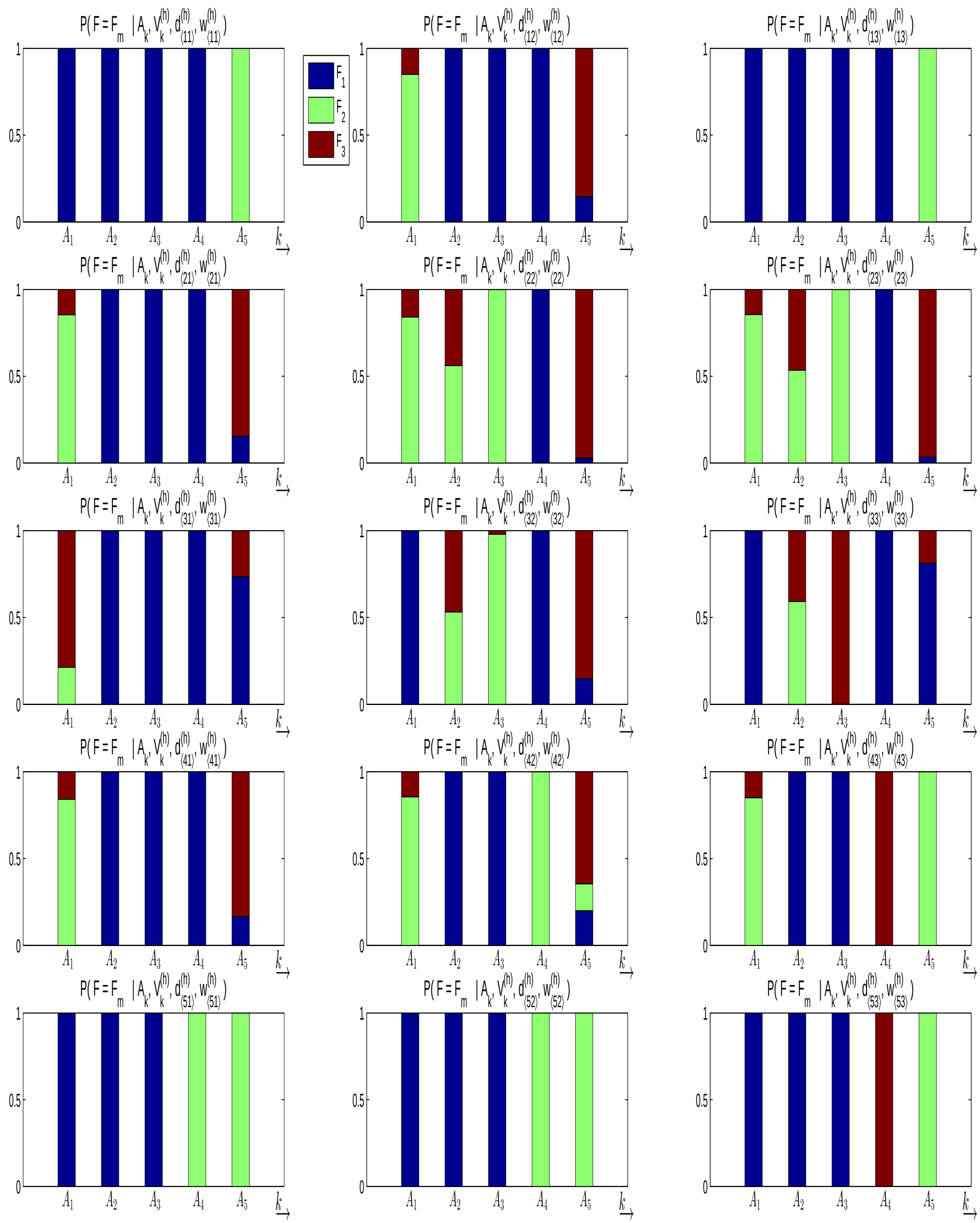


Figure 4.5: Robot estimation of suitable functional features to realize effects given by a human demonstrator $(d_{ij}^{(h)}, w_{ij}^{(h)})$ (measured from the manipulation of action A_i and functional feature F_j) using the arbitrarily given action A_k, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$

$d^{(h)}_{ij}$ \ $d^{(r)}_{km}$	$d^{(r)}_{11}$	$d^{(r)}_{12}$	$d^{(r)}_{13}$	$d^{(r)}_{21}$	$d^{(r)}_{22}$	$d^{(r)}_{23}$	$d^{(r)}_{31}$	$d^{(r)}_{32}$	$d^{(r)}_{33}$	$d^{(r)}_{41}$	$d^{(r)}_{42}$	$d^{(r)}_{43}$	$d^{(r)}_{51}$	$d^{(r)}_{52}$	$d^{(r)}_{53}$
$d^{(h)}_{11}$	0.06*	0.12	0.08	0.12*	0.28	0.25	0.10*	0.21	0.33	0.12*	0.14	0.25	0.10	0.19*	0.13
$d^{(h)}_{12}$	0.18	0.00*	0.20*	0.00*	0.26	0.22	0.20*	0.22	0.40	0.00*	0.08	0.22	0.20*	0.28	0.16*
$d^{(h)}_{13}$	0.02*	0.20	0.00	0.20*	0.32	0.30	0.07*	0.24	0.29	0.20*	0.21	0.30	0.07	0.14*	0.17
$d^{(h)}_{21}$	0.18	0.00*	0.20*	0.00*	0.26	0.22	0.20*	0.22	0.40	0.00*	0.08	0.22	0.20*	0.28	0.16*
$d^{(h)}_{22}$	0.31	0.26*	0.32*	0.26	0.00*	0.03*	0.26	0.09*	0.28	0.26*	0.33	0.48	0.37*	0.46	0.39*
$d^{(h)}_{23}$	0.28	0.22*	0.30*	0.22	0.03*	0.00*	0.24	0.07*	0.29	0.22*	0.30	0.44	0.35*	0.43	0.36*
$d^{(h)}_{31}$	0.07	0.19*	0.07*	0.19*	0.26	0.23	0.01*	0.17	0.24	0.19*	0.23	0.34	0.14*	0.21	0.22*
$d^{(h)}_{32}$	0.23*	0.22	0.24	0.22	0.09*	0.07*	0.18	0.00*	0.22*	0.22*	0.29	0.43	0.30*	0.38	0.34*
$d^{(h)}_{33}$	0.30*	0.40	0.29	0.40	0.28*	0.29*	0.23	0.22	0.00*	0.40*	0.46	0.58	0.36*	0.41	0.45*
$d^{(h)}_{41}$	0.18	0.00*	0.20*	0.00*	0.26	0.22	0.20*	0.22	0.40	0.00*	0.08	0.22	0.20*	0.28	0.16*
$d^{(h)}_{42}$	0.19	0.08*	0.21*	0.08*	0.33	0.30	0.24*	0.29	0.46	0.08	0.00*	0.15	0.19*	0.26*	0.10*
$d^{(h)}_{43}$	0.28	0.22*	0.30*	0.22*	0.48	0.45	0.35*	0.44	0.58	0.22	0.15	0.00*	0.24	0.27*	0.13
$d^{(h)}_{51}$	0.07*	0.20	0.07	0.20*	0.37	0.35	0.14*	0.30	0.36	0.20	0.19*	0.24	0.00	0.09*	0.12
$d^{(h)}_{52}$	0.16*	0.28	0.16	0.28*	0.47	0.44	0.23*	0.39	0.44	0.28	0.24*	0.24	0.10	0.03*	0.14
$d^{(h)}_{53}$	0.23*	0.22	0.24	0.22*	0.46	0.43	0.30*	0.41	0.53	0.21	0.15	0.07*	0.18	0.20*	0.08

The desired effects given by the human for the robot to realize

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.6: Matrix $\mathbf{E}(\mathrm{D})$ representing the gap in the final displacement of target object D, between experienced displacement $d^{(h)}_{ij}$ (given during the evaluation process) and actual displacement $d^{(r)}_{km}$ (he result of action A_k and functional feature F_m, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$). The gap calculated using Equation 4.6 is represented by greyscale parametric shading, i.e., the wider the gap, the darker the background. For the gap corresponding to the actually inferred k shown in Figure 4.5, the '*' mark is used; hence, a narrower gap is expected for a better inference. $d^{(r)}_{53} \in \mathbb{R}^2$ was not used during the learning process, but $d^{(h)}_{53} \in \mathbb{R}^2$ was given during the evaluation process as *novel effect*, as shown in the last row.

The desired effects given by the human for the robot to realize

$w_{ij}^{(h)}$ \ $w_{km}^{(r)}$	$w_{11}^{(r)}$	$w_{12}^{(r)}$	$w_{13}^{(r)}$	$w_{21}^{(r)}$	$w_{22}^{(r)}$	$w_{23}^{(r)}$	$w_{31}^{(r)}$	$w_{32}^{(r)}$	$w_{33}^{(r)}$	$w_{41}^{(r)}$	$w_{42}^{(r)}$	$w_{43}^{(r)}$	$w_{51}^{(r)}$	$w_{52}^{(r)}$	$w_{53}^{(r)}$
$w_{11}^{(h)}$	1.64*	1.64	0.54	1.64*	2.74	2.74	1.50*	2.19	2.25	1.60*	2.19	2.67	1.49	1.52*	1.35
$w_{12}^{(h)}$	0.00	0.00*	2.17*	0.00*	2.18	2.18	2.56*	2.56	3.08	0.03*	1.47	2.18	2.58*	2.11	2.14*
$w_{13}^{(h)}$	2.17*	2.17	0.00	2.17*	3.08	3.08	1.40*	2.30	2.17	2.14*	2.62	3.02	1.41	1.68*	1.46
$w_{21}^{(h)}$	0.02	0.02*	2.16*	0.02*	2.18	2.18	2.54*	2.55	3.07	0.02*	1.47	2.18	2.56*	2.09	2.12*
$w_{22}^{(h)}$	2.18	2.18*	3.08*	2.18	0.00*	0.00*	2.27	1.38*	2.18	2.18*	3.65	4.36	4.19*	3.98	3.91*
$w_{23}^{(h)}$	2.18	2.18*	3.08*	2.18	0.00*	0.00*	2.27	1.38*	2.18	2.18*	3.65	4.36	4.19*	3.98	3.91*
$w_{31}^{(h)}$	0.00	0.00*	2.17*	0.00*	2.18	2.18	2.56*	2.56	3.08	0.03*	1.47	2.18	2.58*	2.11	2.14*
$w_{32}^{(h)}$	2.56*	2.56	2.31	2.56	1.38*	1.38*	1.08	0.00*	0.81*	2.55*	3.88	4.53	3.66*	3.69	3.54*
$w_{33}^{(h)}$	3.08*	3.08	2.17	3.08	2.18*	2.18*	0.77	0.81	0.00*	3.06*	4.24	4.83	3.58*	3.77	3.58*
$w_{41}^{(h)}$	0.00	0.00*	2.17*	0.00*	2.18	2.18	2.56*	2.56	3.08	0.03*	1.47	2.18	2.58*	2.11	2.14*
$w_{42}^{(h)}$	1.47	1.47*	2.62*	1.47*	3.65	3.65	3.58*	3.88	4.24	1.47	0.00*	0.72	2.16*	1.46*	1.65*
$w_{43}^{(h)}$	2.18	2.18*	3.08*	2.18*	4.36	4.36	4.17*	4.56	4.87	2.18	0.72	0.08*	2.29	1.60*	1.84
$w_{51}^{(h)}$	2.58*	2.58	1.41	2.58*	4.19	4.19	2.81*	3.65	3.58	2.55	2.16*	2.22	0.00	0.71*	0.52
$w_{52}^{(h)}$	2.11*	2.11	1.68	2.11*	3.98	3.98	3.00*	3.69	3.77	2.09	1.46*	1.52	0.71	0.00*	0.25
$w_{53}^{(h)}$	2.57*	2.57	2.32	2.57*	4.57	4.57	3.66*	4.33	4.42	2.55	1.55	1.29*	1.10	0.66*	0.85

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.7: Matrix $\mathbf{E}$(W) representing the gap in initial velocity of the object after being hit by the tool W, between the experienced displacement $w_{ij}^{(h)}$ (given during the evaluation process) and actual displacement $w_{km}^{(r)}$ (result of using action A_k and functional feature F_m, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$). The gap calculated using Equation 4.7 is represented by greyscale parametric shading, i.e., the wider the gap, the darker the background. For a gap corresponding to the actually inferred k shown in Figure 4.5, the '*' mark is used; hence, a smaller gap is expected for a better inference. $w_{53}^{(r)} \in \mathbb{R}^2$ was not used during the learning process, but $w_{53}^{(h)} \in \mathbb{R}^2$ was given during the evaluation process as *novel effect*, as shown in the last row.

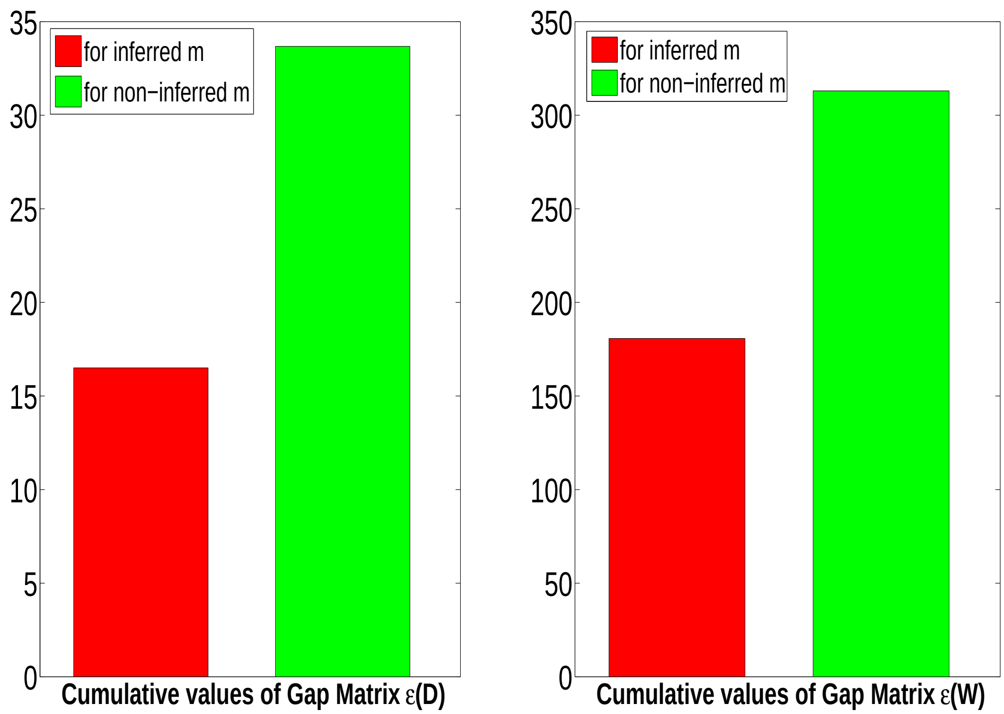


Figure 4.8: Bars representing the gap between actual effect by the robot learner and desired effect given by a human user. The gap is calculated using Equation 4.4 for the values shown in Figure 4.6 (**Left Graph**) and Equation 4.5 for the values shown in Figure 4.7 (**Right Graph**).

4.3 Experiment 2: Manipulation of target object when a randomly varying linear force is applied on the tool.

For each pair of A_i, F_j shown in Table 4.1 a total of 300 manipulation trials are performed. For each such trial, different magnitude of tool velocity in the direction corresponding to action A_i is observed due to the different magnitude of force applied on the tool. The observation data obtained using one manipulation trial is considered as one sample for that particular pair of (A_i, F_j).

In total 240 samples obtained for each pair (A_i, F_j). Since $i \in [1, 5]$, $j \in [1, 3]$, there are $5 \cdot 3 = 15$ combination of *Action* and *functional features* used to manipulate the object and create effect. Out of the 240 samples, 180 are the total samples kept aside for learning the affordances for each pair. The data for 60 instances for each of 15 pairs is kept for evaluating the tool affordances. However, the observation data from all the samples except (A_5, F_3) was used in the learning process. The reason, I do not use the data from (A_5, F_3) for learning is that, I am interested in evaluating the capability of learned affordances for the case when the observations are novel or unseen to the robot i.e. when no similar sensori-motor pattern has previously been learned. Thus the evaluation sample from (A_5, F_3) is mentioned as *novel* or *unseen* effect. Thus, in total, the data from 14 pairs is used for learning the tool affordances and the data from all 15 pair are used for the evaluation i.e. making the inferences. The descriptive statistics for the data which is used for learning the tool affordances is given in Section 4.3.1 while that of evaluation is given in Section 4.3.2.

The following notation is used for describe the data in the tables below:

- v_{i_x} denotes the component of the velocity of the tool used for action A_i along the x-axis.

- v_{i_z} denotes the component of the velocity of the tool used for action A_i along the x-axis.

- d_{ij_x} denotes the x component of final target displacement caused due to the manipulation of target object while using action A_i with functional feature F_j.

- d_{ij_z} denotes the z component of final target displacement caused due to the manipulation of target object while using action A_i with functional feature F_j.

- w_{ij_x} denotes the x component of target velocity (right after impact with the tool) caused due to the manipulation of target object while using action A_i with functional feature F_j.

- w_{ij_z} denotes the z component of target velocity (right after impact with the tool) caused due to the manipulation of target object while using action A_i with functional feature F_j.

4.3.1 Data for learning the tool affordances

Table 4.3: The observations of the target manipulation performed using A_1 and functional feature F_1

	$v1_x$	$v1_z$	$d11_x$	$d11_z$	$w11_x$	$w11_z$
min	0.00	1.30	0.00	0.17	-0.01	1.88
1st Quantile	0.00	1.54	0.00	0.24	-0.01	2.22
Median	0.00	1.73	0.00	0.31	-0.01	2.50
Mean	0.00	1.73	0.00	0.31	-0.01	2.51
Standard Deviation	0.00	0.23	0.00	0.08	0.00	0.33
3rd Quantile	0.00	1.93	0.00	0.39	-0.01	2.81
Max	0.00	2.11	0.00	0.47	-0.01	3.07

Table 4.4: The observations of the target manipulation performed using A_1 and functional feature F_2

	$v1_x$	$v1_z$	$d12_x$	$d12_z$	$w12_x$	$w12_z$
min	0.00	1.53	0.00	0.00	0.00	0.00
1st Quantile	0.00	1.71	0.00	0.00	0.00	0.00
Median	0.00	1.94	0.00	0.00	0.00	0.00
Mean	0.00	1.93	0.00	0.00	0.00	0.00
Standard Deviation	0.00	0.24	0.00	0.00	0.00	0.00
3rd Quantile	0.00	2.13	0.00	0.00	0.00	0.00
Max	0.00	2.33	0.00	0.00	0.00	0.00

Table 4.5: The observations of the target manipulation performed using A_1 and functional feature F_3

	$v1_x$	$v1_z$	$d13_x$	$d13_z$	$w13_x$	$w13_z$
min	0.00	1.53	0.00	0.19	-0.01	2.17
1st Quantile	0.00	1.74	0.00	0.26	-0.01	2.48
Median	0.00	1.94	0.00	0.34	-0.01	2.79
Mean	0.00	1.94	0.00	0.34	-0.01	2.77
Standard Deviation	0.00	0.23	0.00	0.09	0.00	0.35
3rd Quantile	0.00	2.15	0.00	0.42	-0.01	3.09
Max	0.00	2.33	0.00	0.49	-0.01	3.35

Table 4.6: The observations of the target manipulation performed using A_2 and functional feature F_1

	$v2_x$	$v2_z$	$d21_x$	$d21_z$	$w21_x$	$w21_z$
min	-2.33	0.00	0.00	0.00	0.00	0.00
1st Quantile	-2.13	0.00	0.00	0.00	0.00	0.00
Median	-1.93	0.00	0.00	0.00	0.00	0.00
Mean	-1.92	0.00	0.00	0.00	0.00	0.00
Standard Deviation	0.23	0.00	0.00	0.00	0.00	0.00
3rd Quantile	-1.71	0.00	0.00	0.00	0.00	0.00
Max	-1.53	0.00	0.00	0.00	0.00	0.00

Table 4.7: The observations of the target manipulation performed using A_2 and functional feature F_2

	$v2_x$	$v2_z$	$d22_x$	$d22_z$	$w22_x$	$w22_z$
min	-2.33	0.00	-0.50	0.00	-3.35	-0.00
1st Quantile	-2.13	0.00	-0.41	0.00	-3.06	-0.00
Median	-1.93	0.00	-0.33	0.00	-2.76	-0.00
Mean	-1.93	0.00	-0.33	0.00	-2.76	-0.00
Standard Deviation	0.24	0.00	0.09	0.00	0.35	0.00
3rd Quantile	-1.71	0.00	-0.25	0.00	-2.43	-0.00
Max	-1.53	0.00	-0.19	0.00	-2.17	-0.00

Table 4.8: The observations of the target manipulation performed using A_2 and functional feature F_3

	$v2_x$	$v2_z$	$d23_x$	$d23_z$	$w23_x$	$w23_z$
min	-2.34	0.00	-0.51	0.00	-3.37	-0.00
1st Quantile	-2.17	0.00	-0.43	0.00	-3.12	-0.00
Median	-1.89	0.00	-0.32	0.00	-2.70	-0.00
Mean	-1.93	0.00	-0.34	0.00	-2.76	-0.00
Standard Deviation	0.24	0.00	0.10	0.00	0.36	0.00
3rd Quantile	-1.71	0.00	-0.25	0.00	-2.44	-0.00
Max	-1.54	0.00	-0.20	0.00	-2.19	-0.00

Table 4.9: The observations of the target manipulation performed using A_3 and functional feature F_1

	$v3_x$	$v3_z$	$d31_x$	$d31_z$	$w31_x$	$w31_z$
min	-2.34	1.53	-1.08	0.26	-2.20	2.15
1st Quantile	-2.12	1.77	-0.47	0.41	-1.99	2.51
Median	-1.93	1.93	-0.37	0.56	-1.80	2.75
Mean	-1.93	1.93	-0.38	0.58	-1.80	2.75
Standard Deviation	0.23	0.23	0.17	0.22	0.22	0.33
3rd Quantile	-1.77	2.12	-0.25	0.69	-1.64	3.02
Max	-1.53	2.34	-0.14	1.46	-1.40	3.34

Table 4.10: The observations of the target manipulation performed using A_3 and functional feature F_2

	$v3_x$	$v3_z$	$d32_x$	$d32_z$	$w32_x$	$w32_z$
min	-2.33	1.53	-1.40	0.18	-3.34	1.41
1st Quantile	-2.12	1.71	-0.79	0.27	-3.03	1.60
Median	-1.94	1.94	-0.61	0.38	-2.76	1.82
Mean	-1.93	1.93	-0.61	0.40	-2.75	1.81
Standard Deviation	0.24	0.24	0.21	0.16	0.35	0.23
3rd Quantile	-1.71	2.12	-0.42	0.50	-2.43	2.00
Max	-1.53	2.33	-0.27	0.96	-2.15	2.21

Table 4.11: The observations of the target manipulation performed using A_3 and functional feature F_3

	$v3_x$	$v3_z$	$d33_x$	$d33_z$	$w33_x$	$w33_z$
min	-2.33	1.53	-1.68	0.18	-3.16	2.17
1st Quantile	-2.15	1.78	-0.90	0.48	-2.91	2.48
Median	-1.95	1.95	-0.69	0.64	-2.62	2.63
Mean	-1.96	1.96	-0.75	0.65	-2.61	2.67
Standard Deviation	0.22	0.22	0.26	0.23	0.33	0.26
3rd Quantile	-1.78	2.15	-0.55	0.77	-2.33	2.92
Max	-1.53	2.33	-0.34	1.55	-2.08	3.17

Table 4.12: The observations of the target manipulation performed using A_4 and functional feature F_1

	$v4_x$	$v4_z$	$d41_x$	$d41_z$	$w41_x$	$w41_z$
min	1.53	0.00	0.00	0.00	0.00	0.00
1st Quantile	1.72	0.00	0.00	0.00	0.00	0.00
Median	1.91	0.00	0.00	0.00	0.00	0.00
Mean	1.92	0.00	0.00	0.00	0.00	0.00
Standard Deviation	0.24	0.00	0.00	0.00	0.00	0.00
3rd Quantile	2.13	0.00	0.00	0.00	0.00	0.00
Max	2.34	0.00	0.00	0.00	0.00	0.00

Table 4.13: The observations of the target manipulation performed using A_4 and functional feature F_2

	$v4_x$	$v4_z$	$d42_x$	$d42_z$	$w42_x$	$w42_z$
min	1.53	0.00	0.19	0.00	2.18	0.00
1st Quantile	1.74	0.00	0.26	0.00	2.48	0.00
Median	1.90	0.00	0.31	0.00	2.72	0.00
Mean	1.92	0.00	0.32	0.00	2.74	0.00
Standard Deviation	0.23	0.00	0.09	0.00	0.34	0.00
3rd Quantile	2.12	0.00	0.40	0.00	3.04	0.00
Max	2.33	0.00	0.49	0.00	3.35	0.00

Table 4.14: The observations of the target manipulation performed using A_4 and functional feature F_3

	$v4_x$	$v4_z$	$d43_x$	$d43_z$	$w43_x$	$w43_z$
min	1.53	0.00	0.19	0.00	2.17	0.00
1st Quantile	1.75	0.00	0.26	0.00	2.49	0.00
Median	1.93	0.00	0.33	0.00	2.76	0.00
Mean	1.93	0.00	0.33	0.00	2.76	0.00
Standard Deviation	0.22	0.00	0.09	0.00	0.33	0.00
3rd Quantile	2.11	0.00	0.41	0.00	3.03	0.00
Max	2.33	0.00	0.50	0.00	3.35	0.00

Table 4.15: The observations of the target manipulation performed using A_5 and functional feature F_1

	$v5_x$	$v5_z$	$d51_x$	$d51_z$	$w51_x$	$w51_z$
min	1.53	1.53	0.10	0.28	1.40	2.14
1st Quantile	1.73	1.73	0.17	0.36	1.60	2.43
Median	1.93	1.93	0.29	0.48	1.79	2.72
Mean	1.93	1.93	0.31	0.52	1.79	2.72
Standard Deviation	0.24	0.24	0.16	0.19	0.23	0.35
3rd Quantile	2.12	2.12	0.41	0.66	1.98	3.01
Max	2.34	2.34	0.96	1.15	2.19	3.32

Table 4.16: The observations of the target manipulation performed using A_5 and functional feature F_2

	$v5_x$	$v5_z$	$d52_x$	$d52_z$	$w52_x$	$w52_z$
min	1.53	1.53	0.27	0.10	2.16	1.41
1st Quantile	1.73	1.73	0.35	0.24	2.46	1.61
Median	1.95	1.95	0.51	0.31	2.78	1.83
Mean	1.93	1.93	0.54	0.33	2.76	1.82
Standard Deviation	0.24	0.24	0.20	0.15	0.36	0.24
3rd Quantile	2.16	2.16	0.71	0.43	3.11	2.04
Max	2.34	2.34	1.02	0.85	3.36	2.21

Table 4.17: The observations of the target manipulation performed using A_5 and functional feature F_3

	$v5_x$	$v5_z$	$d53_x$	$d53_z$	$w53_x$	$w53_z$
min	1.53	1.53	0.33	0.30	2.16	2.16
1st Quantile	1.68	1.68	0.46	0.40	2.39	2.39
Median	1.87	1.87	0.63	0.57	2.60	2.61
Mean	1.90	1.90	0.75	0.61	2.65	2.66
Standard Deviation	0.24	0.24	0.38	0.26	0.32	0.32
3rd Quantile	2.10	2.10	0.98	0.76	2.94	2.94
Max	2.34	2.34	2.00	1.54	3.28	3.28

4.3.2 Data for evaluation of learnt tool affordances

In this sub-section, the descriptive statistics of the data used for making inferences has been provided. The data used for calculating the inferences listed Table 3.4 is provided by the human to the robot. In total 60 instances of the observations of target manipulation, resulted from each pair of action A_i and F_j is recorded, where $i \in [1, 5]$, $j \in [1, 3]$. Thus, similar to the situation when data for learning the affordances is observed and recorded, to evaluate the inferences, robot uses $5 \cdot 3 = 15$ combination of *Action* and *functional features* to manipulate the object and create effect, used to evaluate its learning.

Table 4.18: The observations of the target manipulation performed using A_1 and functional feature F_1

	$v1_x$	$v1_z$	$d11_x$	$d11_z$	$w11_x$	$w11_z$
min	0.00	1.30	0.00	0.17	-0.01	1.87
1st Quantile	0.00	1.46	0.00	0.22	-0.01	2.11
Median	0.00	1.62	0.00	0.27	-0.01	2.34
Mean	0.00	1.66	0.00	0.29	-0.01	2.40
Standard Deviation	0.00	0.25	0.00	0.09	0.00	0.37
3rd Quantile	0.00	1.89	0.00	0.37	-0.01	2.74
Max	0.00	2.10	0.00	0.46	-0.01	3.06

Table 4.19: The observations of the target manipulation performed using A_1 and functional feature F_2

	$v1_x$	$v1_z$	$d12_x$	$d12_z$	$w12_x$	$w12_z$
min	0.00	1.54	0.00	0.00	0.00	0.00
1st Quantile	0.00	1.75	0.00	0.00	0.00	0.00
Median	0.00	1.94	0.00	0.00	0.00	0.00
Mean	0.00	1.95	0.00	0.00	0.00	0.00
Standard Deviation	0.00	0.23	0.00	0.00	0.00	0.00
3rd Quantile	0.00	2.13	0.00	0.00	0.00	0.00
Max	0.00	2.34	0.00	0.00	0.00	0.00

Table 4.20: The observations of the target manipulation performed using A_1 and functional feature F_3

	$v1_x$	$v1_z$	$d13_x$	$d13_z$	$w13_x$	$w13_z$
min	0.00	1.54	0.00	0.20	-0.01	2.18
1st Quantile	0.00	1.71	0.00	0.25	-0.01	2.44
Median	0.00	1.82	0.00	0.29	-0.01	2.61
Mean	0.00	1.89	0.00	0.32	-0.01	2.71
Standard Deviation	0.00	0.23	0.00	0.09	0.00	0.34
3rd Quantile	0.00	2.07	0.00	0.39	-0.01	2.98
Max	0.00	2.32	0.00	0.49	-0.01	3.34

Table 4.21: The observations of the target manipulation performed using A_2 and functional feature F_1

	$v2_x$	$v2_z$	$d21_x$	$d21_z$	$w21_x$	$w21_z$
min	-2.34	0.00	0.00	0.00	0.00	0.00
1st Quantile	-2.10	0.00	0.00	0.00	0.00	0.00
Median	-1.91	0.00	0.00	0.00	0.00	0.00
Mean	-1.92	0.00	0.00	0.00	0.00	0.00
Standard Deviation	0.23	0.00	0.00	0.00	0.00	0.00
3rd Quantile	-1.71	0.00	0.00	0.00	0.00	0.00
Max	-1.53	0.00	0.00	0.00	0.00	0.00

Table 4.22: The observations of the target manipulation performed using A_2 and functional feature F_2

	$v2_x$	$v2_z$	$d22_x$	$d22_z$	$w22_x$	$w22_z$
min	-2.34	0.00	-0.50	0.00	-3.37	-0.00
1st Quantile	-2.17	0.00	-0.43	0.00	-3.12	-0.00
Median	-1.97	0.00	-0.35	0.00	-2.83	-0.00
Mean	-1.97	0.00	-0.35	0.00	-2.82	-0.00
Standard Deviation	0.24	0.00	0.09	0.00	0.35	0.00
3rd Quantile	-1.79	0.00	-0.28	0.00	-2.55	-0.00
Max	-1.56	0.00	-0.20	0.00	-2.22	-0.00

Table 4.23: The observations of the target manipulation performed using A_2 and functional feature F_3

	$v2_x$	$v2_z$	$d23_x$	$d23_z$	$w23_x$	$w23_z$
min	-2.33	0.00	-0.51	0.00	-3.36	-0.00
1st Quantile	-2.19	0.00	-0.44	0.00	-3.15	-0.00
Median	-2.04	0.00	-0.37	0.00	-2.92	-0.00
Mean	-1.97	0.00	-0.35	0.00	-2.82	-0.00
Standard Deviation	0.26	0.00	0.10	0.00	0.38	0.00
3rd Quantile	-1.75	0.00	-0.27	0.00	-2.50	-0.00
Max	-1.55	0.00	-0.20	0.00	-2.20	-0.00

Table 4.24: The observations of the target manipulation performed using A_3 and functional feature F_1

	$v3_x$	$v3_z$	$d31_x$	$d31_z$	$w31_x$	$w31_z$
min	-2.33	1.53	-1.22	0.26	-2.19	2.15
1st Quantile	-2.14	1.78	-0.56	0.42	-2.01	2.52
Median	-2.01	2.01	-0.42	0.62	-1.88	2.86
Mean	-1.97	1.97	-0.44	0.61	-1.83	2.79
Standard Deviation	0.23	0.23	0.23	0.22	0.22	0.34
3rd Quantile	-1.78	2.14	-0.25	0.75	-1.65	3.05
Max	-1.53	2.33	-0.14	1.19	-1.41	3.33

Table 4.25: The observations of the target manipulation performed using A_3 and functional feature F_2

	$v3_x$	$v3_z$	$d32_x$	$d32_z$	$w32_x$	$w32_z$
min	-2.33	1.56	-1.18	0.18	-3.33	1.45
1st Quantile	-2.18	1.77	-0.84	0.28	-3.12	1.66
Median	-2.01	2.01	-0.66	0.40	-2.87	1.89
Mean	-1.98	1.98	-0.66	0.44	-2.82	1.86
Standard Deviation	0.23	0.23	0.23	0.19	0.35	0.23
3rd Quantile	-1.77	2.18	-0.49	0.55	-2.52	2.06
Max	-1.56	2.33	-0.28	1.02	-2.21	2.20

Table 4.26: The observations of the target manipulation performed using A_3 and functional feature F_3

	$v3_x$	$v3_z$	$d33_x$	$d33_z$	$w33_x$	$w33_z$
min	-2.32	1.53	-1.48	0.32	-3.13	2.17
1st Quantile	-2.10	1.72	-0.82	0.44	-2.82	2.45
Median	-1.89	1.89	-0.63	0.60	-2.58	2.59
Mean	-1.91	1.91	-0.68	0.63	-2.58	2.63
Standard Deviation	0.23	0.23	0.23	0.25	0.31	0.26
3rd Quantile	-1.72	2.10	-0.52	0.75	-2.30	2.83
Max	-1.53	2.32	-0.34	1.45	-2.09	3.14

Table 4.27: The observations of the target manipulation performed using A_4 and functional feature F_1

	$v4_x$	$v4_z$	$d41_x$	$d41_z$	$w41_x$	$w41_z$
min	1.54	0.00	0.00	0.00	0.00	0.00
1st Quantile	1.70	0.00	0.00	0.00	0.00	0.00
Median	1.91	0.00	0.00	0.00	0.00	0.00
Mean	1.92	0.00	0.00	0.00	0.00	0.00
Standard Deviation	0.23	0.00	0.00	0.00	0.00	0.00
3rd Quantile	2.11	0.00	0.00	0.00	0.00	0.00
Max	2.33	0.00	0.00	0.00	0.00	0.00

Table 4.28: The observations of the target manipulation performed using A_4 and functional feature F_2

	$v4_x$	$v4_z$	$d42_x$	$d42_z$	$w42_x$	$w42_z$
min	1.53	0.00	0.19	0.00	2.18	0.00
1st Quantile	1.70	0.00	0.24	0.00	2.43	0.00
Median	1.90	0.00	0.31	0.00	2.72	0.00
Mean	1.91	0.00	0.32	0.00	2.74	0.00
Standard Deviation	0.23	0.00	0.09	0.00	0.34	0.00
3rd Quantile	2.09	0.00	0.39	0.00	3.00	0.00
Max	2.32	0.00	0.49	0.00	3.34	0.00

Table 4.29: The observations of the target manipulation performed using A_4 and functional feature F_3

	$v4_x$	$v4_z$	$d43_x$	$d43_z$	$w43_x$	$w43_z$
min	1.55	0.00	0.20	0.00	2.20	0.00
1st Quantile	1.71	0.00	0.25	0.00	2.44	0.00
Median	1.86	0.00	0.30	0.00	2.66	0.00
Mean	1.90	0.00	0.32	0.00	2.71	0.00
Standard Deviation	0.22	0.00	0.08	0.00	0.32	0.00
3rd Quantile	2.09	0.00	0.40	0.00	3.01	0.00
Max	2.33	0.00	0.51	0.00	3.36	0.00

Table 4.30: The observations of the target manipulation performed using A_5 and functional feature F_1

	$v5_x$	$v5_z$	$d51_x$	$d51_z$	$w51_x$	$w51_z$
min	1.53	1.53	0.10	0.28	1.40	2.14
1st Quantile	1.71	1.71	0.17	0.34	1.58	2.40
Median	1.96	1.96	0.32	0.54	1.82	2.77
Mean	1.94	1.94	0.33	0.53	1.81	2.74
Standard Deviation	0.25	0.25	0.18	0.20	0.25	0.37
3rd Quantile	2.18	2.18	0.42	0.67	2.04	3.10
Max	2.32	2.32	0.81	1.03	2.17	3.29

Table 4.31: The observations of the target manipulation performed using A_5 and functional feature F_2

	$v5_x$	$v5_z$	$d52_x$	$d52_z$	$w52_x$	$w52_z$
min	1.53	1.53	0.27	0.10	2.16	1.41
1st Quantile	1.65	1.65	0.32	0.19	2.34	1.54
Median	1.91	1.91	0.47	0.28	2.73	1.79
Mean	1.90	1.90	0.53	0.32	2.71	1.78
Standard Deviation	0.24	0.24	0.22	0.16	0.36	0.24
3rd Quantile	2.12	2.12	0.72	0.43	3.04	2.00
Max	2.32	2.32	1.31	0.83	3.33	2.20

Table 4.32: The observations of the target manipulation performed using A_5 and functional feature F_3

	$v5_x$	$v5_z$	$d53_x$	$d53_z$	$w53_x$	$w53_z$
min	1.53	1.53	0.33	0.30	2.16	2.16
1st Quantile	1.75	1.75	0.51	0.43	2.43	2.49
Median	1.96	1.96	0.75	0.59	2.73	2.73
Mean	1.93	1.93	0.84	0.64	2.73	2.73
Standard Deviation	0.23	0.23	0.40	0.26	0.33	0.33
3rd Quantile	2.12	2.12	1.18	0.84	3.03	3.03
Max	2.30	2.30	1.84	1.47	3.27	3.27

4.3.3 Results of Experiment 2 and their Discussion

The *Tool Affordances* framework proposed in Figure 3.1 is implemented using the *Bayesian Network*(BN). After robot learns the tool affordances, I evaluate robot's capability to emulate desired effects and test the inference capabilities mentioned in Table 3.4. The effects created for the specific purpose of evaluation are given by the human user to the robot. Hence, the data corresponding to that purpose is denoted using superscript (h), while the data used for learning is denoted using superscript (r).

Thus the 60 samples of data obtained for the manipulation of the target object using all 15 combination of actions and functional features i.e. using each pair of (A_i, F_j) $(i \in [1,5],\ j \in [1,3])$ is given to the robot by human user. The effects resulted from target manipulation are observed, recorded and used to evaluate tool affordadances learnt by the robot. Robot is presented with the evaluation effects $(d_{ij}^{(h)}, w_{ij}^{(h)})$. The task of the robot is to emulate given effects by suitable estimation of inference target for given inputs (Table 4.2).

To validate the inference result and calculate its accuracy, robot calculates the gap in effects given during evaluation process i.e. $(d_{ij}^{(h)}, w_{ij}^{(h)})$ and actual effects $(d_{km}^{(r)}, w_{km}^{(r)})$ resulted from the manipulation of action A_k and functional feature F_m by the robot(denoted by superscript (r)). The gap between these desired and actual effects in $D \in \mathbb{R}^2$ and $W \in \mathbb{R}^2$ is defined as follows:

$$\epsilon(d_{ij}^{(r)}) \stackrel{def}{=} d_{km}^{(r)} - d_{ij}^{(h)} \tag{4.6}$$

$$\epsilon(w_{ij}^{(r)}) \stackrel{def}{=} w_{km}^{(r)} - w_{ij}^{(h)} \tag{4.7}$$

4.3.3.1 Inference 1 : Action Recognition and Selection.

The probability of robot estimation of suitable action A_k to realize given effects by human demonstrator $d_{ij}^{(h)} \in \mathbb{R}^2$ and $w_{ij}^{(h)} \in \mathbb{R}^2$ using the functional feature F_m, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$ shown in Fig 4.9, is calculated using the following equation.

$$P_A = P(A = A_k | F_{(m)}, d_{ij}^{(h)}, w_{ij}^{(h)}) \tag{4.8}$$

$$\epsilon(d_{ij}^{(r)}) = \frac{\displaystyle\sum_{n_e=1}^{60}\Big(\sum_{i=1}^{5}\sum_{j=1}^{3}\Big[\sum_{m=1}^{3}\Big(\sum_{k}(d_{km}^{(r)} - d_{ij}^{(h)})\Big)\Big]\Big)}{\displaystyle\sum_{n_e=1}^{60}\Big[\sum_{i=1}^{5}\sum_{j=1}^{3}\Big(\sum_{m=1}^{3}\sum_{i=1}^{5}(\delta_{i\,k})\Big)\Big]} \tag{4.9}$$

$$\epsilon(w_{ij}^{(r)}) = \frac{\displaystyle\sum_{n_e=1}^{60}\Big(\sum_{i=1}^{5}\sum_{j=1}^{3}\Big[\sum_{m=1}^{3}\Big(\sum_{k}(w_{km}^{(r)} - w_{ij}^{(h)})\Big)\Big]\Big)}{\displaystyle\sum_{n_e=1}^{60}\Big[\sum_{i=1}^{5}\sum_{j=1}^{3}\Big(\sum_{m=1}^{3}\sum_{i=1}^{5}(\delta_{i\,k})\Big)\Big]} \tag{4.10}$$

It can be noticed that that learned tool Affordances enable robot to emulate novel effects novel evaluation effects $d_{53}^{(h)}$ and $w_{53}^{(h)}$. But to validate the inference result, I present matrix $\mathbf{E}$(D) and $\mathbf{E}$(W) as shown in Figure 4.10 and Figure 4.11 respectively with greyscale shading corresponding to the gap values of $\epsilon(d_{ij}^{(r)})$ and $\epsilon(w_{ij}^{(r)})$ calculated using Equation 4.9 and Equation 4.10 respectively.

As could be expected, the gap values of diagonal elements of both matrices are lowest. Analysis of gap values to realize any pair of effect $d_{ij}^{(h)}, w_{ij}^{(h)}$ where using arbitrarily given functional feature F_m and the inferred action A_k (shown using '*') are much lower than the gap values corresponding to non-inferred action A_k. Thus,

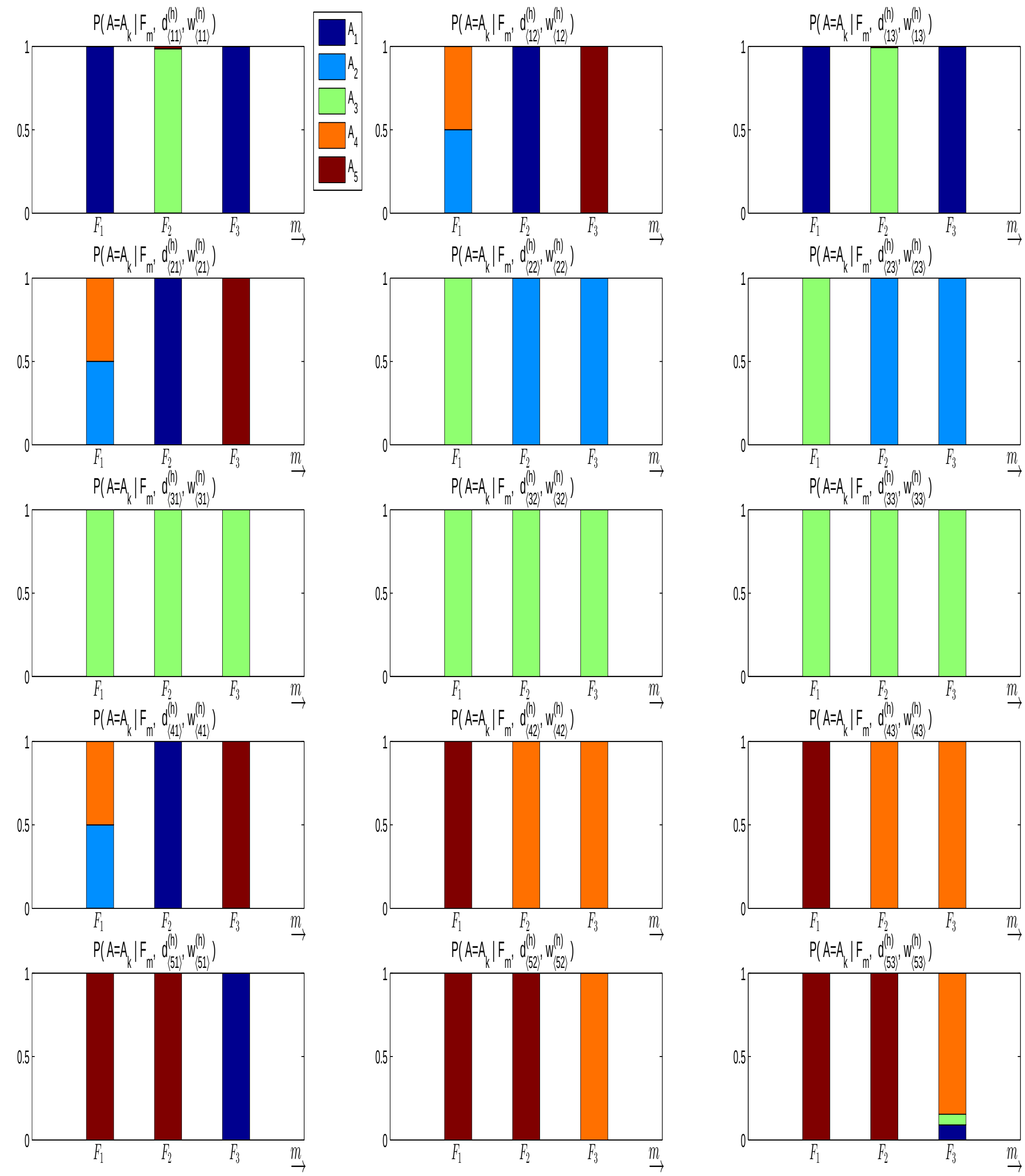


Figure 4.9: Robot estimation of suitable Action A_k to realize given effects by human demonstrator $(d_{ij}^{(h)}, w_{ij}^{(h)})$ (measured from the manipulation of Action A_i and functional feature F_j) using the arbitrarily given functional feature F_m ,where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$

inferred A_k for any arbitrarily given F_m is most suitable tool affordances to emulate both learned and novel desired effects. To further support my argument, I create

The desired effects given by the human for the robot to realize

$d^{(h)}_{ij}$ \ $d^{(r)}_{km}$	$d^{(r)}_{11}$	$d^{(r)}_{12}$	$d^{(r)}_{13}$	$d^{(r)}_{21}$	$d^{(r)}_{22}$	$d^{(r)}_{23}$	$d^{(r)}_{31}$	$d^{(r)}_{32}$	$d^{(r)}_{33}$	$d^{(r)}_{41}$	$d^{(r)}_{42}$	$d^{(r)}_{43}$	$d^{(r)}_{51}$	$d^{(r)}_{52}$	$d^{(r)}_{53}$
$d^{(h)}_{11}$	0.08*	0.34	0.08*	0.34	0.49	0.48	0.47	0.63*	0.81	0.34	0.46	0.47	0.39	0.53*	0.86
$d^{(h)}_{12}$	0.31	0.00*	0.35	0.00*	0.36	0.34	0.69	0.74	0.97	0.00*	0.32	0.33	0.62	0.62	1.02*
$d^{(h)}_{13}$	0.09*	0.35	0.09*	0.35	0.50	0.49	0.46	0.63*	0.80	0.35	0.47	0.48	0.38	0.54*	0.86
$d^{(h)}_{21}$	0.31	0.00*	0.35	0.00*	0.36	0.34	0.69	0.74	0.97	0.00*	0.32	0.33	0.62	0.62	1.02*
$d^{(h)}_{22}$	0.45	0.32	0.48	0.32	0.10*	0.09*	0.58*	0.52	0.77	0.32	0.63	0.65	0.83	0.90	1.28
$d^{(h)}_{23}$	0.45	0.32	0.48	0.32	0.10*	0.09*	0.58*	0.52	0.77	0.32	0.64	0.65	0.83	0.91	1.29
$d^{(h)}_{31}$	0.60	0.80	0.58	0.80	0.69	0.69	0.31*	0.42*	0.45*	0.80	1.01	1.02	0.81	1.04	1.26
$d^{(h)}_{32}$	0.66	0.76	0.65	0.76	0.53	0.54	0.38*	0.28*	0.39*	0.76	1.03	1.04	0.96	1.16	1.44
$d^{(h)}_{33}$	0.86	1.03	0.84	1.03	0.82	0.83	0.46*	0.42*	0.31*	1.03	1.27	1.28	1.10	1.33	1.55
$d^{(h)}_{41}$	0.31	0.00*	0.35	0.00*	0.36	0.34	0.69	0.74	0.97	0.00*	0.32	0.33	0.62	0.62	1.02*
$d^{(h)}_{42}$	0.47	0.35	0.50	0.35	0.71	0.68	0.93	1.05	1.25	0.35	0.10*	0.09*	0.54*	0.39	0.80
$d^{(h)}_{43}$	0.47	0.36	0.50	0.36	0.71	0.69	0.93	1.05	1.26	0.36	0.09*	0.09*	0.54*	0.38	0.79
$d^{(h)}_{51}$	0.47	0.67	0.45*	0.67	0.90	0.88	0.76	0.99	1.10	0.67	0.59	0.59	0.26*	0.39*	0.56
$d^{(h)}_{52}$	0.60	0.69	0.60	0.69	1.00	0.98	0.99	1.19	1.34	0.69	0.48	0.47*	0.38*	0.25*	0.47
$d^{(h)}_{53}$	0.97	1.12	0.96*	1.12	1.40	1.38	1.27	1.51	1.60	1.12	0.93	0.92*	0.66*	0.62*	0.45

The actual effects realized by the robot using inferred action A_k

Figure 4.10: Matrix $\mathbf{E}$(D) representing gap between desired displacement $d^{(h)}_{ij}$ (given during evaluation process) and experienced displacement $d^{(r)}_{km}$ of the target object (resulted using action A_k and functional feature F_m, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred k calculated using Equation 4.6 and shown in Figure 4.9, the '*' mark is used, hence a smaller gap value is expected. For the last column, $d^{(r)}_{53} \in \mathbb{R}^2$ was not used during the learning process, but $d^{(h)}_{53} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of the Figure 4.9).

errorbars (shown in Figure 4.12) using cumulative gap values representing inferred k and non-inferred k (averaged over all the evaluation effects). The errorbar is calculated using Equation 4.9 for the values shown in Figure 4.10 (shown as **Left Graph** in Figure 4.12) and Equation 4.10 for the values shown in Figure 4.11 (shown as **Right Graph** in Figure 4.12).

The desired effects given by the human for the robot to realize

$w^{(h)}_{ij}$ \ $w^{(r)}_{km}$	$w^{(r)}_{11}$	$w^{(r)}_{12}$	$w^{(r)}_{13}$	$w^{(r)}_{21}$	$w^{(r)}_{22}$	$w^{(r)}_{23}$	$w^{(r)}_{31}$	$w^{(r)}_{32}$	$w^{(r)}_{33}$	$w^{(r)}_{41}$	$w^{(r)}_{42}$	$w^{(r)}_{43}$	$w^{(r)}_{51}$	$w^{(r)}_{52}$	$w^{(r)}_{53}$
$w^{(h)}_{11}$	0.31*	2.59	0.40*	2.59	3.85	3.78	1.83	2.88*	2.59	2.59	3.76	3.78	1.86	2.88*	2.70
$w^{(h)}_{12}$	2.49	0.00*	2.82	0.00*	2.86	2.76	3.28	3.32	3.70	0.00*	2.72	2.75	3.31	3.29	3.79*
$w^{(h)}_{13}$	0.43*	2.79	0.34*	2.79	3.99	3.92	1.82	2.94*	2.60	2.79	3.90	3.92	1.86	2.94*	2.71
$w^{(h)}_{21}$	2.49	0.00*	2.82	0.00*	2.86	2.76	3.28	3.32	3.70	0.00*	2.72	2.75	3.31	3.29	3.79*
$w^{(h)}_{22}$	3.66	2.69	3.89	2.69	0.39*	0.35*	2.90*	1.86	2.68	2.69	5.40	5.43	5.28	5.73	5.99
$w^{(h)}_{23}$	3.66	2.69	3.89	2.69	0.38*	0.35*	2.90*	1.86	2.68	2.69	5.40	5.43	5.28	5.72	5.99
$w^{(h)}_{31}$	1.90	3.38	1.88	3.38	3.01	2.98	0.41*	1.41*	0.86*	3.38	5.37	5.39	3.68	4.72	4.54
$w^{(h)}_{32}$	2.84	3.30	2.94	3.30	1.85	1.85	1.39*	0.42*	0.96*	3.30	5.76	5.78	4.67	5.50	5.50
$w^{(h)}_{33}$	2.63	3.74	2.63	3.74	2.70	2.69	0.90*	0.95*	0.43*	3.74	5.96	5.98	4.44	5.43	5.29
$w^{(h)}_{41}$	2.49	0.00*	2.82	0.00*	2.86	2.76	3.28	3.32	3.70	0.00*	2.72	2.75	3.31	3.29	3.79*
$w^{(h)}_{42}$	3.77	2.82	3.99	2.82	5.66	5.57	5.36	5.87	6.00	2.82	0.37*	0.37*	2.95*	1.84	2.71
$w^{(h)}_{43}$	3.76	2.81	3.99	2.81	5.66	5.57	5.36	5.87	6.00	2.81	0.35*	0.35*	2.95*	1.84	2.71
$w^{(h)}_{51}$	1.86	3.28	1.85*	3.28	5.40	5.32	3.62	4.67	4.39	3.28	2.90	2.91	0.44*	1.39*	0.99
$w^{(h)}_{52}$	2.88	3.33	2.97	3.33	5.92	5.83	4.67	5.55	5.42	3.33	1.86	1.86*	1.40*	0.42*	0.97
$w^{(h)}_{53}$	2.74	3.84	2.74*	3.84	6.19	6.11	4.52	5.55	5.29*	3.84	2.74	2.74*	0.97*	1.00*	0.43

The actual effects realized by the robot using inferred action A_k

Figure 4.11: Matrix $\mathbf{E}$(W) representing gap between desired initial velocity $w^{(h)}_{ij}$ (given during evaluation process) and experienced initial velocity $w^{(r)}_{km}$ of the target object after its impact from tool, (resulted using action A_k and functional feature F_m, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred k calculated using Equation 4.7 and shown in Figure 4.9, the '*' mark is used, hence a smaller gap value is expected. For the last column, $w^{(r)}_{53} \in \mathbb{R}^2$ was not used during the learning process, but $w^{(h)}_{53} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of the Figure 4.9).

To test our hypothesis that the result of inference to realize the desired effects than the non-inferred one, we perform hypothesis testing by designing two groups. "Group One" corresponds to the case when inferred action was used and "Group Two" when "non-inferred" action was used. The table also includes gap values obtained for creating novel target displacement.

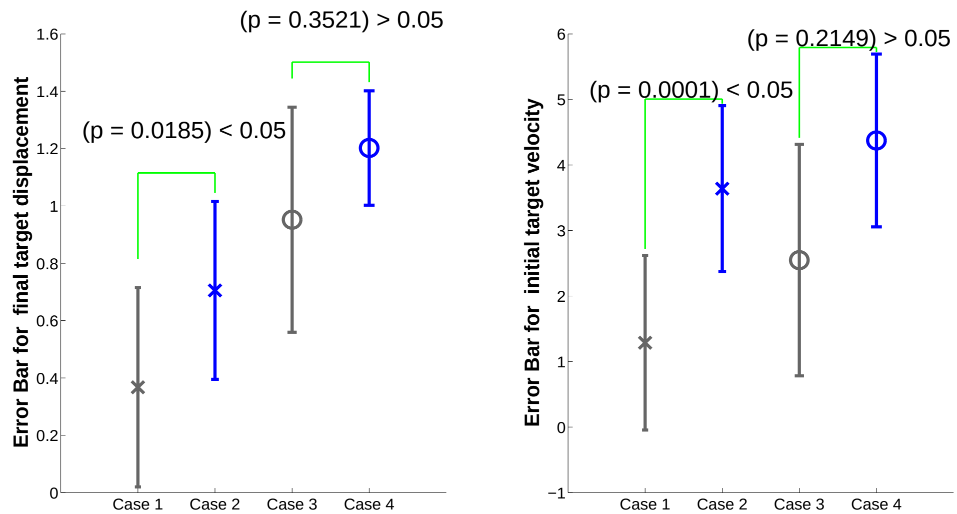


Figure 4.12: Comparison of error bars representing the gap between the effects as desired by human user with the ones experienced by the robot. The functional feature F_m is given to the robot as input to realize $d_{ij}^{(h)}$ (**Left Window**) and $w_{ij}^{(h)}$ (**Right Window**). **Case 1:** inferred A_k is used to create all the 15 types of effects. **Case 2:** non-inferred A_k is used to create all the 15 types of effects. **Case 3:** inferred A_k is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$. **Case 4:** non-inferred A_k is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$.

When the robot was asked to realized effects similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0185. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred A_k to realize the target displacement is statistically significant than the non-inferred A_k.

However, when the robot was asked to realize *only* novel displacement of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.352. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred A_k to realize the novel target displacement may not always yield better results than non-inferred A_k.

When the robot was asked to realize target velocity similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0001. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred A_k to realize the target velocity is statistically significant than the non-inferred A_k.

However, when the robot was asked to realize *only* novel velocity of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.2149. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred A_k to realize the novel target velocity may not always yield better results than non-inferred A_k.

4.3.3.2 Inference 2: Tool Recognition and Selection.

$$P_F = P(F = F_m | A_k, v_k^{(h)}, d_{ij}^{(h)}, w_{ij}^{(h)}) \tag{4.11}$$

The probability of robot estimation of suitable functional feature F_m to realize given effects $d_{ij}^{(h)} \in \mathbb{R}^2$ and $w_{ij}^{(h)} \in \mathbb{R}^2$ using the given Action A_k , where h and r denote human demonstrate and robot respectively, and $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$. The inference calculated using the following equation is shown in Fig 4.5.

To calculate the gap values for target displacement (refer Equation 4.6), the following equation is used.

$$\epsilon(d_{ij}^{(r)}) = \frac{\displaystyle\sum_{n_e=1}^{60}\Big(\sum_{i=1}^{5}\sum_{j=1}^{3}\Big[\sum_{k=1}^{5}\Big(\sum_{m}(d_{km}^{(r)} - d_{ij}^{(h)})\Big)\Big]\Big)}{\displaystyle\sum_{n_e=1}^{60}\Big[\sum_{i=1}^{5}\sum_{j=1}^{3}\Big(\sum_{k=1}^{5}\sum_{j=1}^{3}(\delta_{j\,m})\Big)\Big]} \tag{4.12}$$

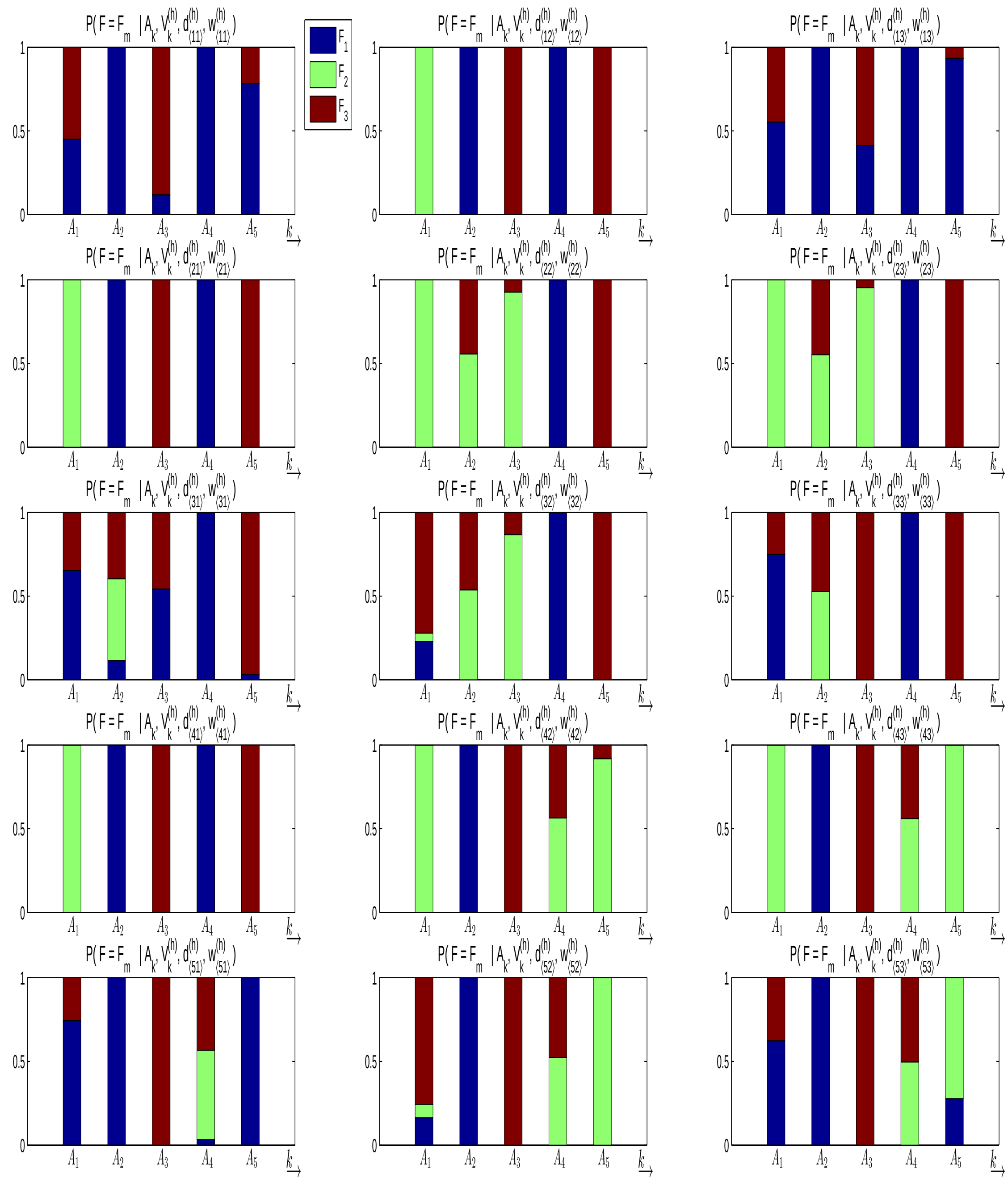


Figure 4.13: Robot estimation of suitable functional feature F_m for the arbitrarily given action A_k to realize given Effects $(d_{ij}^{(h)}, w_{ij}^{(h)})$ measured from the manipulation of Action A_i and functional feature F_j and given to the robot by human user, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$

To calculate the gap values for target velocity after the impact of tool (refer Equation 4.6), the following equation is used.

$$\epsilon(d_{ij}^{(r)}) = \frac{\sum_{n_e=1}^{60} \Big(\sum_{i=1}^{5} \sum_{j=1}^{3} \Big[\sum_{k=1}^{5} \Big(\sum_{m} (w_{km}^{(r)} - w_{ij}^{(h)}) \Big) \Big] \Big)}{60} \quad (4.13)$$

The desired effects given by the human for the robot to realize

$d_{ij}^{(h)}$ \ $d_{km}^{(r)}$	$d_{11}^{(r)}$	$d_{12}^{(r)}$	$d_{13}^{(r)}$	$d_{21}^{(r)}$	$d_{22}^{(r)}$	$d_{23}^{(r)}$	$d_{31}^{(r)}$	$d_{32}^{(r)}$	$d_{33}^{(r)}$	$d_{41}^{(r)}$	$d_{42}^{(r)}$	$d_{43}^{(r)}$	$d_{51}^{(r)}$	$d_{52}^{(r)}$	$d_{53}^{(r)}$
$d_{11}^{(h)}$	0.09*	0.33	0.09*	0.33*	0.49	0.48	0.46*	0.62	0.81*	0.33*	0.47	0.46	0.42*	0.54	0.79*
$d_{12}^{(h)}$	0.32	0.00*	0.35	0.00*	0.36	0.35	0.68	0.72	0.99*	0.00*	0.34	0.33	0.64	0.63	0.95*
$d_{13}^{(h)}$	0.09*	0.34	0.09*	0.34*	0.49	0.48	0.46*	0.62	0.81*	0.34*	0.48	0.47	0.41*	0.54	0.79*
$d_{21}^{(h)}$	0.32	0.00*	0.35	0.00*	0.36	0.35	0.68	0.72	0.99*	0.00*	0.34	0.33	0.64	0.63	0.95*
$d_{22}^{(h)}$	0.47	0.34*	0.49	0.34	0.09*	0.09*	0.57	0.49*	0.79*	0.34*	0.68	0.67	0.87	0.93	1.23*
$d_{23}^{(h)}$	0.48	0.36*	0.50	0.36	0.10*	0.10*	0.57	0.49*	0.79*	0.36*	0.69	0.68	0.88	0.94	1.23*
$d_{31}^{(h)}$	0.51*	0.71	0.49*	0.71*	0.60*	0.60*	0.26*	0.38	0.46*	0.71*	0.93	0.92	0.75*	0.97	1.14*
$d_{32}^{(h)}$	0.67*	0.77*	0.66*	0.77	0.55*	0.55*	0.40	0.30*	0.44*	0.77*	1.05	1.04	0.98	1.17	1.37*
$d_{33}^{(h)}$	0.85*	1.01	0.84*	1.01	0.80*	0.80*	0.47	0.42	0.32*	1.01*	1.27	1.26	1.10	1.33	1.48*
$d_{41}^{(h)}$	0.32	0.00*	0.35	0.00*	0.36	0.35	0.68	0.72	0.99*	0.00*	0.34	0.33	0.64	0.63	0.95*
$d_{42}^{(h)}$	0.45	0.32*	0.47	0.32*	0.67	0.66	0.89	1.00	1.24*	0.32	0.09*	0.08*	0.55	0.40*	0.74*
$d_{43}^{(h)}$	0.47	0.35*	0.49	0.35*	0.70	0.69	0.92	1.03	1.26*	0.35	0.09*	0.09*	0.55	0.39*	0.72*
$d_{51}^{(h)}$	0.46*	0.67	0.45*	0.67*	0.90	0.90	0.75	0.98	1.08*	0.67*	0.59*	0.59*	0.25*	0.39	0.48
$d_{52}^{(h)}$	0.57*	0.66*	0.57*	0.66*	0.97	0.96	0.96	1.16	1.31*	0.66	0.44*	0.44*	0.37	0.23*	0.42
$d_{53}^{(h)}$	1.02*	1.16	1.01*	1.16*	1.45	1.44	1.32	1.55	1.63	1.16	0.95*	0.95*	0.69*	0.64*	0.47

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.14: Matrix $\mathbf{E}$(D) representing gap between the desired displacement $d_{ij}^{(h)}$ (given during evaluation process) and experienced displacement $d_{km}^{(r)}$ of target object (resulted using action A_k and functional feature F_m, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred m calculated using Equation 4.6 and shown in Figure 4.13, the '*' mark is used, hence a smaller gap value is expected. For the last column, $d_{53}^{(r)} \in \mathbb{R}^2$ was not used during the learning process, but $d_{53}^{(h)} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of Figure 4.13.

To validate the inference result, I calculate matrix $\mathbf{E}$(D) and $\mathbf{E}$(W) using Equation 4.6 and Equation 4.7 respectively for $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$, similar to the ones shown in Section 4.3.3.1. The results show that gap values to realize any pair of effect $d_{ij}^{(h)}, w_{ij}^{(h)}$ where using arbitrarily given action A_k and inferred functional feature F_m are much lower than the gap values corresponding to non-inferred action F_m. Thus, I claim that inferred F_m for any arbitrarily given A_k is most suitable Tool

$w^{(h)}_{ij}$ \ $w^{(r)}_{km}$	$w^{(r)}_{11}$	$w^{(r)}_{12}$	$w^{(r)}_{13}$	$w^{(r)}_{21}$	$w^{(r)}_{22}$	$w^{(r)}_{23}$	$w^{(r)}_{31}$	$w^{(r)}_{32}$	$w^{(r)}_{33}$	$w^{(r)}_{41}$	$w^{(r)}_{42}$	$w^{(r)}_{43}$	$w^{(r)}_{51}$	$w^{(r)}_{52}$	$w^{(r)}_{53}$
$w^{(h)}_{11}$	0.35*	2.54	0.45*	2.54*	3.82	3.78	1.82*	2.84	2.60*	2.54*	3.79	3.73	1.88*	2.90	2.66*
$w^{(h)}_{12}$	2.53	0.00*	2.80	0.00*	2.85	2.80	3.24	3.28	3.71*	0.00*	2.81	2.73	3.32	3.33	3.73*
$w^{(h)}_{13}$	0.39*	2.74	0.35*	2.74*	3.95	3.91	1.81*	2.90	2.60*	2.74*	3.93	3.87	1.87*	2.95	2.66*
$w^{(h)}_{21}$	2.53	0.00*	2.80	0.00*	2.85	2.80	3.24	3.28	3.71*	0.00*	2.81	2.73	3.32	3.33	3.73*
$w^{(h)}_{22}$	3.75	2.78*	3.94	2.78	0.35*	0.34*	2.90	1.84*	2.68*	2.78*	5.58	5.49	5.36	5.85	6.01*
$w^{(h)}_{23}$	3.76	2.80*	3.96	2.80	0.40*	0.39*	2.91	1.85*	2.69*	2.80*	5.59	5.51	5.38	5.86	6.03*
$w^{(h)}_{31}$	1.81*	3.25	1.81*	3.25*	2.93*	2.91*	0.39*	1.37	0.90*	3.25*	5.33	5.26	3.62*	4.65	4.42*
$w^{(h)}_{32}$	2.83*	3.27*	2.92*	3.27	1.85*	1.84*	1.39	0.46*	0.99*	3.27*	5.81	5.73	4.66	5.51	5.42*
$w^{(h)}_{33}$	2.60*	3.71	2.60*	3.71	2.70*	2.70*	0.90	0.96	0.43*	3.71*	6.01	5.93	4.41	5.43	5.21*
$w^{(h)}_{41}$	2.53	0.00*	2.80	0.00*	2.85	2.80	3.24	3.28	3.71*	0.00*	2.81	2.73	3.32	3.33	3.73*
$w^{(h)}_{42}$	3.70	2.70*	3.90	2.70*	5.55	5.50	5.23	5.72	5.91*	2.70	0.34*	0.32*	2.92	1.86*	2.66*
$w^{(h)}_{43}$	3.77	2.80*	3.97	2.80*	5.64	5.59	5.31	5.81	5.99*	2.80	0.34*	0.34*	2.95	1.86*	2.67*
$w^{(h)}_{51}$	1.87*	3.31	1.86*	3.31*	5.43	5.38	3.61	4.66	4.41*	3.31*	2.95*	2.92*	0.39*	1.39	0.92
$w^{(h)}_{52}$	2.86*	3.28*	2.94*	3.28*	5.87	5.82	4.62	5.48	5.39*	3.28	1.83*	1.83*	1.39	0.39*	0.94
$w^{(h)}_{53}$	2.78*	3.89	2.77*	3.89*	6.23	6.19	4.53	5.57	5.33*	3.89	2.77*	2.77*	1.01*	1.01*	0.47

The desired effects given by the human for the robot to realize

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.15: Matrix $\mathbf{E}$(W) representing gap between desired initial velocity $w^{(h)}_{ij}$ (given during evaluation process) and experienced initial velocity $w^{(r)}_{km}$ measured after being hit from the tool, (resulted using action A_k and functional feature F_m, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred m calculated using Equation 4.7 and shown in Figure 4.13, the '*' mark is used, hence a smaller gap value is execpted. For the last column, $w^{(r)}_{53} \in \mathbb{R}^2$ was not used during the learning process, but $w^{(h)}_{53} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of Figure 4.13.

Affordances to emulate both learned and novel desired effects. To support my claim, I present errorbars (shown in Figure 4.16) using cumulative gap values representing inferred m and non-inferred m (averaged over all the evaluation effects). The errorbar is calculated using Equation 4.12 for the values shown in Figure 4.14 (shown as **Left Graph** in Figure 4.16) and Equation 4.13 for the values shown in Figure 4.15 (shown as **Right Graph** in Figure 4.16).

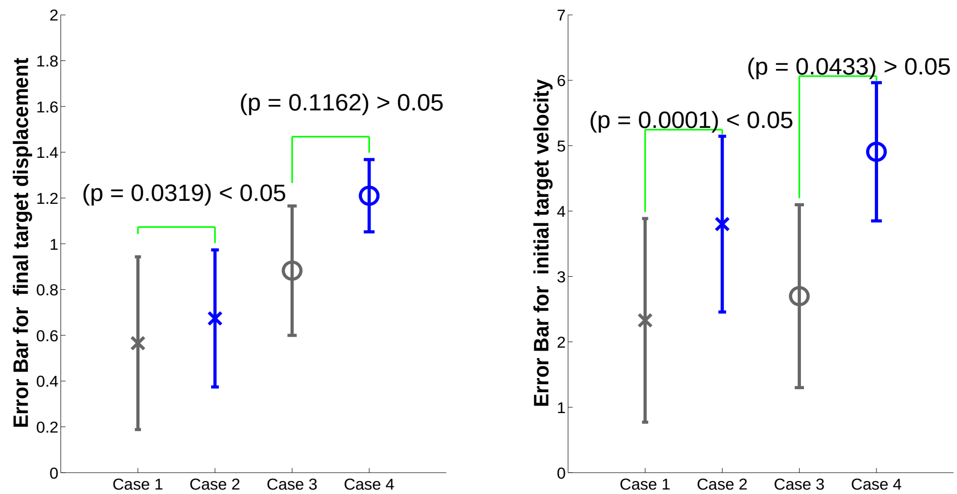


Figure 4.16: Comparison of error bars representing the gap between the effects as desired by human user with the ones experienced by the robot. The action A_k is given to the robot as input to realize $d_{ij}^{(h)}$ (**Left Window**) and $w_{ij}^{(h)}$ (**Right Window**). **Case 1:** inferred functional feature F_m is used to create all the 15 types of effects. **Case 2:** non-inferred F_m is used to create all the 15 types of effects. **Case 3:** inferred F_m is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$. **Case 4:** non-inferred F_m is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$.

To test our hypothesis that the result of inference to realize the desired effects than the non-inferred one, we perform hypothesis testing by designing two groups. "Group One" corresponds to the case when inferred functional feature was used and "Group Two" when "non-inferred" functional feature was used.

When the robot was asked to realized effects similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0319. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred F_m to realize the target displacement is statistically significant than the non-inferred F_m.

However, when the robot was asked to realize *only* novel displacement of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.1162. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred F_m to realize the novel target displacement may not always yield better results than non-inferred F_m.

When the robot was asked to realize target velocity similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0001. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred F_m to realize the target velocity is statistically significant than the non-inferred F_m.

However, when the robot was asked to realize *only* novel velocity of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.0433. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred F_m to realize the novel target velocity may not always yield better results than non-inferred F_m.

4.3.3.3 Inference 3 : Action with Tool Recognition and Selection.

$$P_{A,F} = P(A = A_k, F = F_m | d_{ij}^{(h)}, w_{ij}^{(h)}) \tag{4.14}$$

The probability of robot estimation of suitable action A_k with functional feature F_m to realize given effects by human demonstrator $d_{ij}^{(h)} \in \mathbb{R}^2$ and $w_{ij}^{(h)} \in \mathbb{R}^2$, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$ shown in Fig 4.17, is calculated using the following equation.

As previously demonstrated in Section 4.3.3.1, I validate the inference result by calculate matrix $\mathbf{E}$(D) and $\mathbf{E}$(W) using Equation 4.6 and Equation 4.7 respectively for $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$. Again due to brevity of space, I do not show the gap matrices for this inference, but my analysis show that the gap values to realize any pair of effect $d_{ij}^{(h)}, w_{ij}^{(h)}$ where using inferred A_k, F_m are much lower than the gap values corresponding to non-inferred action A_k, F_m. Thus, I claim that inferred A_k, F_m is most suitable Tool Affordances to emulate both learned and novel desired effects.

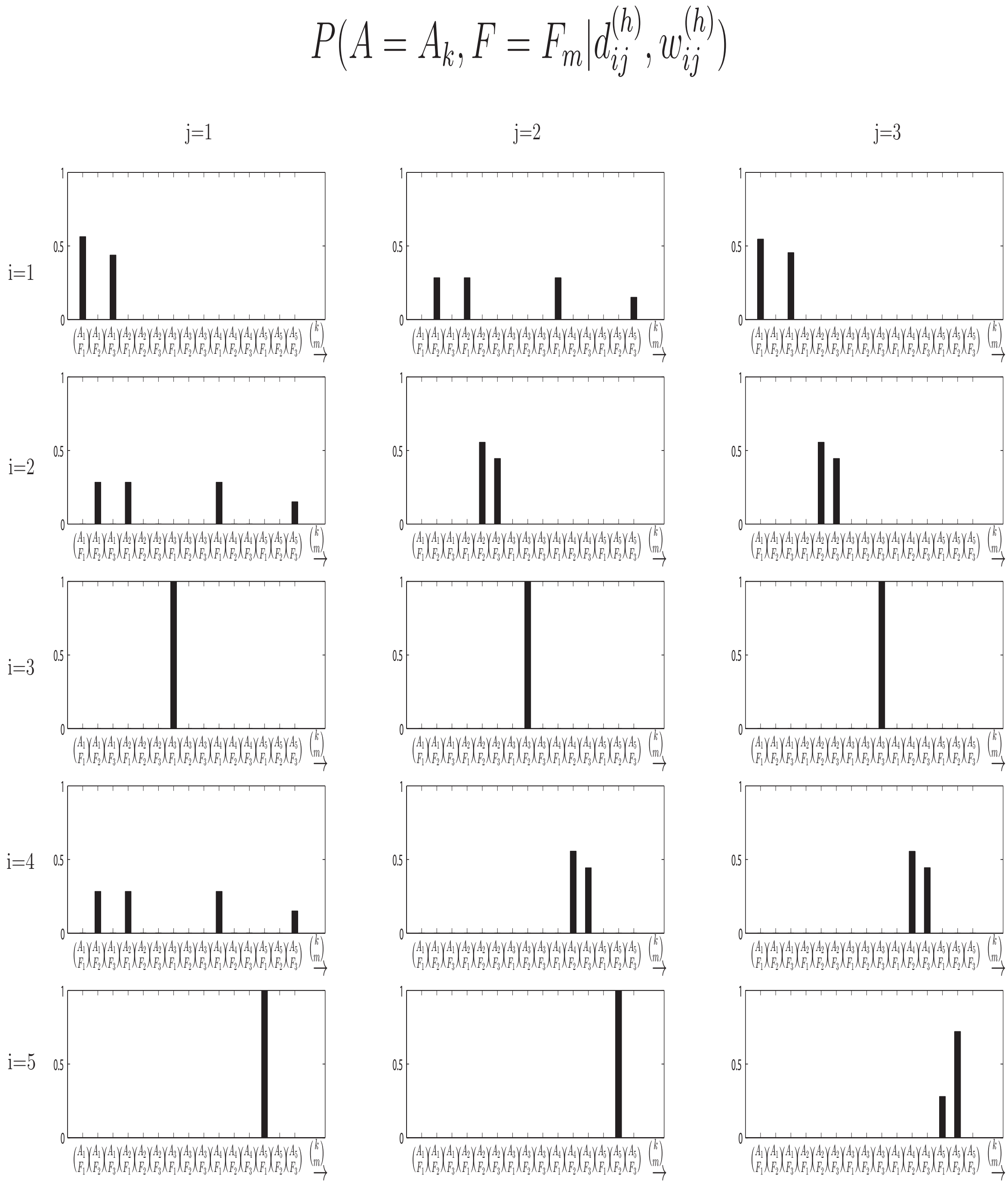


Figure 4.17: Robot estimate of suitable Action A_k and functional feature F_m to realize given effects by human demonstrator $(d_{ij}^{(h)}, w_{ij}^{(h)})$ (measured from the manipulation of Action A_i and functional feature F_j) ,where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$

To support my claim, I present errorbars (shown in Figure 4.20) using cumulative gap values representing inferred (k,m) and non-inferred (k,m) (averaged over all the evaluation effects).

$d^{(h)}_{ij}$ \ $d^{(r)}_{km}$	$d^{(r)}_{11}$	$d^{(r)}_{12}$	$d^{(r)}_{13}$	$d^{(r)}_{21}$	$d^{(r)}_{22}$	$d^{(r)}_{23}$	$d^{(r)}_{31}$	$d^{(r)}_{32}$	$d^{(r)}_{33}$	$d^{(r)}_{41}$	$d^{(r)}_{42}$	$d^{(r)}_{43}$	$d^{(r)}_{51}$	$d^{(r)}_{52}$	$d^{(r)}_{53}$
$d^{(h)}_{11}$	0.09*	0.32	0.09*	0.32	0.44	0.47	0.47	0.61	0.80	0.32	0.45	0.47	0.40	0.53	0.87
$d^{(h)}_{12}$	0.30	0.00*	0.33	0.00*	0.30	0.34	0.69	0.71	0.96	0.00*	0.31	0.34	0.63	0.62	1.02*
$d^{(h)}_{13}$	0.09*	0.35	0.08*	0.35	0.46	0.48	0.46	0.60	0.79	0.35	0.46	0.48	0.39	0.53	0.86
$d^{(h)}_{21}$	0.30	0.00*	0.33	0.00*	0.30	0.34	0.69	0.71	0.96	0.00*	0.31	0.34	0.63	0.62	1.02*
$d^{(h)}_{22}$	0.46	0.35	0.48	0.35	0.10*	0.09*	0.57	0.47	0.74	0.35	0.66	0.68	0.86	0.93	1.31
$d^{(h)}_{23}$	0.45	0.33	0.47	0.33	0.10*	0.10*	0.58	0.49	0.75	0.33	0.64	0.66	0.84	0.91	1.29
$d^{(h)}_{31}$	0.50	0.70	0.49	0.70	0.60	0.60	0.27*	0.39	0.47	0.70	0.89	0.91	0.73	0.95	1.20
$d^{(h)}_{32}$	0.71	0.82	0.71	0.82	0.62	0.60	0.41	0.30*	0.36	0.82	1.08	1.10	1.01	1.20	1.49
$d^{(h)}_{33}$	0.86	1.01	0.85	1.01	0.82	0.81	0.47	0.42	0.32*	1.01	1.25	1.27	1.10	1.33	1.56
$d^{(h)}_{41}$	0.30	0.00*	0.33	0.00*	0.30	0.34	0.69	0.71	0.96	0.00*	0.31	0.34	0.63	0.62	1.02*
$d^{(h)}_{42}$	0.45	0.33	0.47	0.33	0.63	0.67	0.91	1.00	1.23	0.33	0.09*	0.09*	0.54	0.39	0.79
$d^{(h)}_{43}$	0.46	0.35	0.48	0.35	0.65	0.68	0.93	1.02	1.24	0.35	0.09*	0.08*	0.54	0.38	0.78
$d^{(h)}_{51}$	0.49	0.70	0.48	0.70	0.89	0.91	0.77	0.98	1.11	0.70	0.61	0.60	0.23*	0.38	0.52
$d^{(h)}_{52}$	0.61	0.70	0.61	0.70	0.96	0.99	1.00	1.18	1.34	0.70	0.49	0.48	0.38	0.25*	0.44
$d^{(h)}_{53}$	1.01	1.15	1.01	1.15	1.39	1.42	1.31	1.53	1.64	1.15	0.95	0.93	0.70*	0.64*	0.46

The desired effects given by the human for the robot to realize

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.18: Matrix $\mathbf{E}$(D) representing gap of final displacement of target object D, between the desired displacement $d^{(h)}_{ij}$ (given during evaluation process) and experienced displacement $d^{(r)}_{km}$ (resulted using action A_k and functional feature F_m, where $(i, k) \in [1, 5]$ and $(j, m) \in [1, 3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred k, m calculated using Equation 4.6 and shown in Figure 4.17, the '*' mark is used, hence a smaller gap value is expected. For the last column, $d^{(r)}_{53} \in \mathbb{R}^2$ was not used during the learning process, but $d^{(h)}_{53} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of Figure 4.17.

To test our hypothesis that the result of inference to realize the desired effects than the non-inferred one, we perform hypothesis testing by designing two groups. "Group One" corresponds to the case when both inferred action and inferred functional feature was used and "Group Two" when "non-inferred" action and non-inferred functional feature was used.

The desired effects given by the human for the robot to realize

$w_{ij}^{(h)}$ \ $w_{km}^{(r)}$	$w_{11}^{(r)}$	$w_{12}^{(r)}$	$w_{13}^{(r)}$	$w_{21}^{(r)}$	$w_{22}^{(r)}$	$w_{23}^{(r)}$	$w_{31}^{(r)}$	$w_{32}^{(r)}$	$w_{33}^{(r)}$	$w_{41}^{(r)}$	$w_{42}^{(r)}$	$w_{43}^{(r)}$	$w_{51}^{(r)}$	$w_{52}^{(r)}$	$w_{53}^{(r)}$
$w_{11}^{(h)}$	0.35*	2.52	0.42*	2.52	3.65	3.74	1.79	2.80	2.65	2.52	3.68	3.74	1.88	2.87	2.73
$w_{12}^{(h)}$	2.45	0.00*	2.72	0.00*	2.65	2.76	3.20	3.23	3.75	0.00*	2.68	2.76	3.31	3.30	3.82*
$w_{13}^{(h)}$	0.45*	2.79	0.33*	2.79	3.84	3.92	1.78	2.88	2.64	2.79	3.87	3.93	1.86	2.94	2.72
$w_{21}^{(h)}$	2.45	0.00*	2.72	0.00*	2.65	2.76	3.20	3.23	3.75	0.00*	2.68	2.76	3.31	3.30	3.82*
$w_{22}^{(h)}$	3.73	2.81	3.91	2.81	0.38*	0.36*	2.89	1.82	2.70	2.81	5.48	5.56	5.39	5.85	6.13
$w_{23}^{(h)}$	3.66	2.72	3.84	2.72	0.37*	0.37*	2.86	1.81	2.70	2.72	5.39	5.47	5.31	5.76	6.04
$w_{31}^{(h)}$	1.80	3.22	1.79	3.22	2.85	2.89	0.37*	1.35	0.94	3.22	5.20	5.27	3.60	4.61	4.47
$w_{32}^{(h)}$	2.86	3.34	2.93	3.34	1.88	1.88	1.39	0.44*	0.96	3.34	5.76	5.84	4.70	5.54	5.55
$w_{33}^{(h)}$	2.61	3.72	2.60	3.72	2.69	2.69	0.92	0.98	0.43*	3.72	5.90	5.97	4.42	5.41	5.29
$w_{41}^{(h)}$	2.45	0.00*	2.72	0.00*	2.65	2.76	3.20	3.23	3.75	0.00*	2.68	2.76	3.31	3.30	3.82*
$w_{42}^{(h)}$	3.69	2.76	3.88	2.76	5.39	5.51	5.24	5.72	6.00	2.76	0.34*	0.33*	2.93	1.84	2.72
$w_{43}^{(h)}$	3.72	2.80	3.91	2.80	5.43	5.55	5.27	5.76	6.04	2.80	0.33*	0.32*	2.94	1.84	2.72
$w_{51}^{(h)}$	1.91	3.35	1.88	3.35	5.29	5.39	3.61	4.65	4.48	3.35	2.93	2.96	0.40*	1.40	0.96
$w_{52}^{(h)}$	2.90	3.37	2.97	3.37	5.76	5.87	4.65	5.51	5.50	3.37	1.89	1.88	1.41	0.41*	0.96
$w_{53}^{(h)}$	2.77	3.87	2.75	3.87	6.03	6.13	4.49	5.51	5.36	3.87	2.75	2.75	0.99*	1.00*	0.41

The actual effects realized by the robot using action A_k and functional feature F_m

Figure 4.19: Matrix $\mathbf{E}(\mathrm{W})$ representing gap of initial velocity of object after being hit from the tool W, between desired initial velocity $w_{ij}^{(h)}$ (given during evaluation process) and experienced initial velocity $w_{km}^{(r)}$ (resulted using action A_k and functional feature F_m, where $(i,k) \in [1,5]$ and $(j,m) \in [1,3]$). The gap value is represented using greyscale parametric shading i.e. larger the gap value, darker the background. For the gap value corresponding to actually inferred k, m calculated using Equation 4.7 and shown in Figure 4.17, the '*' mark is used, hence a smaller gap value is expected. For the last column, $w_{53}^{(r)} \in \mathbb{R}^2$ was not used during the learning process, but $w_{53}^{(h)} \in \mathbb{R}^2$ was given during evaluation process as *novel effect* as shown in last row of Figure 4.17.

When the robot was asked to realized effects similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0001. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred A_k, F_m to realize the target displacement is statistically significant than the non-inferred A_k, F_m.

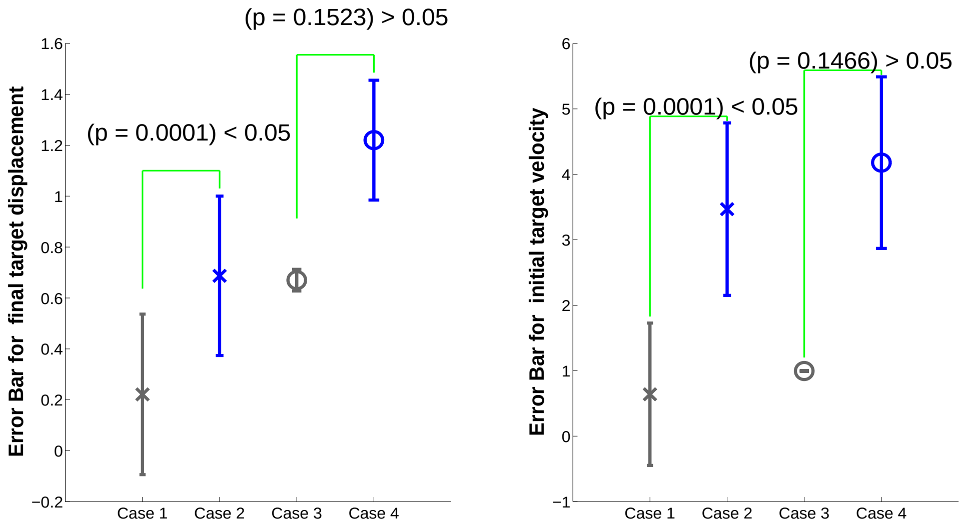


Figure 4.20: Comparison of error bars representing the gap between the effects as desired by human user with the ones experienced by the robot. The effects to realize are $d_{ij}^{(h)}$ (**Left Window**) and $w_{ij}^{(h)}$ (**Right Window**). **Case 1:** inferred action and functional feature A_k, F_m and is used to create all the 15 types of effects. **Case 2:** non-inferred A_k, F_m is used to create all the 15 types of effects. **Case 3:** inferred A_k, F_m is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$. **Case 4:** non-inferred A_k, F_m is used to create only the *novel effect* $(d_{53}^{(h)}, w_{53}^{(h)})$.

However, when the robot was asked to realize *only* novel displacement of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.1523. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred A_k, F_m to realize the novel target displacement may not always yield better results than non-inferred A_k, F_m.

When the robot was asked to realize target velocity similar to that of learned ones as well as novel ones, the two-tailed P value equals 0.0001. By conventional criteria, this difference is considered to be statistically significant. Thus, we can say that using inferred A_k, F_m to realize the target velocity is statistically significant than the non-inferred A_k, F_m.

However, when the robot was asked to realize *only* novel velocity of target object, then the P-test on gap values using both the groups suggest that, the two-tailed P value equals 0.1466. By conventional criteria, this difference is considered to be not statistically significant. Thus, robot should ask human user for feedback, since using inferred A_k, F_m to realize the novel target velocity may not always yield better results than non-inferred A_k, F_m.

4.3.3.4 Inference 4 : Prediction of Effects

After the learning is performed using 180 samples for each action A_i and functional feature F_j, the prediction capability of learning affordanes is tested .The task of the robot is to predict the effects, which a human user can create using an action and tool. To evaluate, the robots prediction capability, the human user manipulates the target 60 times for each pair of action and functional feature. For each of manipulation trial corresponding to the action A_k , the magnitude of tool velocity varies.

The robot observes the action performed by the human user, the velocity of the tool at which it impacts the target and also the functional feature of the tool. However, the robot does not get to observe the effects the human has created, which it is required to predict. Using the Bayesian network, robot predicts the mean and variance of the effects i.e. target displacement and target velocity after the impact of tool. In total 900 such test cases were given to the robot (i.e. 60 manipulations corresponding to each pair of A_k, F_m).

Thus, the gaussian distribution of effects for the action A_k ($k \in [1, 5]$) and functional feature F_m ($m \in [1, 3]$) given by human demonstrator is predicted by the robot using:

$$P_D = P(D = d_{km}^{(r)} | A_k, v_k^{(h)}, F_m) \tag{4.15}$$

$$P_W = P(W = w_{km}^{(r)} | A_k, v_k^{(h)}, F_m) \tag{4.16}$$

where $P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$ and $x \sim \mathcal{N}(\mu, \sigma^2)$

A larger set of continuous effects can also be inferred together using multivariate gaussian distribution over set of random random variables $X_1, \ldots, X_n$. It is a parameterized by an n-dimensional mean vector μ and an n x n positive definitive covariance matrix $\sum$, where

$P(X) = \frac{1}{(2\pi)^{n/2}|\sum|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T(\sum)^{-1}(\mathbf{x}-\mu)\right)$

For each manipulation trial made by the human, the error between actual target displacement resulted by the manipulation made by human user, $d_{km}^{(h)}$ and the target displacement as predicted by the robot, $\hat{d}_{km}^{(r)}$ is measured as:

$$\epsilon(d_{km}) = \hat{d}_{km}^{(r)} - d_{km}^{(h)} \tag{4.17}$$

Similarly, the error between actual target velocity, $w_{km}^{(h)}$ (after its impact by the tool) resulted by the manipulation made by human user and the target velocity as predicted by the robot, $\hat{w}_{km}^{(r)}$ is measured as:

$$\epsilon(w_{km}) = \hat{w}_{km}^{(r)} - w_{km}^{(h)} \tag{4.18}$$

The prediction accuracy is calculated by the robot as:

$$accuracyofprediction = \frac{\text{desired effect - predicted effect}}{\text{desired effect}} \tag{4.19}$$

The mean error of predictions made by the robot is denoted by $\epsilon(d_{km})$ and $\epsilon(w_{km})$. The percentage of mean error of predictions made by robot will give an estimate of how accurately robot can predict. It is calculated as:

$$a(\epsilon(d_{km})) = \frac{\mu(\epsilon(d_{km}))}{\mu(d_{km}^{(h)})} \tag{4.20}$$

$$a(\epsilon(w_{km})) = \frac{\mu(\epsilon(w_{km}))}{\mu(w_{km}^{(h)})} \quad (4.21)$$

Table 4.33: The percentage of mean error in robot's predicting of effects resulted from human manipulation using action A_k and functional feature F_m.

k	m	$\mu(d_{km}^{(h)})$	$\mu(w_{km}^{(h)})$	$a(\epsilon(d_{km}))$	$a(\epsilon(w_{km}))$
1	**1**	0.308	2.510	0.006	0.00
1	**2**	0.00	0.00	0.00	0.00
1	**3**	0.359	2.863	0.004	0.00
2	**1**	0.00	0.00	0.00	0.00
2	**2**	0.363	2.893	0.005	0.00
2	**3**	0.445	3.159	0.001	0.00
3	**1**	0.870	3.981	0.259	0.0001
3	**2**	0.658	3.620	0.281	0.00
3	**3**	0.488	3.072	0.237	0.373
4	**1**	0.0	0.00	0.00	0.00
4	**2**	0.362	2.91	0.0049	0.00
4	**3**	0.320	2.72	0.0050	0.00
5	**1**	0.437	2.763	0.0921	0.00
5	**2**	1.16	3.752	0.3075	0.00
5	**3**	0.743	3.54	0.0407	0.005

Thus for all the pairs of A_k, F_m, accuracy of prediction of target displacement is 80.89 % and of target velocity is 99%.

Thus, by using the Bayesian tool affordanes, a robot can also acquire the capability to estimate probable effects with high accuracy, adding to its capability for generating plans using the known actions and available tools.

4.4 Summary and Conclusion

In this chapter, I attempted to realize the goal of developing a function to determine a tool representation that enables robot to transfer the tool-use skill to different tools as well as select alternative tools by estimating their effects. The problem to realize the goal is that the casual relationship between features of tool and its functionality was not implicitly established because all the features of tool are not functionally relevant. To solve this problem, I proposed encode tool use skill in the features of the tool that have casual influence on its functionality. My hypothesis was that functional features remain invariant across the tools that offer the similar functionality. Using the experiments, I showed that use of functional features in tool-use skill enables robot to estimate the effects of unseen tools.

I also attempted to realize the goal of developing a function to encode tool use skill that enables a robot to infer unobservable information from the observed ones. This function is required to enable acquire the inference capabilities required for an autonomous tool user. These include the capability to determine suitable action and estimate its parameters, to select a suitable tool by determine its function and estimate the outcome of target manipulation.

To realize this goal the problem was that relationship among actions, tool representation and effects is not encoded in such a way that it can deal with hidden stochastic and casual structures as well as noisy data. To address this problem, I proposed robot learning of casual probabilistic dependencies between actions, functional features and observed effects, termed as Tool Affordances. The probabilistic semantics of Bayesian Network is used to learn tool affordances because the uncertainties in domain/ background knowledge and inferences can be expressed in probabilistic terms. Also Bayesian Network presents a large number of learning and inference algorithms that suit the requirements of my goal.

Chapter 5

Conclusion and Future Works

5.1 Conclusion

The goal of my research is to enable a robot to use a variety of tools by generalizing the tool used acquired using few tools.

In Chapter 4, I attempted to realize the goal of developing a function to determine a tool representation that enables robot to transfer the tool-use skill to different tools as well as select alternative tools by estimating their effects. The problem to realize the goal is that the casual relationship between features of tool and its functionality was not implicitly established because all the features of tool are not functionally relevant. To solve this problem, I proposed encode tool use skill in the features of the tool that have casual influence on its functionality. My hypothesis was that functional features remain invariant across the tools that offer the similar functionality. Tool representation is done using the geometrical feature of the tool that influences the functionality of the task. The objective of using these functional features is to generalize of tool use skill to unseen tools,actions and effects. Robot manipulates the target object using a pre-designed action and a functional feature of the T shaped tool. T shaped tool has three functional features i.e. Horizontal Part, Vertical Part and Corner. Using the experiments, I showed that use of functional features in the usage of tool enables robot to estimate the effects of unseen tools.

I also attempted to realize the goal of developing a function to encode tool use skill that enables a robot to infer unobservable information from the observed ones. This function is required to enable acquire the inference capabilities required for an autonomous tool user. These include the capability to determine suitable action and estimate its parameters, to select a suitable tool by determine its function and estimate the outcome of target manipulation.

To realize this goal the problem was that relationship among actions, tool representation and effects is not encoded in such a way that it can deal with hidden stochastic and casual structures as well as noisy data. To address this problem, I proposed robot learning of casual probabilistic dependencies between actions, functional features and observed effects, termed as Tool Affordances. The *effects* of target manipulation are final target displacement and initial target velocity after the impact of functional feature. A total of five actions are pre-designed where each actions is the cluster-id of varying tool velocities in a specific direction. Using all possible combinations of Actions and functional features, robot manipulates the target object and observations are recorded to learn the Tool Affordances.

Tool Affordances encode the probabilistic dependencies between Action, functional feature and the resulting effects of manipulation. To test the learnt tool affordances in acquiring inference capabilities for tool recognition and selection, action recognition and selection as well as prediction of effects of target manipulation, human demonstrator presents effects from all possible combinations of Actions and functional features to the robot. The presented inference results show robot's capability to emulate effects (including effects novel to the robot) given by human demonstrator by performing the target inference. I also perform validation of inference results and show that learned Tool Affordances are suitable for emulation of desired effects.

The probabilistic semantics of Bayesian Network is used to learn tool affordances because the uncertainties in domain/ background knowledge and inferences can be expressed in probabilistic terms. Also Bayesian Network presents a large number of learning and inference algorithms that suit the requirements of my goal. The presented inference results that show robot's capability of tool recognition and selection, action recognition and selection as well as action,tool recognition and selection to realize desired effects. I also showed that the generalization of functional feature enables robot to estimate the effect of unseen tools. Using the internal belief state of the robot expressed in probabilistic terms, a robot can request feedback from human for some structured input about domain/background knowledge and tool-use skill.

For each tool-use task, a separate tool-use model is required. Thus, when different tools like pen, scissors, spoon, tweezers, hammer etc are to perform the tasks which require a distinct functionality, robot is required to observes the target manipulation and model the corresponding tool affordances. For such cases, the approach of Nakamura and Nagai[49] serves as a good example, however as discussed in Section 2.3, the approach requires frequent experimental tuning of hyper-parameters i.e. parameter of a prior distribution, based on the performance of the model. Thus, in order to extend my approach for different tasks, one of the candidate of my future research is to build on their work and develop a function to suitably estimate the hyper-parameters based on cross-validation of the model.

5.2 Future Works

I intend to develop a function to enable robot learn the usage of tools such that learnt skill can be used to solve *similar* tool using tasks. This is required because programming a robot to use each tool is not feasible due to the variations in the desired effects of each task, actions required for the manipulation and the changes in perceivable features of each tool. In my work, two tasks are considered *similar* when their effects match qualitatively. For example, in case of the task of *object movement*, the desired effect of the task is to move a target object from place to other in a specific direction. Thus, the movement of object for 5 cm or 50 cm or any distance are considered the same task as far as their angular movement directions are within certain threshold.

To use a tool autonomously in order to realize the effects of some tool using task, a tool using agent requires the following capabilities:

Capability 1: to determine the required effects of the task.

Capability 2: to identify a suitable tool for the given task.

Capability 3: to determine suitable tool placement pose i.e. correct position and orientation in which tool should be placed relative to the target object and then generate suitable action to achieve such a tool pose

Capability 4: to generate the action to manipulate the tool after it is correctly placed.

For capability 1, a robot is required to learn about the descriptors of each task using which the features of target object that change during tool use can be expressed e.g. final target displacement to express change in position for *object movement* task, surface area to express contour change for *amplification of strength* task. For capability 2 it is required to learn about the features of the tool relevant to its functionality that depend on required effects of the task and structural constraints with the target object. For example, it makes a better tool when constituent part of the tool that manipulates the target has flat surface as geometry and when its surface is longer than surface of target object. For capability 3, a robot is required to pay attention to relationship between desired effects and the spatial constraints between tool and target object. For example, to retrieve a distant object, a suitably placed too will have its surface behind the surface of target object in order to pull it. Then to generate action, it should be able to reason in terms of abstract parameters such as force and torques. Such reasoning is also required for capability 4 when linear forces (for Stick, L-shape tool etc), impact forces (hammer, stone etc) and rotational forces (screwdriver etc) are required to be generated.

The limitation of my work of tool affordances ([95–97]is that learning to use tool is not considered. The knowledge and skills required to use a tool were pre-programmed. The problem is that, it is not feasible for a human designer to provide the prior knowledge and skills for acquiring the capabilities to use a tool and to learn the casual dependencies among them, because such an exercise is quite time consuming and resource intensive. Moreover, different tasks may require different knowledge and tool-use skills so such an approach will not be generally applicable to wide range of tool-use tasks.

In future, I intend to propose an approach (discussed in Section 5.2.1) to learn the usage of tools in an online and incremental manner is proposed, in which the robot acquires the capabilities to use tool via interaction with the human user (refer Setcion 5.2.4 for proposed implementation of my approach). The human user demonstrates the process of tool use, such that concept information is done incrementally to yield the important information associated with each of the four capabilities of tool-use. Robot should formulate an initial hypothesis about the functional features of the tool, tool-pose and actions for tool-placement and target manipulation. It should then learn the casual dependencies among them using BN based tool affordances. The hypothesis is then tested by experimenting with a variety of tools to perform similar tasks (refer Section 5.2.5) for which inferences are made using learnt tool affordances. The uncertainties in domain/task knowledge or the inference results prompt the robot to make an interaction with the user and update its hypothesis by incorporating user feedback in its learning.

Thus, the aim of my future research is to address the problem of minimizing human effort and the computational complexity involved in autonomous tool use of the robot by proposing an approach in which robot learns tool use based on interaction with the human user.

5.2.1 Candidate Concept for learning tool use model

The concept of learning usage of tools can be divided into the parts that represent action, tool representation and effects.

Extraction of Visual Descriptors

The robot is provided with a set of visual descriptors to be used for tool and object recognition and classification. These describe shape, size, position and color of the segmented objects and tools. I extract these Visual Descriptors using 2d images of tool and object.

- Normalized x and y coordinates of the center of the enclosing rectangle and surface normals.
- Normalized width, height and angle (orientation) of the enclosing rectangle.
- Hue normalized color histogram of the pixels inside the object's region.
- Area (number of pixels).
- Convexity - ratio between the perimeter of the object's convex hull and the perimeter of the object's contour.
- Eccentricity - ratio between the minor and major axis of the minimum-area enclosing rectangle.
- Compactness - ratio between the object area and its squared perimeter.
- Circleness - ratio between the object area and the area of its enclosing circle.

- Squareness - ratio between the object area and the area of its minimum-area enclosing rectangle.

- SIFT Features.

Defining Task Spaces to represent effects

The effects are modeled by Gaussian distribution of visual changes that occur between the end and start of manipulation. In the task of *Extension of Reach*, the effects are modeled as relative distance between final and initial object position, to provide generalization over absolute positions of the target object. In case of *Amplification of Strength*, effect are described as modified length of the nail before and after manipulation. To avoid the burden of pre-programming the definition of effects, I propose that visual changes that occur between the end and start of manipulation for each task (see Table 3.1) should be mapped to some task space. I define the following types of effects of tool Use on target objects:

- **Position Change:** relative distance of object placement.

- **Orientation Change:** relative orientation of object.

- **Color Change :** correlation coefficient between color histograms of object

- **Contour Change:** correlation coefficient between fourier descriptors of object

- **Change in no of objects:** count connected components of object.

It is required for the robot to autonomously select an appropriate task space for the given task by observing the effects. I propose that the robot would be provided with a pool of task spaces mentioned above to which observations from human demonstration shall be mapped. Table 5.1 shows the correlation between tools and task spaces.

Identify a good tool for the given task

Table 5.1: The effects required for solving different tasks using the tools

Tools	Color	Contour	Position	Number of objects
Stick, Hook, L-Shape			yes	
Knife, Scissor		yes(shape)		
Hammer, Stone		yes(shape)		yes/no
ScrewDriver		yes(length)	yes	
Push Pin, Stappler				yes
Pen, Pencil	yes			
Paperweight, Clamps		yes(mass)		

I had defined functional features as the casually relevant features of the tool that determine its functionality for a given task. For example, for the tools used to deform object shapes i.e. knife, blade etc; the sharp edge is the functional feature, but for the tools that amplify physical strength i.e. stone, hammer etc; the mass, material are the functional features. Examples of functional features are shown in Table 3.1. my hypothesis is that these functional features remain distinctive and invariant across different tools used for performing similar tasks. Identifying a good tool requires the robot to determine that what are the required functional features to realize the desired effects and which constituent part of the tool has those functional features. It also requires determining what is the required grip on the tool to carry out the given task and whether the available tools provide such a grip. I propose to divide a tool in 3 different segments, each with its own visual descriptors parameters:

1. Handle: It is the graspable part of the tool. The manner in which one grips the handle can be used to transmit force through a tool in several ways, i.e., impact forces (as with using hammers), linear forces (as with using L shape, T shape, Stick tool, saws etc), rotational forces (as with using screwdrivers): in each case, the requirement for transmitting force will have a bearing on the interaction between user and handle. Please refer Table 5.2 for more details.

2. End-effector: The end extremity of the tool which generally come into contact with the object. Also, it is generally the end-effector which carries the functional feature of the tool (but this is not the requirement e.g. in L shape tool, even though the horizontal part of the tool is the end effector, but both vertical and horizontal parts of the tool can be used to manipulate the tool since they both carry the functional feature which is flat surface). Thus, the end effector signifies both form and function.

3.Body: The connecting part between the handle and the end-effector.

I make this discrete parameterizations of each part of a tool, so that it is possible to have a better generalization capability to the learning of the affordances. For example, certain tools might not afford certain actions not because its end-effector is inappropriate (in case the form of end effectors carries functional features) but because the tool might be too short to reach the target object or simply not graspable by the robot. Thus identifying a good tool requires the robot to learn the dependencies between the desired effects, structural constraints of tool and target object and spatial constraints of the environment.

For example, in case of the task *Extension of Reach*, the desired effect is to bring an object kept at distance, closer to the robot. A suitable tool would be the one that enables "Finger-palm" contact to enable "hook" grip affordance. Table 5.2 shows the different types of contact a human makes on the handle of tool to realize the goal of the task. It is shown (Baber[2]) that types of contact between hand and tool has correlation with the types of grip a tool makes on target object and the goal of the task. Thus, such correlation can be used to recognize the intention of the user and also the goal of the task. In my work, human shows the grasped tool to the robot from different viewpoints by rotating its wrist. The robot calculates the SIFT visual descriptors of the hand shape of tool grasp. These descriptors are later trained using Support Vector Machines to classify the types of grip as shown in Table 5.2.

Table 5.2: Types of Grip. The list below is inspired from Table 6.2 of the book "Cognition and Tool Use" by Christopher Baber[2]

Type of Grip	**Contact**	**Description**	**Goal of the task**	**Tools Used**
Hook	Finger-palm	Palm against surface, Pulling a lever and fingers hooked around the tool	Extension of Reach	Stick, L shape, T shape, Egg headed Stick
Power	Hand	Tool rested across palm and enclosed by fingers	Amplification of strength	Hammer, Saw
Scissor	Thumb-two fingers (inside)	Fingers and thumb placed inside handles of the tool	Cutting paper or cloth	Scissors
Pinch	Thumb-finger-palm	Tool resting against palm, and grasped between thumb and fingers	Positioning screw driver head onto a screw	Screwdriver
Lateral	Thumb-forefinger	Tool between thumb and forefinger	To catch an object and lift up	Using tweezers
Pen	Thumb-two fingers (outside)	Tool rested on thumb and pressed by two fingers	Coloring, writing on a surface	Pen

5.2.2 Generation of suitable tool placement strategy

After the suitable contact surface of the tool is selected, it is placed to manipulate the target object. By observing the human demonstration of tool placement, robot forms a hypothesis of what constituents a suitable tool placement. For example, when human uses flat geometry of horizontal part of L-shape tool to move a distant object closer in *Extension of Reach* task, the robot observes that the normal of the contact surface of tool should be aligned with vector describing target position change. Then such correlation between surface normal of contact surfaces and displacement vector representing the effect is learned by the robot. Henceforth when some tool is given to the robot to realize the similar effect, it can generate surface normals of its surfaces and using the learned correlation find a suitable surface and placement strategy.

In case of task of *amplification of strength* using hammer to insert a nail into wood, the axis of nail should be at right angle relative to surface of wood and axis of heavy head should be aligned with axis of nail. In this case, the effect is described as contour change (see Section 5.2.1) i.e. modified length of the nail before and after manipulation. Here, robot learns the correlations between the surface normals of nail, head of the hammer and surface of the wood with that of vector representing the effect.

5.2.3 Manipulation of Target Object

A robot requires the action model to manipulate target object using the L-shape tool. The action models describe how executing each behavior affects the world. For example, to manipulate a cubic target, it needs the following action plan in order:

- go-to(L-shape-tool) /* The robot goes towards the L-shaped tool */
- pick-up(L-shape-tool) /* The robot picks up the L-shaped tool in its gripper */
- generate-placement(L-shape-tool, horizontal-part, target-object) /* The robot places L-shape tool in a position and orientation such that it "hooks" the target object using its horizontal-part */
- pull-with-tool(L-shape-tool, target-object) /* The target object is pulled using the L-shaped tool */
- match (target-current-position, desired-position) /* If target reaches its desired position, robot will stop */
- drop(L-shape-tool) /* The tool is detached from its gripper*/

Such sequence of actions is required to realize the effects. The action model of some of these actions is provided to the robot as domain knowledge by the human designer. But when for example the action models corresponding to the **placement-of-tool** and **pull-with-tool** are not provided, then robot should learn them by requesting human user for a demonstration. The robot has to construct action model and domain knowledge from the human demonstration, then test and refine it by trial and error process as well as further instruction. Robot thus initially forms the hypothesis about using a particular tool. The informative experiments that test this hypothesis is created by the human user, thus limiting the search space to a practical number of experiments. The human user starts by demonstrating single observation to the robot. The objective of single demonstration is to enable robot learn from fewer manipulation trials in an incremental manner, such that learning outcome of one manipulation trial helps in refining the subsequent trials to achieve the desired effects.

5.2.4 Candidate Implementation of learning from human interaction

The underlying process of learning from interaction is:

1. Robot observes the demonstrator manipulate a target object using some action and tool to realize the required effects.

2. (a) Robot maps the effects on the task spaces defined in Section 5.2.1.

   (b) The *goal of the task* is determined by observing the type of contact user makes on the tool while manipulation to enable a suitable grip, as shown in Table 5.2.

   (c) The functional features are selected using the approach detailed in Section 5.2.1.

   (d) The action model for the placement of constituent part of the tool that has contact surface relative to target object is constructed using approach mentioned in Section 5.2.2.

   (e) The action model for subsequent target manipulation after it is suitably placed is constructed as explained in Section 5.2.3. To perform manipulation, the human user applies linear force or impact force or rotational force to manipulate the target object depending on the task and selected tool. These forces are dependent of several parameters e.g. in case of task of *amplification of strength* the impact force is proportional to the energy imparted to the head, which, in turn, relates to the mass of the head, the angular velocity of the swing, the distance covered during the swing and the magnitude of the force applied by the user. Such dependency graph is presented to the robot by human programmer.

3. Robot learns the dependencies between desired effects, functional features of the tool with action models for tool placement and target manipulation. Such learning is performed using the probabilistic semantics of Bayesian networks.

4. The human user designs an experiment in which,robot is required to perform the similar task realize i.e. realize the similar effects, but by using some different tool. This experiment is to test the robots learning of selection of functional features and placement strategy by identifying the required dependencies between desired effects, features of the tool and structural constraints of tool and target object.

5. If robot fails in performing any of the required four components of tool use or if the accuracy of inference shows a smaller probability, then it makes queries to human user for a demonstration using the same tool. The interaction enables the robot to incorporate more domain and task-specific knowledge for subsequent manipulation trial. Steps 1-5 are repeated till the success is achieved.

5.2.5 Candidate Experimental condition

I intend to perform the experiments in which robot learns to use the tools shown in Figure 5.2. The effects to realize is move a target object kept at distance closer to the robot. The experiments are performed in SIGVerse (see Inamura et al.[5]) , which is a simulator that combines dynamics, perception, and communication simulations for synthetic approaches to research into the genesis of social intelligence. Please refer to Section 6.2 for more details about research and development of SIGVerse.

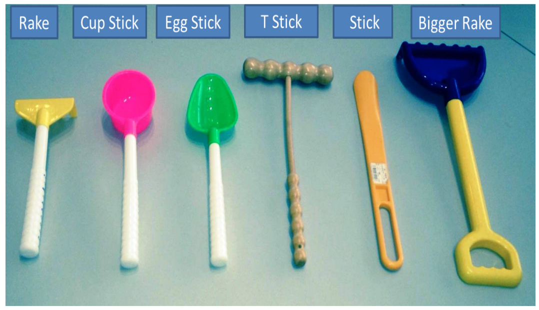
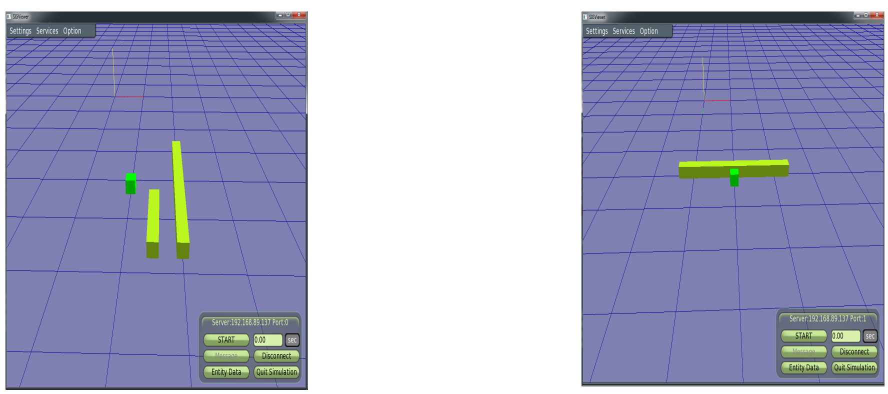


Figure 5.1: Different candidate tools to be used to realize the task of extension of reach.

Experiment 1: Robot manipulation of target object using stick as the tool.

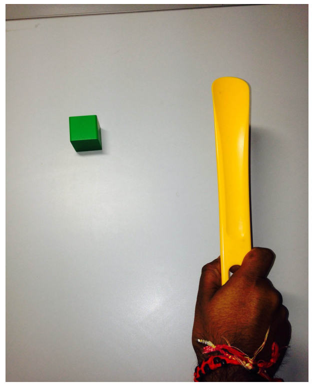
Two candidate sticks are kept on the table as show in (a) of Figure 5.2. Robot observes the tools and target object and calculates the visual descriptors. But it is unable to determine the suitable functional features, tool placement strategy and



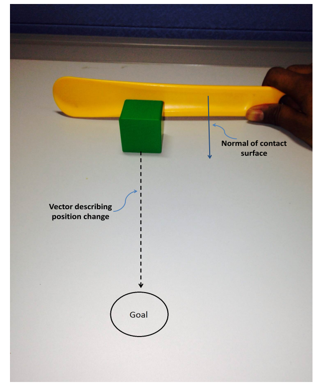
(a) Two candidate sticks to manipulate the target object from its initial position to the goal.

(b) Human selects larger of the two sticks and places it behind the target object by rotating the stick.

Figure 5.2: Robot is required to perform extension of reach task using stick tool as shown inside SIGVerse simulation environment.



(a) The human user grips the stick tool with *Finger-palm* contact.



(b) The normal of contact surface is aligned with the vector representing effect.

Figure 5.3: Human demonstrates how to grasp and place the tool to enable successful target manipulation.

action to manipulate target object. Thus, it requests demonstration from human user.

Experiment 2: Robot learning stick-tool use by observing the human demonstration.

Human demonstrates target object manipulation using the Stick-tool through process as described in Section 5.2.4.

1. **Selection of tool:** Human selects the larger stick tool.

2. **Grasping the tool:** Human grasps the tool as shown in (a) of Figure 5.3.

3. **Placement of tool:** Human rotates the wrist and places the surface of tool behind the target object as shown in (b) of Figure 5.3.

4. **Target manipulation:** Human uses **pull-with-tool** action to manipulate target from its initial position to the goal.

What robot learns from interaction?

Robot learns the domain knowledge about what constitutes a suitable grasp, what are the functional features of the tool for the given task, a suitable tool placement strategy and the action required for subsequent manipulation of target. It also learns the spatial constraints of environment and how to map the observed effects into a pool of task spaces provided to the robot. The details of its learning presented using literals is shown in Table 5.3. Along with domain knowledge and individual components of tool use, the robot encodes the casual dependencies between the cause and effects of target manipulation from human demonstration using a Bayesian network shown in Figure 5.4. This BN is required to evaluate the tool affordances by making inferences listed in Table 3.4.

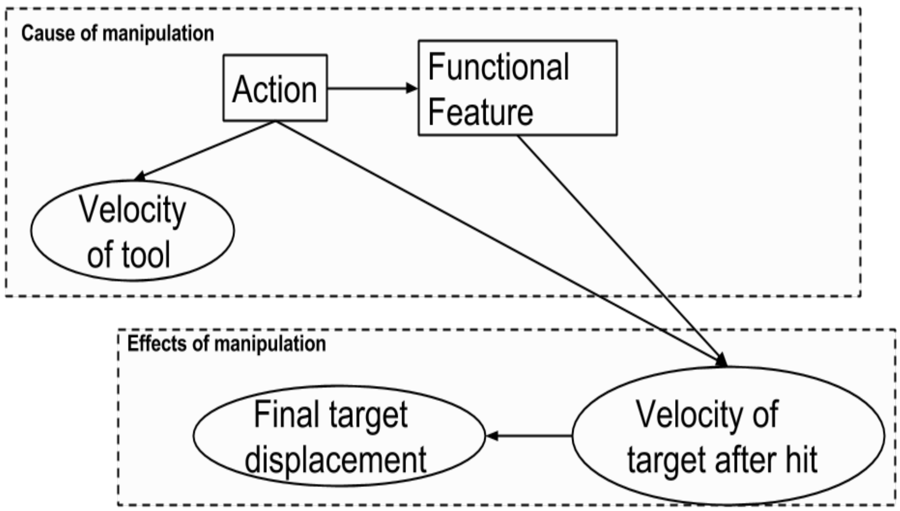


Figure 5.4: The dependencies between the cause and effects of target manipulation are encoded in Bayesian network(BN). The box represents the nodes having discrete data and ovals represent nodes with continuous data. The two actions used are **rotate**(tool) and **pull-with-tool**(tool, target-object). The former is used for placement of tool and later one for the subsequent manipulation. Action causes the placement of constituent part of the tool which has functional feature, so an arc is added. Using action and functional feature, the target is manipulated which causes its displacement, hence arcs representing casual dependencies are added.

Experiment 3: Robot manipulation of target object using T shape tool to test its learning.

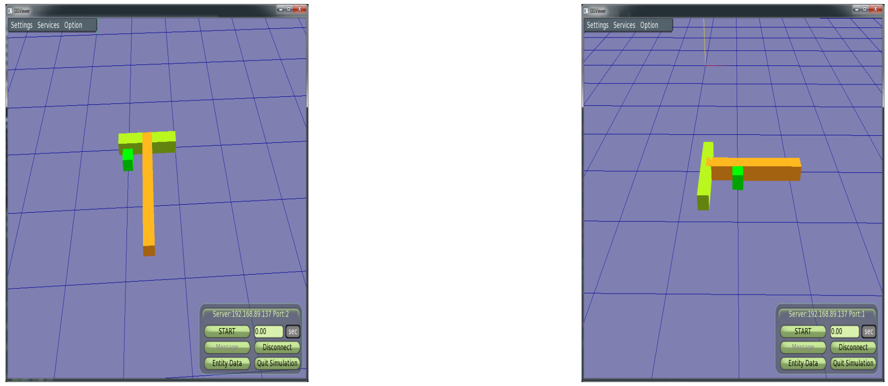
Robot is required to perform similar task i.e. realize similar effects using a T-shape tool. The T-shape tool is kept on table with the target object. The functional features of the T-shape tool are shown in Figure 4.1. Robot checks whether the requirements

Table 5.3: Explanation of learning to use stick-tool from human demonstration.

Target of learning	**Representation of what is learnt from human**	**Explanation**
Spatial Constraint	length(tool-handle, end-effector, L_{he}), distance(robot, target-object, d_{rt}), $L_{he} > d_{rt}$	The length of body connecting the handle of tool and end effector should be larger than the distance between robot and initial position of target object. Suitable length is required to reach the target object.
Grasp	hook-grip(Tool, Finger-palm)	Human uses *Finger-palm* contact with tool to provide hook grip with target object .
Functional Feature	geometry(Tool, flat-surface) length(Tool, L_s), length(target-object, L_t), $L_s > L_t$	The geometry used is flat surface. Length of contact surface of tool should be greater than the length of contact surface of object.
Placement of tool	distance(position-of-tool-surface, position-of-goal, d_{sg}), distance(position-of-target, position-of-goal, d_{tg}), $d_{sg} > d_{tg}$ touching(tool, target-object) angle(contact-surface-of-tool, α_c), angle(target-object, α_t), $\lvert\alpha_c\rvert = \lvert\alpha_t\rvert$ normal(tool-surface, $\vec{n}$), target-displacement(target-object, $\vec{r}$), $\vec{n} = c\ \vec{r}$	the contact surface should always be behind the the target object in order to provide a hook affordance The contact surface of tool and target object should touch. The contact surface of tool and target should have same orientation. The normal vector of the contact surface of tool should be parallel with the vector describing final target displacement.
Target manipulation	pull-with-tool(Tool, target-object)	Human uses **pull-with-tool** action to manipulate target from its initial position to the goal.
Task Space	relative-target-position(initial, final)	Effects are relative distance between initial and final object position

of spatial constraints and functional features are met (refer Table 5.3). Then, it selects the contact surface that meets the requirement for placement of tool. Both horizontal and vertical parts satisfy the condition of having a surface whose normal is aligned to the vector representing target displacement. After the placement, robot applies the similar **pull-with-tool**(tool, target-object) action to manipulate the target and achieve the desired effects.

Experiment 4: Robot manipulation of target object using Rakes to test its learning.



(a) Using horizontal part as functional feature

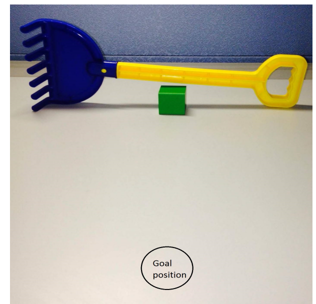
(b) Using vertical part as functional feature

Figure 5.5: Both horizontal and vertical parts can be used to manipulate the target object, as shown inside SIGVerse simulation environment.

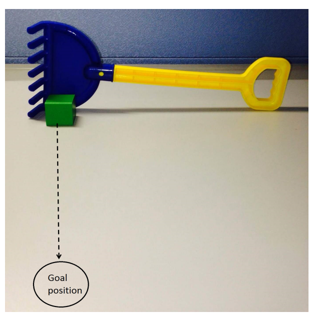
Robot manipulates the target object using **pull-with-tool**(tool, target-object) action after selected the suitable surface and placing the tool as shown in (a)-(d) of Figure 5.6.

In future, I would like to test this approach on generalization of tool-use model to a variety of unseen novel tools in order to realize novel effects. For example, after observing human move the object to its left using the flat surface of vertical stick, how to enable robot retrieve a distant object by pulling it with cup/egg head shaped sticks which have convex geometry. Also, I intend to make robot learn *functional features* using combination of computer vision, social interaction and social cues with the human user and extend the concept of *functional features* to the cases where *geometry* is not the only functionally relevant feature e.g. in case of hammer to crack the nut *stiffness* of the material is functional feature, in case of paper weight tool, *mass* would be the functional feature. The robot's capability to learn affordances for several actions and functional features using social interaction to acquire various inference capabilities is required to realize this goal of an autonomous robot tool user.

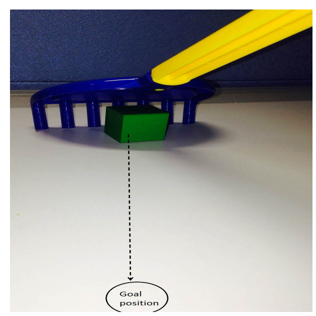
(a) Two candidate rakes to manipulate the target object. The smaller rake is not chosen since it does not satisfy the spatial constraints.



(b) The placement of tool fails because it does not satisfy the literal **touching**(tool, target-object)



(c) The shown contact surface is selected for target manipulation



(d) The shown contact surface is selected for target manipulation

Figure 5.6: Robot selects suitable contact surface for target manipulation, if it satisfies the requirements related to spatial constraints, functional features and placement of tool as mentioned in Table 5.3

Chapter 6

Appendix

6.1 Bayesian Network: A probabilistic graphical model

As Jordan[98] stated:

> *"Graphical models are a marriage between probability theory and graph theory. They provide a natural tool for dealing with two problems that occur throughout applied mathematics and engineering – uncertainty and complexity – and in particular they are playing an increasingly important role in the design and analysis of machine learning algorithms. Fundamental to the idea of a graphical model is the notion of modularity – a complex system is built by combining simpler parts. Probability theory provides the glue whereby the parts are combined, ensuring that the system as a whole is consistent, and providing ways to interface models to data. The graph theoretic side of graphical models provides both an intuitively appealing interface by which humans can model highly-interacting sets of variables as well as a data structure that lends itself naturally to the design of efficient general-purpose algorithms.*
>
> *Many of the classical multivariate probabalistic systems studied in fields such as statistics, systems engineering, information theory, pattern recog-*

nition and statistical mechanics are special cases of the general graphical model formalism – examples include mixture models, factor analysis, hidden Markov models, Kalman filters and Ising models. The graphical model framework provides a way to view all of these systems as instances of a common underlying formalism. This view has many advantages – in particular, specialized techniques that have been developed in one field can be transferred between research communities and exploited more widely. Moreover, the graphical model formalism provides a natural framework for the design of new systems."

Probabilistic graphical models (PGM) are graphs in which nodes represent random variables, and the (lack of) arcs represent conditional independence assumptions. The assumptions for conditional independence among variables enable a compact representation of joint probability distributions. For example, for n binary random variables $X_1, \ldots, X_n$, representing joint probability distribution will need $O(2^N)$ parameters, but thanks to the assumption of conditional independence, few variables can do the job.

There are two types of graphical models: 1) un-directed and, 2) directed. Bayesian Networks (also known as influence diagrams (Shachter[99]), causal probabilistic networks (Jensen et al.[100]), recursive graphical models (Lauritzen[101]), causal networks (Heckerman[102]), bayesian belief networks (Cheng et al.[103]), belief networks (Darwiche[104]) etc) is a *directed graphical model*.

For a better understanding of the Bayesian Networks, I will begin with a brief explanation of Bayes rule and the notion of conditional independence. Bayesian network as the name suggests is based on Bayes rule (Equation 6.1), where Pr denotes the probability while A and B are the random variables connected by a directed arc as shown in .

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B|A)\Pr(A) + \Pr(B|\neg A)\Pr(\neg A)} \tag{6.1}$$

A graph consists of vertices and links between them i.e $G = (V, A)$, where $V = \{v_1, \ldots, v_n\}$ be the set of vertices present in the graph and A represent finite set of arcs or edges. In a directed graph, all the arcs or edges have an associated direction from one node to another.

Each arc a = (u,v) can be defined either as an ordered or an unordered pair of nodes, which are said to be connected by and incident on the arc and to be adjacent to each other. Since they are adjacent, u and v are also said to be neighbors. If (u,v) is an ordered pair, u is said to be the tail of the arc and v the head; then the arc is said to be directed from u to v and is usually represented with an arrowhead in v ($u \rightarrow v$). It is also said that the arc leaves or is outgoing for u and that it enters or is incoming for v. If (u,v) is unordered, u and v are simply said to be incident on the arc without any further distinction. In this case, they are commonly referred to as undirected arcs or edges, denoted with $e \in E$ and represented with a simple line ($u \rightarrow v$).

The characterization of arcs as directed or undirected induces an equivalent characterization of the graphs themselves, which are said to be directed graphs (denoted with G = (V,A)) if all arcs are directed, undirected graphs (denoted with G = (V,E)) if all arcs are undirected, and partially directed or mixed graphs (denoted with G = (V,A,E)) if they contain both directed and undirected arcs.

Let the given set of random variables be $\mathbf{X} = X_1, X_2, ..., X_p$.

The **Markov property** of Bayesian networks, which follows directly from d-separation (Pearl[90]), enables the representation of the joint probability distribution of the random variables in $\mathbf{X}$ (the global distribution) as a product of conditional probability distributions (the local distributions associated with each variable X_i). This is a direct application of the *chain rule* (Korb and Nicholson[105]). In the case of discrete random variables, the factorization of the joint probability distribution P_X is given by

$$P_{\mathbf{X}}(\mathbf{X}) = \prod_{i=1}^{p} P_{X_i}(X_i \mid \Pi_{X_i}) \tag{6.2}$$

where Π_{X_i} is the set of the parents of X_i; in the case of continuous random variables, the factorization of the joint density function $f_{\mathbf{X}}$ is given by

$$f_{\mathbf{X}}(\mathbf{X}) = \prod_{i=1}^{p} f_{X_i}(X_i \mid \Pi_{X_i}) \tag{6.3}$$

Similar results hold for mixed probability distributions (i.e., probability distributions including both discrete and continuous random variables).

After we have the joint probability distribution P_X, the next task is to learn the Bayesian network. Learning denotes the task of fitting a Bayesian Network (Koller and Friedman[106]). It involves two steps:

1. Learning the structure for model selection.

2. Learning the underlying parameters of the global distribution of variables for the selected model.

The structure of the Bayesian Network can be built using the prior information available on the data and the knowledge of some human expert about the domain. But it can also be learned from the data itself and Several algorithms have been proposed for this task. They fall under three broad categories: *constraint-based*, *score-based*, and *hybrid algorithms*. Few names are mentioned for the well known algorithms for each of the three category for the readers reference.

- Constraint Based Structure Learning Algorithms: PC algorithm (Spirtes et al.[107]), Grow-Shrink (GS)(Margaritis[108]), Incremental Association (IAMB)(Tsamardinos et al.[109]), Fast Incremental Association (Fast-IAMB) (Yaramakala and Margaritis[110]), Interleaved Incremental Association (Inter-IAMB) (Tsamardinos et al.[111]) etc.

- Score Based Structure Learning Algorithms: Greedy search algorithms such as hill-climbing (Chickering[112], Bouckaert[113]), Genetic algorithms Larranaga et al[114], Simulated annealing (Bouckaert[113]) etc.

- Hybrid Structure Learning Algorithms: Sparse Candidate algorithm (SC) (Friedman et al[115]), Max-Min Hill-Climbing (MMHC) algorithm (Tsamardinos et al.[116]).

The second step is called *parameter learning.* The most commonly used algorithm known as maximum likelihood estimation (used in this thesis) is explained with proofs in next section.

6.1.1 Maximum Likelihood Parameter Estimation

I will start with some formal definitions. The n random variables that arise from a random sample are denoted with sub-scripted upper case letters as:

$$X_1, X_2, ..., X_n$$

The corresponding observed values of a specific random sample are denoted as subscripted lowercase letters:

$$x_1, x_2, ..., x_n$$

Now, we use the some "sample statistics" to summarize the data of the sample. These sample statistics are refereed as *parameters* of the data that arrive from sample. The range of possible values of the parameter θ is called the parameter space. This space is usually donated by Ω. For example, if μ denotes the mean of distances traveled by a number of mobile robots in a two-dimensional plane, then the parameter space is:

$$\Omega = \{\mu : 0 \leq \mu \leq 2\}$$

And, if p denotes the proportion of robots who reached their destination, then the parameter space is:

$$\Omega = \{p : 0 \leq p \leq 1\}$$

The **point estimator** of parameter θ is the function of $X_1, X_2, ...X_n$. This function is denoted here as the statistic:

$$\varrho(X_1, X_2, ...X_n)$$

For example, the function:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

is a point estimator of the population mean μ.

The function:

$$\hat{p} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

(where $X_i = 0, 1$) is a point estimator of the population proportion p measured using the data obtained from a random sample of the population.

And, the function:

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

is a point estimator of the population variance σ^2.

And the observed **point estimate** of parameter θ is the function $\varrho(x_1, x_2, ...x_n)$.

Now, with the above definitions, I will explain the method of maximum likelihood.

Statement of the Problem

Suppose we have a random sample $X_1, X_2, ...X_n$ whose assumed probability distribution depends on some unknown parameter θ. Now, given the observed values of the random sample $x_1, x_2, ...x_n$, our interest is in finding a point estimator i.e. the function $\varrho(X_1, X_2, ...X_n)$ so as to determine a *good* point estimate of θ i.e. $\varrho(x_1, x_2, ...x_n)$. For example, if we assume that all the X_i are normally distributed with mean μ and variance σ^2, then our goal is to determine a good estimate of μ and σ^2, using the observed data $x_1, x_2, ...x_n$.

Approach

A good estimate of the unknown parameter θ would be the value of θ that maximizes the likelihood of getting the data that we obtained from our specific random sample, observed. Thus maximizing the likelihood of unknown parameter is the basic idea behind this method.

To implement this method, let's assume that the probability density function for each X_i in our random sample $X_1, X_2, ...X_n$ is $f(x_i; \theta)$. Then, the joint probability density function of $X_1, X_2, ...X_n$ can be written as:

$$L(\theta) = P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)$$

where **likelihood function** $L(\theta)$ is written as a function of θ. Now, in a random sample all the X_i are independent from each other. So, the definition of the joint probability mass function can be written as:

$$L(\theta) = P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = f(x_1;\theta) \cdot f(x_2;\theta) \cdots f(x_n;\theta)$$

By taking the product of indexed terms, we can write:

$$L(\theta) = P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = f(x_1;\theta) \cdot f(x_2;\theta) \cdots f(x_n;\theta) = \prod_{i=1}^{n} f(x_i;\theta)$$

Formal definitions

Here I define the terms (1) likelihood function, (2) maximum likelihood estimators, and (3) maximum likelihood estimates.

Definition. Let $X_1, X_2, ...X_n$ be a random sample from a distribution. Let's assume that this distribution depends on some unknown parameters $\theta_1, \theta_2, \ldots, \theta_m$. Let the probability mass (or density) function be defined as $f(x_i; \theta_1, \theta_2, \ldots, \theta_m)$. Now, lets suppose that these m unknown parameters are constrained to a given parameter space. Let's denote this parameter space by Ω. Then:

1. The Likelihood function: We can write the joint probability density (or mass) function of our random sample $X_1, X_2, ...X_n$ as a function of the parameters $\theta_1, \theta_2, \ldots, \theta_m$ of the distribution that generated it, as:

$$L(\theta_1, \theta_2, \ldots, \theta_m) = \prod_{i=1}^{n} f(x_i; \theta_1, \theta_2, \ldots, \theta_m) \tag{6.4}$$

   Here, $\theta_1, \theta_2, \ldots, \theta_m \in \Omega$ is called the *likelihood function*.

2. Maximum Likelihood Estimator: If the m-tuple defined by the observations $x_1, x_2, ...x_n$ for our random sample can be written as:

$$[\varrho_1(x_1, x_2, ...x_n), \varrho_2(x_1, x_2, ...x_n), \ldots, \varrho_m(x_1, x_2, ...x_n)]. \tag{6.5}$$

   Then, the *maximum likelihood estimator* of θ_i for $i = 1, 2, \ldots, m$ can be written as:

$$\hat{\theta}_i = u_i(X_1, X_2, \ldots, X_n)$$

3. Maximum likelihood estimate: The observed values mentioned in Equation 6.5,

$$[u_1(x_1, x_2, \ldots, x_n), u_2(x_1, x_2, \ldots, x_n), \ldots, u_m(x_1, x_2, \ldots, x_n)]$$

   are called as the *maximum likelihood estimates* of θ_i for $i = 1, 2, \ldots, m$

Note: The one and only difference between the formulations for the estimator and the estimate is that:

- the likelihood estimator is define with the uppercase letters (to denote that its value is random), and
- the point estimate is defined using lowercase letters (to denote that its value is based on an obtained sample and hence is fixed)

Now, for the easier understanding of how to determine the value of θ that maximizes the likelihood function, I will provide here two examples. Example 1 considers the case when the observed values are discrete; considered to have boolean values just for the sake of simplicity. In example 2, I will explain the situation when the observed values are continuous.

Example 1:

Suppose we have a random sample $X_1, X_2, ...X_n$, where X_i are independent Bernoulli random variables with an unknown parameter p such that:

- $X_i = 0$: if a randomly selected mobile robot does not reach the destination, and
- $X_i = 1$: if a randomly selected mobile robot reaches the destination

The probability mass function of each X_i is:

$$f(x_i; p) = p^{x_i}(1-p)^{1-x_i}$$

, where $x_i = 0, 1$ and $0 < p < 1$.

Now, the **likelihood function** $L(p)$ written as joint probability mass function using the product of index terms as:

$$L(p) = \prod_{i=1}^{n} f(x_i; p) = p^{x_1}(1-p)^{1-x_1} \times p^{x_2}(1-p)^{1-x_2} \times \cdots \times p^{x_n}(1-p)^{1-x_n}$$

Upon simplification by summing up the exponents, we obtain:

$$L(p) = p^{\sum x_i}(1-p)^{n-\sum x_i}$$

Now, our objective is to determine some p that maximizes our likelihood function $L(p)$. To do that, rather than directly differentiating the likelihood function with respect to p, we first take its natural logarithm and then differentiate. It is done because natural logarithm is a monotonically increasing function of x. That means that the value of p that maximizes the natural logarithm of the likelihood function $log(L(p))$ is also the value of p that maximizes the likelihood function $L(p)$. Such a small manipulation is done to make differential bit easier. Now, the natural logarithm of the likelihood function is:

$$\log L(p) = (\sum x_i)\log(p) + (n - \sum x_i)\log(1-p)$$

We then take its derivative and set it to 0:

$$\frac{\partial log(L(p))}{\partial p} = \frac{\Sigma x_i}{p} - \frac{n - \Sigma x_i}{1-p} = 0$$

Now, multiplying through by p(1p), we get:

$$(\sum x_i)(1-p) - (n - \sum x_i)p = 0$$

Upon distributing few terms cancel out leaving us with:

$$\sum x_i - np = 0$$

Thus, we can obtain a point estimate as:

$$\hat{p} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Our estimator function would then be:

$$\hat{p} = \frac{\sum_{i=1}^{n} X_i}{n}$$

To verify whether the obtained estimate is maximum, we can take second derivative of log likelihood with respect to p and confirm that it is negative.

Example 2:

Suppose we have a random sample $X_1, X_2, ...X_n$, where X_i are drawn from a normal distribution with unknown parameters θ and variance σ^2. The Bayesian networks for this situation are called Gaussian Bayesian networks (Geiger and Heckerman[117], Neapolitan[118]).

To fine the estimator function and point estimates, the first step is to write down the join density function of X_i as a function of the parameters of the distribution that generated the sample. Thus, it can be written as:

$$f(x_i; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left[-\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$

The parameter space can be written as:

$$\Omega = (\mu, \sigma) : -\infty < \mu < \infty; 0 < \sigma < \infty$$

Therefore, the likelihood function for $-\infty < \mu < \infty$ and $0 < \sigma < \infty$ from Equation 6.4 is:

$$L(\mu, \sigma) = \sigma^{-n}(2\pi)^{-n/2}\exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\right]$$

To maximize the likelihood function with respect to parameters μ and σ, we need to take its derivative with respect to the parameters. To avoid the confusion over parameters, let's re-write the join density function of X_i as:

$$f(x_i; \theta_1, \theta_2) = \frac{1}{\sqrt{\theta_2}\sqrt{2\pi}}\exp\left[-\frac{(x_i-\theta_1)^2}{2\theta_2}\right]$$

where $\theta_1 = \mu$ and $\theta_2 = \sigma^2$, and the likelihood function as:

$$L(\theta_1, \theta_2) = \prod_{i=1}^{n} f(x_i; \theta_1, \theta_2) = \theta_2^{-n/2}(2\pi)^{-n/2}\text{exp}\left[-\frac{1}{2\theta_2}\sum_{i=1}^{n}(x_i - \theta_1)^2\right]$$

As explained in Example 1 above, the log of likelihood function is used to make the derivation easier. It can written as:

$$\text{log}L(\theta_1, \theta_2) = -\frac{n}{2}\text{log}\theta_2 - \frac{n}{2}\text{log}(2\pi) - \frac{\sum(x_i - \theta_1)^2}{2\theta_2}$$

I then take the partial derivative of the log of likelihood function with respect to parameter θ_1, and setting it to 0.

$$\frac{\partial log(L(\theta_1, \theta_2))}{\partial\theta_1} = \frac{-2\Sigma(x_i - \theta_i)(-1)}{2\theta_2} = 0$$

On solving, few terms cancel each other out. Upon multiplying both sides by θ_2 and then distributing the summation, we obtain:

$$\sum x_i - n\theta_1 = 0$$

Thus, upon maximizing the likelihood function with respect to $\theta_1 = \mu$, that the maximum likelihood estimator (see Equation 6.5 of μ is:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i = \bar{X}$$

And therefore, base on the given sample, a maximum likelihood estimate of μ is:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Now, let's solve for another parameter θ_2, where $\theta_2 = \sigma^2$. Taking the partial derivative of the log of likelihood function with respect to θ_2, and setting to 0, we obtain:

$$\frac{\partial log(L(\theta_1, \theta_2))}{\partial \theta_2} = \frac{-n}{2\theta_2} + \frac{\Sigma((x_i - \theta_i)^2)}{2\theta_2^2} = 0$$

Multiplying through by $2\theta_2^2$:

$$\frac{\partial log(L(\theta_1, \theta_2))}{\partial \theta_2} = \left[\frac{-n}{2\theta_2} + \frac{\Sigma((x_i - \theta_i)^2)}{2\theta_2^2} = 0\right] \times 2\theta_2^2$$

We obtain:

$$-n\theta_2 + \sum (x_i - \theta_1)^2 = 0$$

Upon solving for the parameter θ_2, we calculate that its maximum likelihood estimator function (Equation 6.5) is:

$$\hat{\sigma}^2 = \frac{\sum (X_i - \bar{X})^2}{n}$$

and its maximum likelihood estimate is:

$$\hat{\theta}_2 = \hat{\sigma}^2 = \frac{\sum (x_i - \bar{x})^2}{n}$$

Now, again to verify whether the obtained estimate is maximum, the second derivative of log of likelihood function with respect to the parameters μ and σ^2 can be taken and confirmed that it is negative.

6.1.2 Tools to learn the Bayesian Networks

There are several packages on CRAN (package repository for statistical computing using R) dealing with Bayesian networks. They can classified on the basis of the two core functionalities required to model the Bayesian Network namely structure learning and parameter learning. Some packages perform only one of the two functionality. I am mentioning some packages that perform both of them. These packages are **bnlearn** (Scutari[119,120], **deal** (Bttcher and Dethlefsen[121]), **pcalg** (Kalisch et al.[122]), and **catnet** (Balov and Salzman[123]). In particular, the **bnlearn** package site mentions that it offers a wide variety of structure learning algorithms, parameter learning approaches (maximum likelihood for discrete and continuous data, Bayesian estimation for discrete data), and inference techniques (cross-validation, bootstrap, conditional probability queries, and prediction).

6.2 SIGVerse: Socio IntelliGenesis simulator

Figure 6.1: The concept image of SIGVerse which appeared in Inamura et al.[5]

Sigverse is a simulation platform for social interaction between agents that include humans in rea world and robots in virtual world as shown in Figure 6.1(refer Inamura et al.[5] for an introduction). It enables seamless human-robot cooperation and collaboration in which humans can connect with the robots using cyberspace (Inamura[124]) and simulation the interaction (Tan and Inamura[125]). An agent inside SIGVerse can be a human avatar, a robot or some other entity can supports interactive behavior. The human(s) in real world can interact with virtual agents inside SIGVerse via peripheral devices meant for multi-modal interaction e.g. Microsoft Kinect sensor, Sony's PlayStation Move, Wii Motion Controller, JoyStick, Mouse etc. This makes it very easy and inexpensive to perform embodied and multi-modal human-robot interaction (Tan and Inamura[126,127]) even at a large scale in less time (Inamura and Tan[128]) using cloud based architecture (Tan and Inamura[129]). It is being used as official platform for Robocup competition that focus on HRI (Tan et al.[130,131]) and have found use in developing a chatter-bot system for supporting embodied interactions (Tan and Inamura[132,133]) and networked driving simulator (Takahasi and Shibata[134]).

I am using SIGVerse for simulating the manipulation of target via tool-use. The robot observations from these manipulation trials is then fed to the Bayesian Network to learn and evaluate the tool affordances as explained in Chapter 4. However, the main utilization of SIGVerse in my project is to use it for learning the tool-use model via interaction between humans and robots. The interaction based learning experiments are the candidate for future works as discussed in Section 5.2.

Bibliography

[1] W. H. Joanna Mcgrenere, "Affordances: Clarifying and evolving a concept," in *Proceedings of Graphics Interface 2000*, pp. 179–186, 2000.

[2] C. Baber, *Cognition and tool use: Forms of engagement in human and animal use of tools*. CRC Press, 2003.

[3] A. Stoytchev, *Robot Tool Behaviour: A Developmental Approach to Autonomous Tool Use*. PhD thesis, College of Computing, Georgia Institute of Technology., 2007.

[4] J. Sinapov and A. Stoytchev, "Detecting the functional similarities between tools using a hierarchical representation of outcomes," in *7th IEEE International Conference on Development and Learning.*, pp. 91 –96, 2008.

[5] T. Inamura, T. Shibata, H. Sena, T. Hashimoto, N. Kawai, T. Miyashita, Y. Sakurai, M. Shimizu, M. Otake, K. Hosoda, S. Umeda, K. Inui, and Y. Yoshikawa, "Simulator platform that enables social interaction simulation; sigverse: Sociointelligenesis simulator," in *IEEE/SICE International Symposium on System Integration (SII)*, pp. 212–217, Dec 2010.

[6] R. S. Amant and A. B. Wood, "Tool use for autonomous agents.," in *AAAI*, pp. 184–189, 2005.

[7] S. T. Parker and K. R. Gibson, "Object manipulation, tool use and sensorimotor intelligence as feeding adaptations in cebus monkeys and great apes," *Journal of Human Evolution*, vol. 6, no. 7, pp. 623–641, 1977.

[8] A. Seed and R. Byrne, "Animal tool-use," *Current biology*, vol. 20, no. 23, pp. R1032–R1039, 2010.

[9] G. R. Hunt, "Manufacture and use of hook-tools by new caledonian crows," *Nature*, vol. 379, no. 6562, pp. 249–251, 1996.

[10] N. Inoue-Nakamura and T. Matsuzawa, "Development of stone tool use by wild chimpanzees (pan troglodytes).," *Journal of Comparative Psychology*, vol. 111, no. 2, p. 159, 1997.

[11] J. Van Lawick-Goodall, "Tool-using in primates and other vertebrates," *Advances in the Study of Behavior*, vol. 3, pp. 195–249, 1970.

[12] J. Alcock, "The evolution of the use of tools by feeding animals," *Evolution*, pp. 464–473, 1972.

[13] B. B. Beck, "Animal tool behavior: The use and manufacture of tools by animals," New York: Garland STMP Press, 1980.

[14] S. Chevalier-Skolnikoff, "Tool use by wild cebus monkeys at santa rosa national park, costa rica," *Primates*, vol. 31, no. 3, pp. 375–383, 1990.

[15] B. Preston, "Cognition and tool use," *Mind & Language*, vol. 13, no. 4, pp. 513–547, 1998.

[16] D. Lestel and E. Grundmann, "Tools, techniques and animals: the role of mediations of actions in the dynamics of social behaviours," *Social science information*, vol. 38, no. 3, pp. 367–407, 1999.

[17] T. Matsuzawa, *Primate foundations of human intelligence: a view of tool use in nonhuman primates and fossil hominids*. Springer, 2001.

[18] N. P. Holmes and C. Spence, "The body schema and multisensory representation (s) of peripersonal space," *Cognitive processing*, vol. 5, no. 2, pp. 94–105, 2004.

[19] V. K. Bentley-Condit and E. Smith, "Animal tool use: current definitions and an updated comprehensive catalog," *Behaviour*, vol. 147, no. 2, p. 185, 2010.

[20] C. Lovchik and M. A. Diftler, "The robonaut hand: A dexterous robot hand for space," in *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 907–912, 1999.

[21] W. Bluethmann, R. Ambrose, M. Diftler, S. Askew, E. Huber, M. Goza, F. Rehnmark, C. Lovchik, and D. Magruder, "Robonaut: A robot designed to work with humans in space," *Autonomous Robots*, vol. 14, pp. 179–197, 2003.

[22] D. Whitney, C. Lozinski, and J. M. Rourke, "Industrial robot forward calibration method and results," *Journal of dynamic systems, measurement, and control*, vol. 108, no. 1, pp. 1–8, 1986.

[23] R. Hara, T. Mizuno, S. Sakakibara, and K. Toyoda, "Industrial robot circular arc control method for controlling the angle of a tool," Oct. 6 1987. US Patent 4,698,777.

[24] Y. Takeuchi, N. Asakawa, and D. Ge, "Automation of polishing work by an industrial robot: system of polishing robot," *JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing*, vol. 36, no. 4, pp. 556–561, 1993.

[25] P. Dario, E. Guglielmelli, B. Allotta, and M. C. Carrozza, "Robotics for medical applications," *Robotics & Automation Magazine, IEEE*, vol. 3, no. 3, pp. 44–56, 1996.

[26] G. Guthart and J. Salisbury, J., "The intuitivetm telesurgery system: overview and application," in *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 618–621 vol.1, 2000.

[27] M. J. Tierney, T. G. Cooper, C. A. Julian, S. J. Blumenkranz, G. S. Guthart, and R. G. Younge, "Surgical robotic tools, data architecture, and use," Dec. 18 2001. US Patent 6,331,181.

[28] A. D. Christiansen, M. T. Mason, and T. M. Mitchell, "Learning reliable manipulation strategies without initial physical models.," *Robotics and Autonomous Systems*, vol. 8, no. 1-2, pp. 7–18, 1991.

[29] T. Zrimec and P. Mowforth, "Toward learning robots," ch. Learning by an Autonomous Agent in the Pushing Domain, pp. 19–29, Cambridge, MA, USA: MIT Press, 1993.

[30] K. Lynch, "Estimating the friction parameters of pushed objects," in *In Proc. of the 1993 IEEE/RSJ Int'l Conference on Intelligent Robots and Systems*, pp. 186–193, 1993.

[31] T. Yoshikawa and M. Kurisu, "Indentification of the center of friction from pushing an object by a mobile robot," in *Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on*, pp. 449–454 vol.2, Nov 1991.

[32] M. Salganicoff, G. Metta, A. Oddera, G. Sandini, M. Salganico, G. Metta, A. Oddera, and G. Sandini, "A vision-based learning method for pushing manipulation," in *In AAAI Fall Symposium Series: Machine Learning in Vision: What Why and*, 1993.

[33] M. Salganicoff, L. Ungar, and R. Bajcsy, "Active learning for vision-based robot grasping," *Machine Learning*, vol. 23, no. 2-3, pp. 251–278, 1996.

[34] O. Fuentes and R. C. Nelson, "Learning dextrous manipulation skills for multifingered robot hands using the evolution strategy," *Autonomous Robots*, vol. 5, no. 3-4, pp. 395–405, 1998.

[35] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, "Learning about objects through action-initial steps towards artificial cognition," in *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 3140–3145, 2003.

[36] D. Katz, Y. Pyuro, and O. Brock, "Learning to manipulate articulated objects in unstructured environments using a grounded relational representation."

[37] L. Bogoni, "Identification of functional features through observations and interactions," 1995.

[38] A. Stoytchev, "Behavior-grounded representation of tool affordances," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3071–3076, 2005.

[39] A. Stoytchev, "Learning the affordances of tools using a behavior-grounded approach," in *Proceedings of the 2006 international conference on Towards affordance-based robot control*, (Berlin, Heidelberg), pp. 140–158, Springer-Verlag, 2008.

[40] J. J. Gibson, *The Senses Considered as Perceptual Systems.* Houghton Mifflin Company, 1966.

[41] J. J. Gibson, *The ecological approach to visual perception.* Psychology Press, 2013.

[42] S. Nishide, J. Tani, T. Takahashi, H. G. Okuno, and T. Ogata, "Tool-body assimilation of humanoid robot using a neurodynamical system," *IEEE T. Autonomous Mental Development*, pp. 139–149, 2012.

[43] C. Nabeshima, Y. Kuniyoshi, and M. Lungarella, "Adaptive body schema for robotic tool-use," *Advanced Robotics*, pp. 1105–1126, 2006.

[44] C. Nabeshima, Y. Kuniyoshi, and M. Lungarella, "Towards a model for tool-body assimilation and adaptive tool-use," in *IEEE 6th International Conference on Development and Learning*, pp. 288 –293, 2007.

[45] C. C. Kemp, "Robot manipulation of human tools: Autonomous detection and control of task relevant features," in *5th IEEE International Conference on Development and Learning*, 2006.

[46] R. Radwin, J. Haney, and A. I. H. A. E. Committee, *An Ergonomics Guide to Hand Tools.* Ergonomics Guide Series, American Industrial Hygiene Association, 1996.

[47] S. Brown, *A relational approach to tool-use learning in robots.* PhD thesis, School of Computer Science and Engineering, The University of New South Wales, 2009.

[48] S. Muggleton and C. Feng, "Efficient induction of logic programs," in *New Generation Computing*, Academic Press, 1990.

[49] T. Nakamura and T. Nagai, "Object concept modeling based on the relationship among appearance, usage and functions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5410–5415, Oct 2010.

[50] D. G. Lowe, "Object recognition from local scale-invariant features," in *The proceedings of the seventh IEEE international conference on Computer vision*, vol. 2, pp. 1150–1157, 1999.

[51] H. Attias, "Inferring parameters and structure of latent variable models by variational bayes," in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, (San Francisco, CA, USA), pp. 21–30, Morgan Kaufmann Publishers Inc., 1999.

[52] F. Guerin, N. Kruger, and D. Kraft, "A survey of the ontogeny of tool use: from sensorimotor experience to planning," *IEEE Transactions on Autonomous Mental Development*, vol. 5, no. 1, pp. 18–45, 2013.

[53] A. Chemero, "An outline of a theory of affordances," *Ecological psychology*, vol. 15, no. 2, pp. 181–195, 2003.

[54] A. Chemero and M. T. Turvey, "Gibsonian affordances for roboticists," *Adaptive Behavior*, vol. 15, no. 4, pp. 473–480, 2007.

[55] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: A new formalization of affordances toward affordance-based robot control," *Adaptive Behavior*, vol. 15, no. 4, pp. 447–472, 2007.

[56] R. R. Murphy, "Case studies of applying gibson's ecological approach to mobile robots," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 29, no. 1, pp. 105–111, 1999.

[57] M. Cakmak, M. Dogar, E. Ugur, and E. Sahin, "Affordances as a framework for robot control," in *Proceedings of the 7th international conference on epigenetic robotics EpiRob'07*, 2007.

[58] L. Paletta, G. Fritz, F. Kintzler, J. Irran, and G. Dorffner, "Learning to perceive affordances in a framework of developmental embodied cognition," in *IEEE 6th International Conference on Development and Learning*, pp. 110–115, 2007.

[59] O. Brock, A. Fagg, R. Grupen, R. Platt, M. Rosenstein, and J. Sweeney, "A framework for learning and control in intelligent humanoid robots," *International Journal of Humanoid Robotics*, vol. 2, no. 03, pp. 301–336, 2005.

[60] E. Ugur, M. R. Dogar, M. Cakmak, and E. Sahin, "The learning and use of traversability affordance using range images on a mobile robot," in *IEEE International Conference on Robotics and Automation*, pp. 1721–1726, 2007.

[61] E. Erdemir, C. B. Frankel, K. Kawamura, S. M. Gordon, S. Thornton, and B. Ulutas, "Towards a cognitive robot that uses internal rehearsal to learn affordance relations," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2016–2021, 2008.

[62] E. Erdemir, C. B. Frankel, S. Thornton, B. Ulutas, and K. Kawamura, "A robot rehearses internally and learns an affordance relation," in *7th IEEE International Conference on Development and Learning*, pp. 298–303, 2008.

[63] L. Montesano and M. Lopes, "Learning grasping affordances from local visual descriptors," in *IEEE 8th International Conference on Development and Learning*, pp. 1–6, 2009.

[64] R. Detry, E. Baseski, M. Popovic, Y. Touati, N. Kruger, O. Kroemer, J. Peters, and J. Piater, "Learning object-specific grasp affordance densities," in *IEEE 8th International Conference on Development and Learning*, pp. 1–7, 2009.

[65] C. de Granville, J. Southerland, and A. H. Fagg, "Learning grasp affordances through human demonstration," in *Proceedings of the International Conference on Development and Learning*, 2006.

[66] R. Detry, D. Kraft, A. G. Buch, N. Kruger, and J. Piater, "Refining grasp affordance models by experience," in *IEEE International conference on Robotics and automation*, pp. 2287–2293, 2010.

[67] I. Cos-Aguilera, L. Canamero, and G. Hayes, "Learning object functionalities in the context of behaviour selection," in *Proceedings of the 4th British Conference Towards Intelligent Autonomous Robots*, 2003.

[68] I. Cos-Aguilera, G. Hayes, and L. Cañamero, "Using a sofm to learn object affordances," in *Procs 5th Workshop of Physical Agents (WAF'04)*, University of Edinburgh, 2004.

[69] I. Cos-Aguilera, L. Canamero, and G. M. Hayes, "Motivation-driven learning of object affordances: First experiments using a simulated khepera robot," in *In Proceedings of the 9th International Conference in Cognitive Modelling*, pp. 57–62, 2003.

[70] D. Kraft, R. Detry, N. Pugeault, E. Başeski, J. Piater, and N. Krüger, "Learning objects and grasp affordances through autonomous exploration," in *Computer Vision Systems*, pp. 235–244, Springer, 2009.

[71] J. D. Sweeney and R. Grupen, "A model of shared grasp affordances from demonstration," in *7th IEEE-RAS International Conference on Humanoid Robots*, pp. 27–35, 2007.

[72] B. Moldovan, P. Moreno, M. van Otterlo, J. Santos-Victor, and L. De Raedt, "Learning relational affordance models for robots in multi-object manipulation tasks," in *IEEE International Conference on Robotics and Automation*, pp. 4373–4378, 2012.

[73] J. Sun, J. L. Moore, A. Bobick, and J. M. Rehg, "Learning visual object categories for robot affordance prediction," *The International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 174–197, 2010.

[74] N. Dag, I. Atil, S. Kalkan, and E. Sahin, "Learning affordances for categorizing objects and their properties," in *20th IEEE International Conference on Pattern Recognition*, pp. 3089–3092, 2010.

[75] G. Fritz, L. Paletta, R. Breithaupt, E. Rome, and G. Dorffner, "Learning predictive features in affordance based robotic perception systems.," in *IROS*, pp. 3642–3647, 2006.

[76] A. B. Wood, T. E. Horton, and R. S. Amant, "Effective tool use in a habile agent," in *IEEE Systems and Information Engineering Design Symposium*, pp. 75–81, 2005.

[77] J. Sinapov and A. Stoytchev, "Learning and generalization of behavior-grounded tool affordances," in *IEEE 6th International Conference on Development and Learning*, pp. 19–24, 2007.

[78] T. E. Horton and R. S. Adviser-Amant, *A partial contour similarity-based approach to visual affordances in habile agents.* North Carolina State University, 2011.

[79] T. E. Horton, L. Williams, W. Mu, and R. S. Amant, "Visual affordances and symmetries in canis habilis: A progress report.," in *AAAI Fall Symposium: Naturally-Inspired Artificial Intelligence*, pp. 72–78, 2008.

[80] D. A. Norman, *The Design of Everyday Things*. New York: Basic Books, reprint paperback ed., 2002.

[81] R. J. Sternberg, "Cognitive psychology. 2nd. ed," 1996.

[82] D. A. Norman, *Turn signals are the facial expressions of automobiles.* Basic Books, 1992.

[83] D. A. Norman, "Things that make us smart," 1993.

[84] D. A. Norman, *The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution*. MIT press, 1998.

[85] W. W. Gaver, "Technology affordances," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, (New York, NY, USA), pp. 79–84, ACM, 1991.

[86] S. Grafton and A. Hamilton, "Evidence for a distributed hierarchy of action representation in the brain.," *Hum Mov Sci*, vol. 26, no. 4, pp. 590–616, 2007.

[87] A. Martin, J. Haxby, F. Lalonde, C. Wiggs, and L. Ungerleider, "Discrete cortical regions associated with knowledge of color and knowledge of action.," *Science*, vol. 270, no. 5233, pp. 102–105, 1995.

[88] A. Martin, C. Wiggs, L. Ungerleider, and J. Haxby, "Neural correlates of category-specific knowledge.," *Nature*, vol. 379, no. 6566, pp. 649–652, 1996.

[89] E. Koechlin and T. Jubault, "Broca's area and the hierarchical organization of human behavior," *Neuron*, vol. 50, no. 6, pp. 963 – 974, 2006.

[90] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998.

[91] J. Pearl(1998), *Causality: models, reasoning, and inference.* New York, NY, USA: Cambridge University Press.

[92] D. Heckerman, "A tutorial on learning with bayesian networks," tech. rep., Learning in Graphical Models, 1996.

[93] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[94] C. Huang and A. Darwiche, "Inference in belief networks: A procedural guide," *International Journal of Approximate Reasoning*, vol. 15, pp. 225–263, 1996.

[95] R. Jain and T. Inamura, "Learning of tool affordances for autonomous tool manipulation," in *IEEE/SICE International Symposium on System Integration*, pp. 814 –819, 2011.

[96] R. Jain and T. Inamura, "Estimation of suitable action of tool manipulation to realize target effects using tool affordances representation," in *The 30th Annual Conference of the Robotics Society of Japan, Sapporo, Japan*, 2012.

[97] R. Jain and T. Inamura, "Bayesian learning of tool affordances based on generalization of functional feature to estimate effects of unseen tools," *Artificial Life and Robotics*, vol. 18, no. 1-2, pp. 95–103, 2013.

[98] M. I. Jordan, ed., *Learning in Graphical Models*. Cambridge, MA, USA: MIT Press, 1999.

[99] R. D. Shachter, "Evaluating Influence Diagrams," *Operations Research*, vol. 33, no. 6, 1986.

[100] F. V. Jensen, K. G. Olesen, and S. K. Andersen, "An algebra of bayesian belief universes for knowledge-based systems.," *Networks*, vol. 20, no. 5, pp. 637–659.

[101] S. L. Lauritzen, "The em algorithm for graphical association models with missing data," *Comput. Stat. Data Anal.*, vol. 19, pp. 191–201, Feb. 1995.

[102] D. Heckerman, "A bayesian approach to learning causal networks," in *In Uncertainty in AI: Proceedings of the Eleventh Conference*, pp. 285–295, 1995.

[103] J. Cheng, D. A. Bell, and W. Liu, "Learning belief networks from data: An information theory based approach," in *Proceedings of the Sixth International Conference on Information and Knowledge Management*, CIKM '97, (New York, NY, USA), pp. 325–331, ACM, 1997.

[104] A. Darwiche, "A logical approach to factoring belief networks," 2002.

[105] K. Korb and A. Nicholson, *Bayesian Artificial Intelligence*. Chapman and Hall, 2nd ed., 2010.

[106] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[107] P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search*. Cambridge, MA, USA: The MIT Press, second ed., Jan. 2001.

[108] D. Margaritis, *Learning Bayesian Network Model Structure from Data*. PhD thesis, School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, May 2003. Available as Technical Report CMU-CS-03-153.

[109] I. Tsamardinos, C. F. Aliferis, and A. Statnikov, "Algorithms for Large Scale Markov Blanket Discovery," in *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, pp. 376–381, AAAI Press, 2003.

[110] S. Yaramakala and D. Margaritis, "Speculative Markov Blanket Discovery for Optimal Feature Selection," in *Proceedings of the Fifth IEEE International Conference on Data Mining*, pp. 809–812, IEEE Computer Society, 2005.

[111] I. Tsamardinos, C. F. Aliferis, and A. Statnikov, "Time and Sample Efficient Discovery of Markov Blankets and Direct Causal Relations," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 673–678, ACM, 2003.

[112] D. M. Chickering, "A Transformational Characterization of Equivalent Bayesian Network Structures," in *UAI '95: Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, pp. 87–98, Morgan Kaufmann, 1995.

[113] R. Bouckaert, *Bayesian Belief Networks: from Construction to Inference*. PhD thesis, Utrecht, Netherlands, 1995.

[114] P. Larranaga, B. Sierra, M. J. Gallego, M. J. Michelena, and J. M. Picaza, "Learning bayesian networks by genetic algorithms. a case study in the prediction of survival in malignant skin melanoma," 1997.

[115] N. Friedman, M. Goldszmidt, and A. Wyner, "Data Analysis with Bayesian Networks: A Bootstrap Approach," in *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 196–205, Morgan Kaufmann, 1999.

[116] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm," *Machine Learning*, vol. 65, no. 1, pp. 31–78, 2006.

[117] D. Geiger and D. Heckerman, "Learning Gaussian Networks," tech. rep., Microsoft Research, 1994. Available as Technical Report MSR-TR-94-10.

[118] R. E. Neapolitan, *Learning Bayesian Networks.* Prentice Hall, 2003.

[119] R. Nagarajan, M. Scutari, and S. Lèbre, *Bayesian Networks in R with Applications in Systems Biology.* Use R! series, Springer, 2013.

[120] M. Scutari, "Structure Variability in Bayesian Networks," *ArXiv Statistics - Methodology e-prints*, 2009.

[121] S. G. Boettcher and C. Dethlefsen, "deal: A package for learning bayesian networks," *Journal of Statistical Software*, vol. 8, pp. 1–40, 12 2003.

[122] M. Kalisch, M. Mchler, D. Colombo, M. H. Maathuis, and P. Bhlmann, "Causal inference using graphical models with the r package pcalg," *Journal of Statistical Software*, vol. 47, pp. 1–26, 5 2012.

[123] N. Balov and P. Salzman, "How to use catnet package," 2010.

[124] T. Inamura, "Human-robot cooperation system using shared cyber space that connects to real world - development of sociointelligenesis simulator sigverse toward hri," in *Proceedings of 1st International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pp. 429–434, 2011.

[125] J. T. C. Tan and T. Inamura, "What are required to simulate interaction with robot? sigverse - A simulation platform for human-robot interaction," in *IEEE International Conference on Robotics and Biomimetics*, pp. 2878–2883, 2011.

[126] J. T. C. Tan and T. Inamura, "Embodied and multimodal human-robot interaction between virtual and real worlds," in *IEEE International Symposium on Robot and Human Interactive Communication*, pp. 296–297, 2013.

[127] J. T. C. Tan and T. Inamura, "Integration of work sequence and embodied interaction for collaborative work based human-robot interaction," in *ACM/IEEE International Conference on Human-Robot Interaction*, pp. 239–240, 2013.

[128] T. Inamura and J. T. C. Tan, "Development of robocup @home simulator: simulation platform that enables long-term large scale HRI," in *ACM/IEEE International Conference on Human-Robot Interaction*, pp. 145–146, 2013.

[129] J. T. C. Tan and T. Inamura, "Sigverse - A cloud computing architecture simulation platform for social human-robot interaction," in *IEEE International Conference on Robotics and Automation*, pp. 1310–1315, 2012.

[130] J. T. C. Tan, T. Inamura, Y. Hagiwara, K. Sugiura, T. Nagai, and H. Okada, "A new dimension for robocup @home: human-robot interaction between virtual and real worlds," in *ACM/IEEE International Conference on Human-Robot Interaction*, p. 332, 2014.

[131] J. T. C. Tan, T. Inamura, K. Sugiura, T. Nagai, and H. Okada, "Human-robot interaction between virtual and real worlds: Motivation from robocup @home," in *Social Robotics - 5th International Conference*, pp. 239–248, 2013.

[132] J. T. C. Tan and T. Inamura, "Extending chatterbot system into multimodal interaction framework with embodied contextual understanding," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pp. 251–252, 2012.

[133] J. T. C. Tan, F. Duan, and T. Inamura, "Multimodal human-robot interaction with chatterbot system: Extending aiml towards supporting embodied interactions," in *IEEE International Conference on Robotics and Biomimetics*, pp. 1727–1732, 2012.

[134] T. Bando and T. Shibata, "Networked driving simulator based on sigverse and lane-change analysis according to frequency of driving," in *15th International IEEE Conference on Intelligent Transportation Systems*, pp. 1608–1613, 2012.