THE EFFECTS OF VISUAL AFFORDANCES AND FEEDBACK ON A GESTURE-BASED INTERACTION WITH NOVICE USERS

A Dissertation

by

CARLOS XAVIER SOTO

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Robin R. Murphy
Committee Members,	Ricardo Gutierrez-Osuna
	Andruid Kerne
	Takashi Yamauchi
Head of Department,	Dilma Da Silva

December 2017

Major Subject: Computer Engineering

ABSTRACT

This dissertation studies the roles and effects of visual affordances and feedback in a general-purpose gesture interface for novice users. Gesture interfaces are popularly viewed as intuitive and user-friendly modes of interacting with computers and robots, but they in fact introduce many challenges for users not already familiar with the system. Affordances and feedback – two fundamental building blocks of interface design – are perfectly suited to address the most important challenges and questions for novices using a gesture interface: *what can they do? how do they do it? are they being understood? has anything gone wrong?* Yet gesture interfaces rarely incorporate these features in a deliberate manner, and there are presently no well-adopted guidelines for designing affordances and feedback for gesture interaction, nor any clear understanding of their effects on such an interaction.

A general-purpose gesture interaction system was developed based on a virtual touchscreen paradigm, and guided by a novel gesture interaction framework. This framework clarifies the relationship between gesture interfaces and the application interfaces they support, and it provides guidance for selecting and designing appropriate affordances and feedback. Using this gesture system, a 40-person (all novices) user study was conducted to evaluate the effects on interaction performance and user satisfaction of four categories of affordances and feedback. The experimental results demonstrated that affordances indicating how to do something in a gesture interaction are more important to interaction performance than affordances indicating what can be done, and also that

system status is more important than feedback acknowledging user actions. However, the experiments also showed unexpectedly high interaction performance when affordances and feedback were omitted. The explanation for this result remains an open question, though several potential causes are analyzed, and a tentative interpretation is provided.

The main contributions of this dissertation to the HRI and HCI research communities are 1) the design of a virtual touchscreen-based interface for general-purpose gesture interaction, to serve as a case study for identifying and designing affordances and feedback for gesture interfaces; 2) the method and surprising results of an evaluation of *distinct* affordance and feedback categories, in particular their effects on a gesture interaction with novice users; and 3) a set of guidelines and insights about the relationship between a user, a gesture interface, and a generic application interface, centered on a novel interaction framework that may be used to design and study other gesture systems. In addition to the intellectual contributions, this work is useful to the general public because it may influence how future assistive robots are designed to interact with people in various settings including search and rescue, healthcare and elderly care.

DEDICATION

To my mother, Ofelia, who endured uncommon hardships and made untold sacrifices so that I may have the opportunities she never did. And to my step-father, Javier, for loving and supporting her as she deserves.

Gracias

ACKNOWLEDGEMENTS

I would like to first thank my advisor and committee chair, Dr. Robin R. Murphy, and my graduate committee members, Drs. Ricardo Gutierrez-Osuna, Anduid Kerne, and Takashi Yamauchi for their guidance and feedback throughout the course of this research. Special thanks is due in particular to Dr. Murphy, who has for eight years now helped me navigate the world of research, opened doors of opportunity to me, and continually urged me to succeed.

Second, I wish to thank the staff, leadership, and patrons of the scholarships, fellowships, and research programs that made it possible for me to pursue my undergraduate and graduate degrees and enriched them with further opportunities to grow. Texas A&M's Regents' Scholars Association, Association of Former Students, Prospective Student Center, the Department of Computer Science and Engineering's Industrial Affiliates Program, and the MG James Ursano Scholarship program allowed me to pursue my baccalaureate in computer engineering, which would have been immeasurably out of my reach without their financial support. The Dwight Look College of Engineering's Undergraduate Student Research Grant and the National Science Foundation's Louis Stokes Alliance for Minority Participation introduced me to research as an undergraduate and brought me into Dr. Murphy's research lab. As a graduate student, I was fortunate enough to be a part of the National Science Foundation's Bridge to Doctorate Fellowship and the Dwight Look College of Engineering's Graduate Teaching

Fellows Program, which helped fund my research and provided me with invaluable teaching experience. To all these people and programs: thank you.

Thanks also to my labmates, classmates and professors, present and past. The long road here has been filled with lessons, advice, friendships, and memories, and I’m grateful to all of you for being a part of it.

Finally, thanks most of all to my wife, confidant, and dearest friend... Irene, your patience and support made this achievement possible, and your love and respect made it worth it.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a dissertation committee consisting of Drs. Robin R. Murphy, Ricardo Gutierrez-Osuna, and Andruid Kerne of the Department of Computer Science and Engineering and Dr. Takashi Yamauchi of the Department of Psychology.

All work for the dissertation was completed by the student, under the advisement of Robin R. Murphy of the Department of Computer Science and Engineering.

Funding Sources

This research and precursors to it were supported in part by the Bridge to Doctorate Fellowship from the National Science Foundation, and by a Graduate Teaching Fellowship from the Dwight Look College of Engineering.

NOMENCLATURE

HCI	Human-computer interaction
HRI	Human-robot interaction
UI	User interface
PC	Personal computer

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Computational gesture recognition has been studied and used in robotics and human-computer interaction for decades [2, 41, 79], and is considered a more intuitive and natural means of interaction than traditional computer interfaces like keyboards and mice [2, 71]. Together with natural language processing and touch input devices, gesture interfaces (especially those for hand gestures) are enabling people to interact with machines in much the same way as they interact with other people and everyday objects.

Interfaces that support hand gesture recognition have been sought and developed for a wealth of applications in research, industrial, entertainment, and education domains [49, 59, 71]. Often, gesture recognition is pursued to make an interaction more interesting or immersive (as in entertainment applications like video games). But in other cases, gesture recognition is used because an application or domain requires a level of interaction simplicity not fulfilled by traditional interfaces. One such domain is assistive robots for use in healthcare, elderly care and search and rescue [71]. There and elsewhere, gesture interaction is sought for its expected ease-of-use. Appendix A provides a comparison of gesture interaction to other natural user interfaces for assistive robots.

However, there remain important challenges and questions surrounding the use of gesture interfaces for computers and robots of any sort, especially when the expected users are novices. Gesturing is an imperfect communication medium, and certainly there are misunderstandings when people gesture to one another – subtle body language may go unrecognized, or culturally-specific gestures may be misinterpreted – but the challenges that arise when gesturing at a machine are more fundamental in nature, as the following motivation demonstrates.

1.1. A Motivating Example

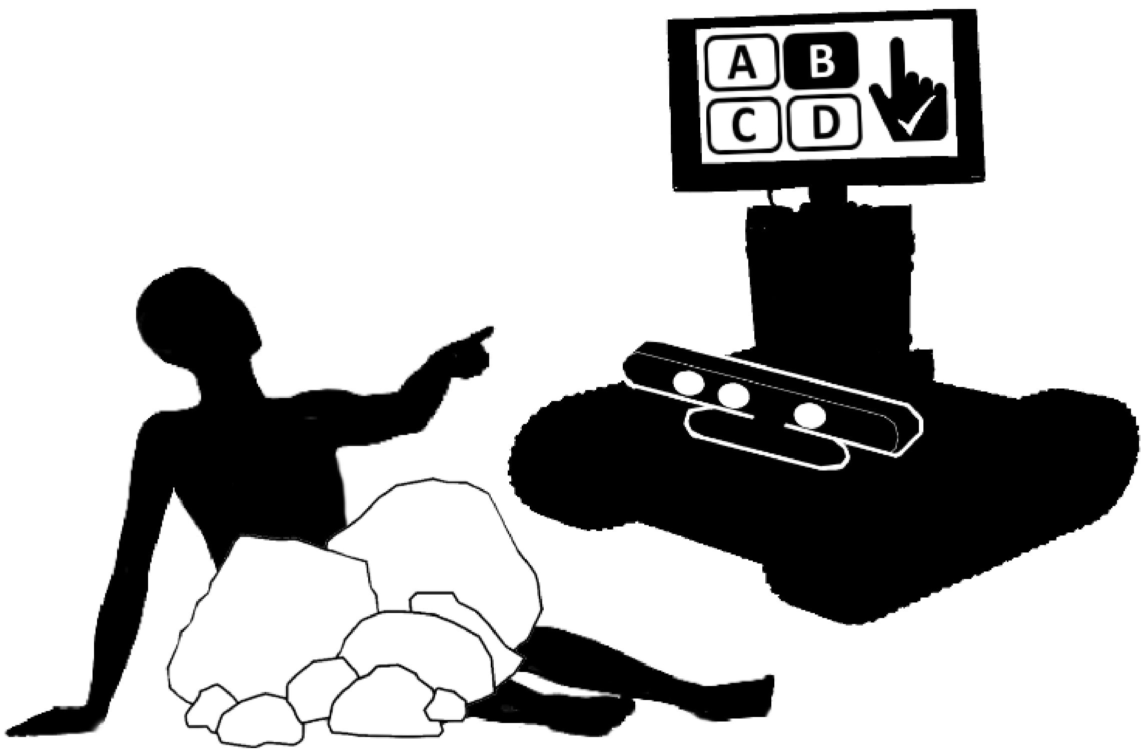


Figure 1. An example interaction scenario between a trapped victim and an assistive robot. Although the robot supports gesture recognition, the victim has no prior experience with it, and no opportunity to train or practice with the system in advance. It is up to the interface then to guide the user and provide useful feedback.

Consider the following scenario, shown in Figure 1. Search and rescue teams respond to a building collapse with search dogs and rescue robots to locate victims trapped in the rubble. One of these rescue robots encounters a person who is trapped in a precarious void, and responders determine it will take several hours to safely extract them without risking a secondary collapse, so the robot remains with the victim to keep them company and act as a communication link to the outside world. Fortunately, this robot was designed for just such a situation: it has cameras and microphones to see and hear the victim, as well as speakers and a display to respond and present information to them. Although the noise throughout the rescue site makes it difficult for the robot and victim to clearly hear one another, the robot is luckily also equipped with a sensor to detect and recognize hand gestures.

An important – and often overlooked – issue with this setup is that the victim may never before have interacted with a robot nor used a gesture interface. How would they know what to do, and how to do it? How would they know what the robot can do, and what it cannot? Will they recognize when something has gone wrong? Will they be able to help the system recover from those errors? Bear in mind that not only is this victim likely to have never interacted with such a system before, but there is also no opportunity for them to train or practice with it. So they can't be properly "taught" how to user the interface; their first contact with it is "in the field," so to speak.

In this scenario, having favorable answers to these questions could be the difference between keeping the victim calm and cooperative with the responders, and frustrating or panicking them further than they already were. But the questions – and the underlying challenges posed by novice users – apply to any gesture interface. If a novice user is to successfully use a gesture interface to interact with a computer or robot, they will need some help from the interface itself.

1.2. Challenges and Ideals for Gesture Interaction with Novices

Novice users present unique challenges for many tasks and applications, and this section discusses three particular challenges with respect to their using a gesture interface, as well as ideals to aim for in designing solutions for those challenges. These challenges and ideals that helped shape the requirements and design for the interface described later, as well as the research questions answered in my experiments. Notably, these challenges are not specific to the use of gesture interfaces with assistive robots (nor robots of any sort), so the contributions and findings of this work are applicable to gesture interaction in general. Also, note that there are certainly other challenges worth considering: system failures, gesture fatigue (i.e. the so-called "gorilla arm syndrome"), the limitations of communicating via gestures, and so on. However, these particular challenges are fundamental to the very nature of gesture interaction, and generalize to different implementations and applications.

The first challenge, which may be called the "bootstrapping challenge", is about how to successfully start a gesture-based interaction with a novice user. The user will need to know that gesturing is supported by the interface, as well as *what* gestures are supported. They will also need to know *what will happen* when they perform a particular action (i.e. what action will be triggered). In short, they will need to know *what to do*, *how to do it*, and *what the results of their actions will be*. And they will need to be informed of all this during the interaction itself, without the opportunity for formal training or instruction.

The second challenge is about maintaining the gesture interaction once it has started. How do we make sure the user knows they are being understood? How will they know they are performing gestures "correctly"? And how will they know when something has gone wrong? People are highly adaptive, but to maintain an interaction when errors and misunderstandings arise, there needs to transparency to the user about the state of the gesture system. This means it should be clear to them what the system is interpreting from their actions, and it should be clear when there is something wrong that requires their input.

Finally, a gesture interface has to accommodate different users. Different people will want to use different gestures (e.g. due to personal preferences, social custom, or physical comfort or limitations). And they'll have different expectations – about the system's capabilities and "preferences", as well as about the system's expectations of *them*, the user. Different users will start with different mental model of the interaction, driven largely by their own experiences and expectations.

The idealized gesture interface should provide guidance to novice users within the interface itself, allowing them to bootstrap an interaction without outside aid. This means providing appropriate *affordances* that indicate to users what they can do, how to do it, and what to expect. The interface should also be transparent enough about its status that the user recognizes when something has gone wrong, particularly if it's something that can be fixed by some action on their part. This should be done via responsive *feedback*. As to the challenge of supporting different users, perhaps the most straightforward approach is to focus on gestures that are as universal as possible. Large and specialized gesture sets may be useful for niche applications, but they are ill-suited to general-purpose interaction, and that should be the ideal when aiming to support novices first. The interaction should be driven primarily by *universal gestures*, if they exist.

These ideals form the basis for the gesture interface that was developed in this work, the related work corresponding to it, and the research questions that the experiments address. Appropriate affordances, feedback and universal gestures *may* support a user-friendly interface, but there is as yet no formal understanding of how these features actually affect gesture-based interactions with novice users, nor to what degree they do so. A better understanding of the effects of affordances and feedback may help motivate gesture interfaces that better support novice users, including for applications like assistive robots, where a person's well-being in on the line.

The reason this work pays special attention to gesture interaction with assistive robotics is that these robots represent a fast-growing industry that with a greater societal impact than other common gesture applications, like entertainment (e.g. video games), productivity (e.g. interactive whiteboards), and industry (e.g. industrial robot control). And within this domain, assistive robots for search and rescue represent many of the interaction challenges of the entire field. Users are frequently novices; they may have limited mobility and varying language preferences; there may be uncontrolled lighting and occlusions; and it is important to keep the user calm and encourage their cooperation with the robot and its operators or peers (e.g. responders, doctors and nurses). Gestural interactions with such robots may soon become commonplace, so it is important to understand their requirements and challenges.

1.3. Contributions of this Dissertation

This dissertation presents three major contributions: 1) a, 2) a novel general-purpose gesture interface built on a virtual touchscreen paradigm and designed to support novice users, 3) an in-depth evaluation of affordances and feedback in this interface, and 4) a set of findings and guidelines that can inform the design of future gesture interfaces, in particular those that aim to support novice users. These guidelines center on a novel framework for gesture interaction design and the role of affordances and feedback in gesture interfaces.

Different gesture interfaces and applications may call for distinct affordances and feedback, but the framework presented in this work organizes their selection and design based upon four fundamental categories: *affordances for what a user can do*, *affordances for how to do something*, *feedback acknowledging user actions*, and *feedback providing system status updates and/or warnings*. These categories are derived from the affordances and feedback described in the HCI literature on gesture interfaces, as well as from insights about the nature of affordances from the cognitive psychology literature.

Affordances and feedback in the four framework categories are incorporated in the novel gesture interface presented in this work, which is general-purpose and centers on a virtual touchscreen paradigm. This interface is designed for in-air gesturing with a computer or robot, and serves as a wrapper around any interface that can be operated with a touchscreen or mouse input.

The same gesture interface was used in a 40-person user study to answer two core research questions: 1) *How does including visual affordances and feedback impact the* *performance of a hand gesture interaction with a novice user?* and 2) *What types of affordances and feedback are most important for such an interaction?* The study is divided into three sub-studies with a total of six study conditions (two conditions each), and is evaluated with quantitative performance measures and qualitative participant responses. Several covariates are analyzed as well to determine their effects on the experimental results.

The findings of the experiments were surprising in several ways, and together with the guidelines derived from them, they make up the final major contribution of this dissertation. These guidelines tie together the framework for affordances and feedback in gesture interfaces with the results and findings of the experiments. They are intended to serve as motivation and reference points for the design of future gesture interfaces.

2. RELATED WORK

Affordances	Perceivable cues or signals that tell users *what they can do* in an interface, and suggest *how to do it*.
Feedforward	Cues or signals that suggest to interface users *what the result of their actions will be*.
Feedback	Information presented to the user for the purpose of *clarifying the state of the system*, particularly with respect to user actions.
Universal Gestures	Gestures that: *everyone knows how to do*, *everyone does intuitively*, and *everyone understand when someone else is going them*.

Table 1. Key terms used in this related work section, and in the remainder of the dissertation. These are defined with respect to their application in a gesture interface.

The related work described in this chapter is separated into 3 sections: 1) existing gesture interfaces for HRI and their shortcomings with respect to assistive robotics; 2) the role of affordances and feedback in gesture interfaces; and 3) the use of in-air gestures with virtual touch screens. The chapter closes with summary of findings and a classification of visual affordances and feedback into four categories that are of particular importance to gesture interfaces. Table 1 defines key terms used in this chapter and throughout the remainder of the dissertation.

2.1. Gesture Interfaces for Human-robot Interaction

Surveys by Mitra [41], Wachs [71], and Suarez [59] describe some of the gesture interaction systems that have been developed for robots, but even a cursory search reveals that there are far too many such systems to fully enumerate; instead, 17 representative works are briefly described here. These works demonstrate commonplace methods used for gesture recognition, applications for gesturing with a robot, and a set of trends regarding the types of gestures supported and the assumptions made about the users of these systems.

The earliest usage of vision-based gesture recognition for interacting with a robot was by Bonasso and Kortenkamp [7, 29], who used a method they called chained proximity spaces to track arm joints in stereo images. Like many of the researchers that followed them, they focused on recognizing and interpreting pointing gestures and other static arm poses. Other early examples include works by Cipolla [12] and Triesch [62], both of whom also used stereo vision systems, but tracked features of the hand, rather than the whole arm. Cipolla's work focused on a single pointing-hand gesture, whereas Triesch recognized six static hand poses.

Becker [5], Rogalla [53], and Malima [36] also primarily supported static hand poses, but without the use of stereo imaging. Waldherr [72] and Van den Bergh [64] designed interfaces for full-arm pose recognition and pointing, like Bonasso and Kortenkamp had done. Waldherr's was also one of the earliest to support dynamic gestures (that is, gestures that include movement) – specifically, a waving gesture. Calinon [10] and Qian [50] also incorporated dynamic gestures and Qian's is an example of a system using the popular and low-cost Kinect sensor for image acquisition and tracking. More recent systems continue to use deictic or pointing gestures to direct a robots attention [38, 54], or a fixed set of static [60] or dynamic gestures [51, 76, 80] which trigger a particular action. Frankly, the novelty of these and many other similar systems is often minimal, and the contribution is usually to simply emphasize a new gesture recognition method or application. However, many published works are indistinguishable even in these respects.

Two clear trends emerge when looking at these systems: 1) they support either pointing gestures (occasionally in conjunction with spoken commands) or else a set of static poses or dynamic gestures which trigger specific predetermined actions; and 2) their usage of gestures is specialized for a particular interaction. This means that for systems that use a gesture set (be it static or dynamic), the user has to already know the supported gestures. And for both those systems and the ones that use a pointing-based interaction, the user has to also know ahead of time what action or response will be triggered for a particular gesture (i.e. they have to know that the gestures "mean"). The expectation is that users will be instructed or trained to use the system and have time to practice with it. However this expectation fails for large classes of gesture applications, including their use by assistive robots, which by their nature frequently interact with novices.

So current gesture interfaces for human-robot interaction are not well suited for novice users, as they require prior training or instruction to use effectively and they lack built-in guidance for new users. Furthermore, these interfaces don't provide explicit interaction feedback to the user – making it difficult to interpret mistakes and failures – or else the feedback they do provide is itself only understood with prior instruction. This problem is exacerbated by the fact that people apply social norms to interactions with computers and robots, and they often implicitly assign them [and their interfaces] greater-than-supported expectations of usability and responsiveness [43]. Novice users will not know how to use these interface; they will not understand whether the interaction is proceeding as expected; and any assumptions they make about the interface's usage or capabilities will likely be broken, leaving them particularly frustrated with the experience.

2.2. Affordances and Feedback

Affordances and feedback are fundamental aspects of interaction design, yet their integration in gesture interfaces has not been holistically studied. *Affordances* are perceivable action possibilities – they tell users what they can do. They may be inherent or explicitly designed, but in either case they should be recognized by observers rather than recalled, with little or no cognitive effort [46]. *Feedback* is real-time information presented to a user to make clear to them the status of a system, particularly regarding actions that the user has taken or needs to take. Without the benefit of formal instruction, appropriate affordances and feedback within a gesture interface may provide novice users with enough guidance and support to help them start a gesture-based interaction, and to maintain it as errors or misunderstanding may arise.

These principles have been used in an ad hoc manner in some gesture interface, and specific implementations have occasionally been evaluated for their effectiveness in some particular task or purpose. But there is not presently a general-purpose framework for the design and use of affordances and feedback in gesture interfaces, nor has there been an evaluation of the types of them which are best suited for such interfaces.

2.2.1. Insights from Cognitive Psychology and HCI

Affordances were first described by James J. Gibson in his works on animal ecology [18]. Gibson's affordances are *inherent* in an environment, whether observed or not, they are *directly perceived* by an observer, and they are primarily *visual*. They may depend on the observer's perspective, but they are independent of the changing needs of the observer. Don Norman later introduced affordances to the fields of design and human-computer interaction (and by extension human-robot interaction) [46], but with some changes. Norman's affordances are still directly perceivable – and it was he who emphasized that affordances are recognized rather than recalled – but he focused primarily on *designing* affordances for a particular need [45].

Discussions by Gaver [17], McGrenere [39] and others have attempted to clarify the meaning of affordances for interface design, and made efforts to classify types of affordances and distinguish them from other concepts. These classifications and distinctions include: affordances vs event-like properties [33], function vs affordance [35], perceptible affordances vs hidden affordances [17], affordances in information vs affordances in articulation [69], and Gibson's vs Norman's affordances [39]. In particular, the distinction McGrenere makes in [39] is that Gibson's affordances are "action possibilities", while Norman's affordances are "perceived suggestions". These works demonstrate the diversity of perspectives on designed interface affordances, but three common recommendations can be derived: 1) affordances should be directly related to possible actions, 2) affordances should serve to improve the usability and usefulness of an interface, and 3) they should do so by elucidating the utility of elements in the interface.

So affordances in a gesture interface should be clear and direct, and they should clarify to the user what they can do in the interface and how to do it. Importantly, affordances may relate to physical user actions (e.g. pointing) as well as interface-specific directives (e.g. selecting). For some interface-specific actions, an affordance may also need to suggest what will result from a particular action – this can be done by incorporating feedforward.

Feedforward is an interaction concept closely related to affordances, and is defined as a cue or signal that suggest to users what the *results* of their actions will be [14, 66, 77]. Feedforward is therefore specific to each interface; a user's actions – whether prompted by affordances or not – will have different results in different interfaces. However, feedforward can itself be thought of as a type of affordance, and may be incorporated in the design of affordances for a particular interface [66]. Such an affordance then serves a dual purpose: it should be designed primarily to tell a user about an action they can perform (like other affordances do), but it should also suggest to them what might result when they perform that action in the particular interface.

Feedback is another cornerstone facet of interface design, and it is most commonly described as information presented to the user which correspond to actions they perform in the interface [55, 56]. Most feedback is provided visually, but other forms that are common include audio [31, 58] and haptic [13, 34, 47] feedback. In any case, the purpose of feedback is to "close the interaction loop" by making the system's status perceivable [31]. And feedback should be about user actions: actions the system has detected or interpreted, and actions the user needs to take in response to the system's status (e.g. when the user needs to act in response to errors and warnings) [56].

A common recommendation for interface design is that feedback should be provided in a timely manner for *each* user action, in order to give them time to react if they make an error [4, 55, 56]. This recommendation was formed in the domain of traditional PC software though, where actions are primarily keyboard and mouse button presses, which are temporally discrete and not usually subject to hardware failure. In other words, each *detected* action tends to correspond to an *actual* user action, and the feedback primarily serves to allow the user to detect unintentional actions they performed. On the other hand, gesture input is less robust and more ambiguous than traditional PC input, and it involves actions that are continuous in time and space, making them difficult to segment. So a response for each "action" in a gesture interface may not be tenable. Instead feedback may be provided for each detected action that is meaningful to the interface (e.g. each detected gesture). Also, in this case the feedback also serves to inform the user of misrecognized or misclassified gestures in addition to inadvertent ones. Similar to the distinction made here, Beaudouin-Lafon [4] makes a distinction between 1) user actions and their corresponding feedback and 2) interface commands and their corresponding responses.

Finally, it is meaningful to understand the role of different *types* of affordances and feedback in a gesture-based interface. Independent of the particular interface or task, a user of a gesture interface needs to know *what* gestures are supported, *how* to perform them, and *whether they are being understood* by the interface. When considering the *contents* of the interface itself (independent of the means of gesturing), the user needs to know *what* actions they can perform, and also *what will occur* in the interface in response to those actions. Additionally, users need to know about the *state* of the system as a whole, especially with respect to errors and any actions they (the user) need to take. So the two types of affordances that this work focuses on are those that tell a user *what* can be done in an interface (which may also suggest *what will occur*), and those that suggest to them *how* to do something. And the two types of feedback focused on are feedback *acknowledging user actions* and feedback about the *system's status*.

2.2.2. Feedback and Affordances in Gesture Recognition Systems

The inclusion of explicit affordances and feedback in gesture recognition systems is uncommon, and the examples that do include them only do so for some facets of the interaction. In a hand-pose estimation system by Lin [32], the user is provided with a rendering of the hand pose the system has detected so that they may adjust their pose as necessary. Lin does not measure or discuss the performance impact of this approach, but Kratz [30] used visual feedback in a similar manner and observed a gesture recognition-rate improvement of 35% for dynamic sketching gestures performed with a hand-held device. The OctoPocus system by Bau and Mackay [3] showed a slight reduction in input time (8% - 16%) when using visual affordances to interactively train new users to perform arbitrary gesture commands, compared to conventional help menus.

Showing users a system's supported gestures and how to perform them is a key aspect of the "walk-up-and-use" experience that Bragdon et al. aimed for in their GestureBar work [8]. They showed that users discovered and learned unfamiliar gesture commands much more easily with their familiar toolbar-like interface than with reference crib sheets. Gesture performance improvement was also found by Freeman et al. [16] in their ShadowGuides work, which combines a visual representation of the current recognized hand pose and feedforward guide information to help users complete a supported gesture. And the integration of affordances was also found in the Choreographic

Buttons work by Webb at el. [75], which investigated the integration of full-body gestures, visual feedback, and a choreographic grid serving as an interaction space.

The aforementioned examples demonstrate the importance of making transparent to the user what the system has interpreted from their actions, as well as informing them of how to perform supported gestures. But still missing is feedback about system status, especially concerning whether the user's hands are visible to the system – which is not guaranteed for complex, non-laboratory conditions. This may be especially true for novice users, who must learn the interaction mechanics during the interaction itself, and thus may be more prone to misunderstanding the system's requirements and limitations.

2.3. Universal Pointing and Virtual Touchscreens

The effort to support general-purpose gesture interaction and to avoid large and specialized gesture sets led to the concept of universal gestures – that is, gestures that even novices don't have to learn because everyone knows how to do them. In particular, a universal gesture may be defined as one that everyone knows how to do, every does it intuitively, and everyone understands when someone else is doing it. A search of the literature on communication and psychology discovered that there does exist such a gesture, and it is perhaps unique: pointing. Pointing is a direct and truly intuitive means of communication, and is considered by Kita and McNeill to be the most universal of all gestures [26, 40]. In fact, pointing is known to be used from infancy [9] – although there is not complete agreement in developmental psychology on why. A once-highly regarded hypothesis by Vygotsky [70] that pointing is an extension of the desire to reach for food and objects has been found to have little evidence [9, 37].

But although pointing is universal, it is clear that *manners* of pointing are not universal. Different cultures and sub-cultures point differently – e.g. pointing with one or more fingers outstretched, with the remaining fingers loose or tight, with the thumb raised or lowered, and even pointing with the thumb itself (as has become common in recent decades by politicians). And although these different forms are easily understood as pointing across cultures, they may present a challenge to a system that is detecting such a gesture.

Pointing alone does not afford a particularly rich interaction, but it can be leveraged to mimic a now highly-familiar modality for computer interaction: touch screens. Familiarity with touch input has become ubiquitous over the past decade as smartphone and tablet adoption has continued to soar and even saturate across different age groups [6]. Several researchers have already capitalized on that familiarity to make their own touchscreen-like interfaces. Cheng [11] used visual tracking of the hand and face to create a pointing vector that directs an on-screen cursor. Moeller's ZeroTouch [42] uses an array of infrared LEDs in a rectangular frame to detect fingers and other objects within the space, enabling multi-touch, pen, and free-air interaction. Tosas's mixed-reality system [61] uses visual fingertip tracking to support interaction with a virtual keypad in a virtual reality application. Jing [24] used Kinect-based hand tracking as a proxy method for detecting and recognizing pointing gestures, which can be used to select one of six large targets on a wall. Jing did not make it clear how selection of targets was done, though it appears that simply pointing at a target is counted as "selecting" it (i.e. no hovering or distinct selection gesture appears to have been used). Wang's RF-IDraw [73] demonstrates uses a finger-worn RFID tag that is tracked in 3D space, but focuses on trajectory accuracy rather than a particular interaction. And Wilson's TouchLight [78] use a projected display on a vertical glass panel that then serves as a touchscreen on which hands and fingers are visually segmented to support a "Minority Report-like interface".

A disadvantage of using a virtual touchscreen compared to a physical one is that the physical affordances and feedback present in the latter interaction are lost. Touchscreens afford touching; they have clear and tangible interaction bounds; they support a one-to-one touch interaction because users can see precisely where their fingers will land on the interface; and they provide tactile feedback on contact (both passive contact as well as active haptic responses for interface-specific feedback). This limitation applies to "in-air" gestures, which were used by the majority of the described systems. However, unlike the rest, TouchLight [78] uses contact-based gestures for interaction. These preserve the physical affordances and feedback of a traditional touchscreen, but require a large, physical, pre-placed glass panel for interaction.

Three of the described works on virtual touchscreens contain explicit affordances and/or feedback or else discuss their effects on the interaction. Tosas's virtual keypad uses a physical grid (a rectangular frame with some string) in front of the camera to provide context about the dimensions and position of the virtual panel, and presents a virtual hand model (in a head-mounted virtual reality display) with which the user aligns their own hand to initialize the hand tracking (Tosas also discussed – but seems not to have implemented – audio feedback to indicate "contact" with the virtual panel). The frame of ZeroTouch's interface similarly provides affordances about the size and position of the interface, however the authors discuss that users were often confused about whether their hands were inside the sensor. The authors regard "activation feedback" as a pressing need for the interface, and offer that integration with a haptic feedback device may best address this need. Cheng's hand-head interaction vector is represented directly on-screen with a cursor, and the display is designed to move with the user's head so that the "interaction space" remains within reach. This work also includes a distinct selection gesture, triggered by either crossing a virtual threshold (which was difficult to detect with their system), or by hovering (which required that users hold still for one second).

A key advantage of a virtual touchscreen paradigm for interaction is that it supports *general-purpose* interaction. Unlike interface-specific gesture sets, a virtual touchscreen can serve as a front-end for almost any interface that is designed for touch input (or even for more traditional mouse input). This is the case for ZeroTouch, which demonstrates two distinct interface applications: a painting application called intangibleCanvas, and a 3D navigation/manipulation application called Timepiece Explorer. This flexibility is the case in general for these types of interfaces, but the implementation must take into account affordances and feedback for the application as well as for the interface.

2.4. Summary of Related Work

Three conclusions are derived from this related work. 1) Gesture interaction with novice users is not well studied or supported in HRI. 2) In lieu of formal training, novice users of a gesture interface can be guided by affordances that tell them *what they can do*, *how to do it*, and *what might result* from their actions, as well as feedback that *acknowledges their actions*, and feedback that clarifies the *system status*. And 3) a pointing-based gesture interface with appropriate feedback and affordances may serve as a general-purpose paradigm for interacting with touch-enabled applications.

Real world use of gesture recognition with an assistive robot would necessarily involve use cases with conditions that are challenging to any hand-tracking and gesture-recognition algorithms (poor lighting, occluded bodies, complex backgrounds, and multiple foreground objects). These have not been well studied or tested in HRI, and certainly the performance for any given gesture system would be expected to decline under such conditions, leading to more tracking and recognition errors. However, appropriate feedback may mitigate these errors, and potentially make them recoverable. So I believe that *studying the effects of feedback on interaction performance is a more pressing endeavor than attempting to simply improve tracking and recognition rates*.

Without an opportunity to be trained to use a system ahead of time, novice users of a gesture system must learn the interaction mechanics during the interaction itself. The presumption made in this work is that doing so without the benefit of appropriate guidance and status information via affordances and feedback is much more challenging, and the experiments described in chapters 5 and 6 sought to determine whether this is truly is the case.

Four categories of affordances and feedback were identified to explore in detail, in particular to determine their relative importance in a gesture interaction. The two categories of affordances are those that indicate to the user what they can do in an interaction (i.e. classical *perceivable action possibilities*), and those that suggest to the user how to do something, such as how to perform an unfamiliar gesture (affordances for interaction guidance). This evaluation will clarify whether it is more impactful to show a user *what they can do* or *how to do it*. The two types of feedback investigated are *acknowledging user actions* (to “close the interaction loop”), and *system status* feedback (to aid the user in detecting and recovering from errors).

Recalling that the objective of this work is to support novice gesture interaction, the system developed leverages the familiarity and general-purpose nature of a virtual touchscreen paradigm for interaction. This enables a focus on universal pointing gestures, and minimizes the degree to which novice users have to learn to use the interface. Such an interface requires that designed affordances and feedback support use of the gesture interface as well as the underlying interactive application.

3. A GENERAL-PURPOSE GESTURE INTERACTION SYSTEM

This chapter describes the general-purpose gesture interface developed, including the initial framework created to help design the interface and its affordances and feedback. The system is designed specifically to accommodate novice users – with particular consideration for the needs of assistive robotics – and also to maximize its generalizability to different interactive applications. To this end, the system design emphasizes integration of affordances and feedback, and is modeled on a virtual touchscreen paradigm. This approach helps provide a clear mapping between gestures and in-application actions, and also eases familiarity for novice users. Following the description of the framework, the gesture interface is described, with emphasis on its affordances and feedback, and finally the rationale for several design choices is discussed.

3.1. A Preliminary Framework for Interface Affordances and Feedback

Building upon the related work on affordances and feedback in human-machine interfaces, a basic framework was developed for incorporating affordances and feedback in a gesture interface. This framework inform the design of the general-purpose gesture interface described in the following chapter, which was in turn used in a series of user studies to evaluate the effectiveness of its affordances and feedback. Importantly, it is not the framework that is evaluated by the experiments, but rather the four categories of affordances and feedback that were identified and implemented with the help of the framework. However, the framework is re-evaluated following the analysis of the experimental results and presented in an updated form in chapter 6.

In developing this framework, this work began with Don Norman's fundamental principles [46] of interface design – visibility, constraints, mappings, and of course affordances and feedback – and then considered how the implementation of these concepts might be affected by the integration of gesture input in an interface. It was intended that the framework be generic with respect to types of gestures, interface applications, and the familiarity of the user. The framework is therefore meant to serve as a reference to aid in the design of novice-friendly affordances and feedback for combinations of gesture interfaces and user applications.

Norman described the relationship between a person and a system as a pair of gulfs – the *gulf of execution* and the *gulf evaluation* – which the user, system, and system designer must together bridge [44]. In bridging the gulf of execution, the user transforms their goals into relevant *intentions*, which they in turn specify as *actions* to be executed via the system's *interface mechanisms*. It is the role of affordances to clarify the usage of those interface mechanisms. To bridge the gulf of evaluation, the system provides an interface for the user to *perceive*, the user must then *interpret* what they see, and finally *evaluate* it with respect to their goals. So along with affordances and the actual content of the application, the interface should also provide feedback that allows users to discern whether the actions they've executed align with their goals.

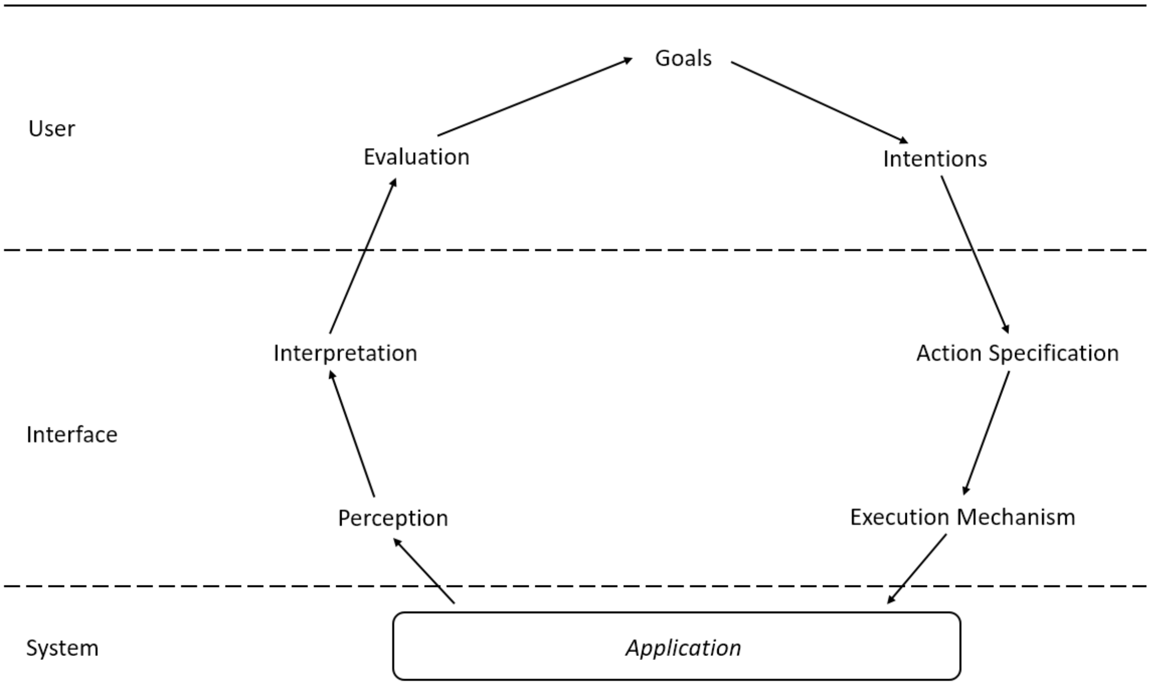


Figure 2. The seven stages of interaction, with respect to a computer interface. The three left stages bridge the gulf of evaluation, while the three right stages bridge the gulf of execution. The interface comprises the four stages which are dependent on both the user and the system. Adapted in part from [46].

These seven stages form a cycle of interaction, shown in Figure 2. At the bottom is the application with which the user is interacting (which Norman called the "world"), at the top is the user's goals with respect to the application. The gulf of evaluation is bridged by the three interaction stages on the left, while the gulf of execution is bridged by the three stages on the right. The figure also demonstrates how some stages depend exclusively on the user, while others depend on both the user and the system with which they are interacting. It is this latter set of stages which fall within the scope of the system's interface.

While it may be clear that *perception* and *execution mechanism* stages are part of the interface, it is perhaps less obvious that *interpretation* and *action specification* are as well. After all, it is the *user* that interprets what they perceive from the interface, and it is also they that specify the actions they will perform. However, the user must first know what actions they can perform, and so the interface's affordances affect the action specification stage. And the user's interpretation of what they perceive is similarly dependent on the clarity of the feedback that the interface provides.

The two categories of affordances identified previously readily correspond to two of the stages in this interaction model. Interface affordances indicating *what a user can do* provide guidance for the *action specification* stage. These affordances may optionally suggest *what will result* from particular actions as well. The *execution mechanism* stage, on the other hand, is closely tied to interface affordances indicating *how to do something*. For users unfamiliar with the interface, these two types of affordances may be crucial to bridging Norman's gulf of execution.

While the *perception* and *interpretation* stages of interaction are both supported by the feedback the interface provides, this interaction model hides an important distinction about the *types* of feedback the interface should provide. When applied to interface design and analysis, Norman's model of interaction favors a view from the user's perspective, centering on the user's actions and the system's response to those actions. This view supports the integration of feedback indicating the system's reaction to user actions, but it omits consideration of feedback that is not directly tied to user actions. Feedback serves to make visible the state of a system, so additional status information needs to occasionally be presented to the user. In particular, the "health" of the system is important to any interaction, so warnings and errors tangential to the interaction need to be considered as well. This type of feedback would be perceived and interpreted much as interaction feedback is, but it doesn't fit as neatly into a user-centric view of the interaction (e.g. goals and intentions).

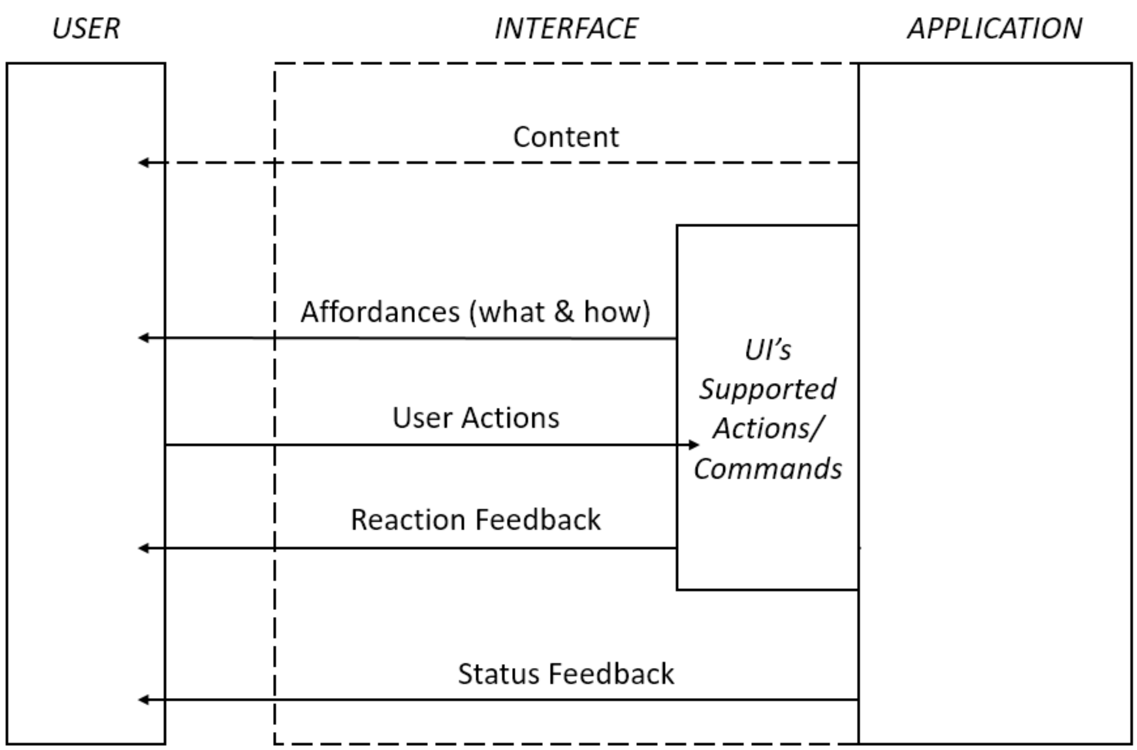


Figure 3. The role of affordances and feedback in an interaction with a *familiar* physical interface. In this model, the application interface may be new to the user, but the physical interface mechanism is familiar (e.g. keyboard and mouse, touch).

Figure 3 presents a view of an interaction from the *system's* perspective, focusing on the affordances and feedback it should provide to the user alongside the actual application content. This model preserves the action-reaction cycle, but simplifies it, and places emphasis instead on the interface contents (including affordances and feedback).

As a companion to Norman's seven stages of interaction, this view serves as a further reference for the design of a system's interface. Specifically, it highlights how affordances and feedback in the interface are related to the user's actions, the actions supported by the application, and the status of the application.

Users interact with traditional interactive applications via a display and a set of supported actions or commands. The supported UI commands should map to actions the user can take, and it is the job of affordances (both "*what can be done*" and "*how to do it*" types) to make that mapping clear. These affordances can be considered precursors to the action-reaction cycle that follows, in which the interface provides a clear reaction to user actions in the form of feedback *acknowledging user actions*. The affordances may remain a part of that interaction cycle though, to serve as action reminders and to elucidate further possible actions as they become relevant. Finally, outside the interaction cycle lies the feedback about the *system's status*, which may indicate errors or warnings, and may directly prompt the user to take some new action. This feedback is not strictly a part of the action-reaction cycle, but it may be presented in a way that avoids interrupting the interaction.

Importantly, this particular view of an interaction allows for user unfamiliarity with the application interface, but presumes a familiarity with the physical interface medium – which limits its applicability to gesture-based interfaces. The principle shortcoming is in the assumed directness of user actions. As users may generally be expected to know how to use a keyboard and mouse/trackpad for traditional PC interfaces, and even a touchscreen for smartphone and tablet interface, interpretation of their physical actions is not a major consideration. But gesturing with a machine interface is not only less familiar, but also much more ambiguous in meaning. Because there is not a clear or universal mapping between physical gestures and any commands or actions they may correspond to in an interface, interpretation will be necessary. And as discussed further in the following section, gesturing in the air lacks the *physical* affordances and feedback of other modalities. So affordances and feedback must be considered for both the application interface and the gesture interface.

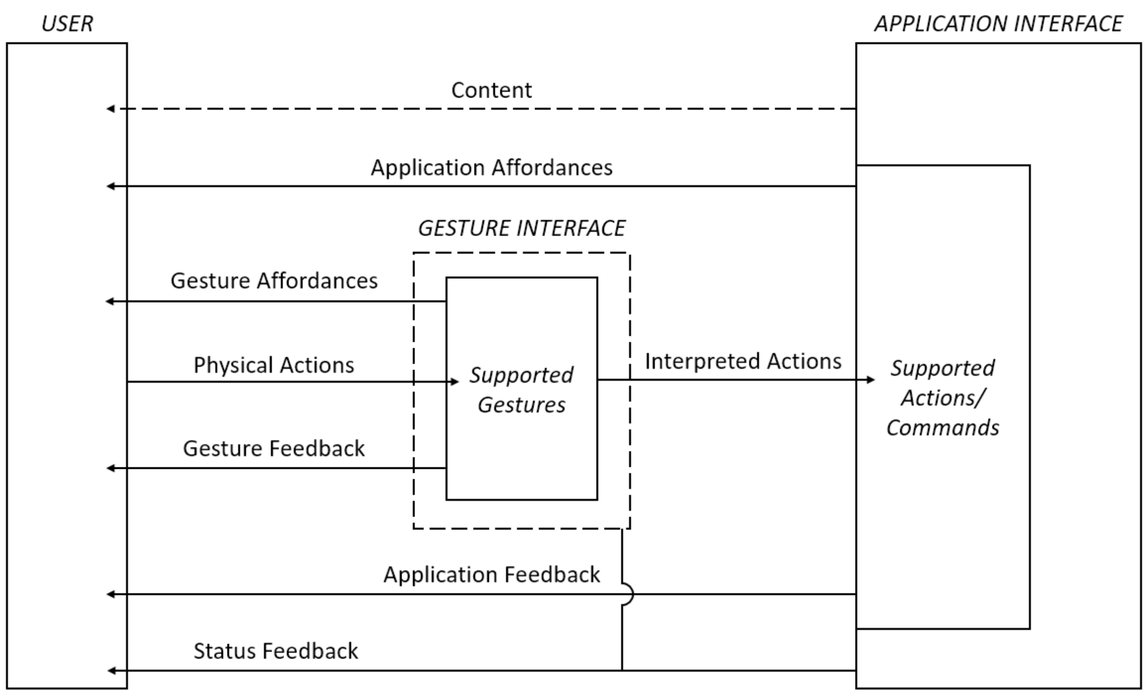


Figure 4. The role of affordances and feedback in an interaction with a gesture-based interface. Two distinct interfaces must be considered: the application interface and the intermediary gesture interface, and both require their own affordances and feedback.

Figure 4 presents a revision of system's interaction model, now including the role of the gesture interface in the interaction – between the user and the application – as well as the affordances and feedback needed for gesturing. It is now more apparent that there are in fact *two* interfaces that must be considered: the application interface and the gesture interface. Whereas a user's actions are interpreted more-or-less directly by an application when using a familiar physical interface, here there is an intermediary gesture system between the user and the application. The system supports a set of gestures, and it translates the user's physical actions into interpreted actions for the application's interface to process. But the gesture system itself re-introduces Norman's gulfs of execution and evaluation, which must be bridged.

The gesture system should be thought of as an interface in its own right – distinct from the application interface – with its own action-reaction cycle and its own need for appropriate affordances and feedback to bridge its execution and evaluation gulfs. As with the application's supported commands, users need to know what gestures are supported, but perhaps more critically they need to know how to perform those gestures. And reaction feedback is needed for both interfaces as well: users should know what gesture the system interpreted, and they should know how the application responded to it. Finally, the gesture interface adds a new point of failure in an interaction, so status feedback must be provided for it as well in the event of errors or other situations requiring the user's attention of action.

A further consideration of this joint interaction is the mapping between the supported gestures of the gesture interface and the supported actions and commands of the application interface. Affordances may clarify what gestures are supported and how to perform them, as well as what actions are supported in the application, but appropriate *constraints* and *conventions* may be needed as well to make the mapping between the two interfaces clear. In particular, semantic constraints can be used to design gestures with clear meaning in the scope of a particular application (e.g. a poking gesture means selecting or tapping something), and logical constraints can provide a natural mapping between some gestures and their corresponding actions in an application (e.g. moving a hand to the left should correspond to some action or element at the left of the application interface). Finally, conventions can be used to leverage user familiarity with a related interface (e.g. the virtual touchscreen that is described in the following section).

This framework (visualized in Figure 4) serves to highlight the implicitly dual-interface nature of a gesture interaction and to clarify the role of affordances and feedback in the interaction. The framework is a companion to Norman's seven-stage interaction model (Figure 2), with a shift in perspective from the user's view to the system's. This view better illuminates the needed affordances and feedback in the action-reaction cycles of both the gesture interface and the application interface, as well as the need for status feedback outside either interaction cycle. The framework's separation of the gesture interface from the application also underlines the need for clear mapping between the supported gestures and the supported actions in the application, which may be addressed by a combination of affordances, constraints and conventions.

3.2. Interaction using a Virtual Touchscreen Paradigm

The general-purpose gesture interface developed is modeled on a virtual touchscreen paradigm and uses a visual interface incorporating affordances and feedback which was designed using the framework described in the previous section. The virtual touchscreen paradigm was selected primarily for its familiarity. The present ubiquity of smartphones, tablet computers, and other touch-input devices is such that the vast majority of the public is at least passingly familiar with touchscreen interaction. So this interface's resemblance to a touchscreen simplifies the learning process for novice users and focuses performance and learning effects on the intended affordances and feedback, rather than on the types of gestures supported. Those affordances and feedback must take into account the needs of the user with respect to the gesture interface and to the application, as well as help clarify the mapping between the two.

In contrast to traditional gesture recognition systems which typically introduce specialized gesture sets, this interface is driven primarily by a single gesture: pointing. Specialized gesture sets – like patterns sketched in the air or specific hand and finger poses – make good demonstrations for hand tracking and gesture recognition algorithms, but they are not particularly useful to untrained users. Pointing, on the other hand, is a highly universal and multi-purpose gesture that needs no introduction. And when used to interact with a virtual touchscreen, simple pointing gestures can support a broad scope of interfaces. Indeed, it can support any interface that is designed primarily for touch or mouse input.

In addition to the factors already discussed which informed the choice to use a virtual touchscreen paradigm – user familiarity, ease of implementation, generalizability to different applications, and support for a clear mapping between gestures and application – this approach provides ample opportunity to design and evaluate affordances and feedback for the interaction in a non-contrived way. Pointing may not require much guidance, but complimentary gestures may be included that need to be learned and understood by users. And transforming a touchscreen interaction from a material, tactile display to an intangible in-air experience means that the physical affordances, feedback and constraints lost in translation need to be replaced somehow in the visual interface. So the system that was developed has multiple instances of each type of affordance and feedback that were identified to be evaluated.

3.2.1. Gestures for a Virtual Touchscreen

A physical touch screen has inherent affordances and feedback which are lost in an in-air interaction; these have to be restored via designed affordances and feedback. The most critical of these is positional feedback. When a user holds their finger over a touchscreen, they already know what part of the interface they will select upon touching it. To preserve this positional feedback in my interface, a cursor is shown in the display which follows the user's hand as long as it is in view. Also, a major part of the reason that pointing is universally understood by people is that we can see an observer's reaction when we point, and our spatial awareness allows us to perceive from their body language whether the observer correctly interpreted the direction in which we are pointing. The cursor takes the place of that reaction in this interface, making it an integral part of the preservation of universal pointing.

Visually detecting and tracking hands in a reliable and robust manner is a challenging and interesting problem in its own right. The method used in this system is a modified form of the boosted cascades algorithm originally described by Viola and Jones

[68], using a depth camera instead of a standard RGB camera. A detained description of the method, and the rationale for its use is found in Appendix B. The algorithm returns a bounding box for each detected hand in a depth image, from which the hand may be segmented and further processing may be applied to identify its pose, track the fingers, etc. However, the gesture system described in this chapter tracks and uses only the hand's centroid to support gesture interaction. This is done for two primary reasons. First, it greatly simplifies the (already quite difficult) hand tracking problem. And second, it allows a focus on tracking robustness rather than sensitivity to fine gestures. This is important because it helps support different forms of pointing, as regardless of pointing type (i.e. the pose and orientation of the hand while pointing), the centroid of the hand moves proportionally to the pointing gesture, both in direction and magnitude.

Unlike the well-defined spatial limits of touchscreens, gesturing in the air is inherently unbounded: it can be done at any scale, in any direction, and from any starting point. This ambiguity does not lend itself well to a consistent interaction, so constraints need to be imposed on the in-air gestures. A 3D bounding box is an efficient solution, and orienting it with the display allows for easy and predictable mapping from the user's hand to the cursor on screen. This mechanism also simplifies the challenge of gesture detection, i.e. when did the user start and stop pointing? There are many different ways to indicate the beginning and end of a gesture, like the position, speed, shape or orientation of the hand. But the fixed interaction space of a 3D bounding box applies a social engineering approach which efficiently solves the problem. Users are considered to be pointing when and only when their hand is inside the 3D interaction space. And a pointing gesture is terminated when the user's hand leaves the bounded space.

To simplify gesture recognition, the gesture interface is purposefully limited to include only the core gestures of a touchscreen interaction: tapping/selection, swiping/dragging, and releasing/dropping (in particular, multi-touch gestures such as two-finger "pinching-to-zoom" are omitted), but pointing alone cannot fully support these. There needs to be a substitute for the act of pressing and lifting a finger from the screen. That is, there needs to be dedicated "tap" and "release" gestures to compliment pointing. The aforementioned choice to track only hand centroids may limit the way these gestures can be expressed, but it actually provides several advantages here as well. First, detecting only coarse hand motion provides a more robust gesture detection than is possible with fine finger gestures, as well as extends support to a larger population as they are easier to perform by people with limited mobility. Second, because these gestures are integral to the interaction but are not universally intuitive (as pointing is), they can serve as a as a non-artificial case study for designing affordances and feedback to teach novices to perform unfamiliar gestures. As the intent is to evaluate the affordances and feedback rather than the gestures themselves, it makes sense to keep the gestures as simple and consistent as possible.

Simple tap and release gestures to compliment pointing can be robustly detected using a virtual selection plane within the 3D bounded interaction space. This a mechanism leverages a natural mapping to the corresponding actions on a physical touchscreen: when the user's hand is forward of the selection plane it is equivalent to "touching" the interface, and when they move their hand back they have effectively lifted their hand from the "surface". The selection plane is vertical and aligned with the display, and while "behind" it the user's hand can be thought of as hovering above the virtual contact surface (as long as it remains within the interaction bounds).

As with pointing, the two complimentary gestures need their own constraints to support a clear mapping to the corresponding press and lift gestures on a physical touchscreen. Namely, the forms of the gestures are fixed (e.g. only full hand movement forward will register as a tap gesture) and the virtual selection plane used to detect them is programed at a set distance from the display and gesture sensor. So these interface features must be discovered by the user as well, and it is the role of affordances and feedback to aid in that discovery.

The three gestures (pointing, tap and release) together with their detection mechanisms (a 3D bounding box and a virtual selection plane) provide a simple but complete analog for a touchscreen-like interaction using in-air gestures. Though the interface is designed to be as familiar as possible for novices, certain features still need to be learned or discovered by the user: including the non-universal tap and release gestures, the bounds of the interaction space, and the placement of the selection plane. Nonetheless, the gesture interface remains compared to the systems found in the related work, and importantly, it is generic with respect to the applications that it may support.

3.2.2. A Visual Interface for General-purpose Gesture Interaction

In accordance with the framework described at the start of the chapter, the display of the gesture interaction system integrates two interfaces: the application interface and the gesture interface. It therefore contains the contents of the application, the affordances and feedback for the application, and the affordances and feedback for the gesture recognition system. The approach taken is to nest the application interface within the gesture interface, as seen in Figure 5.

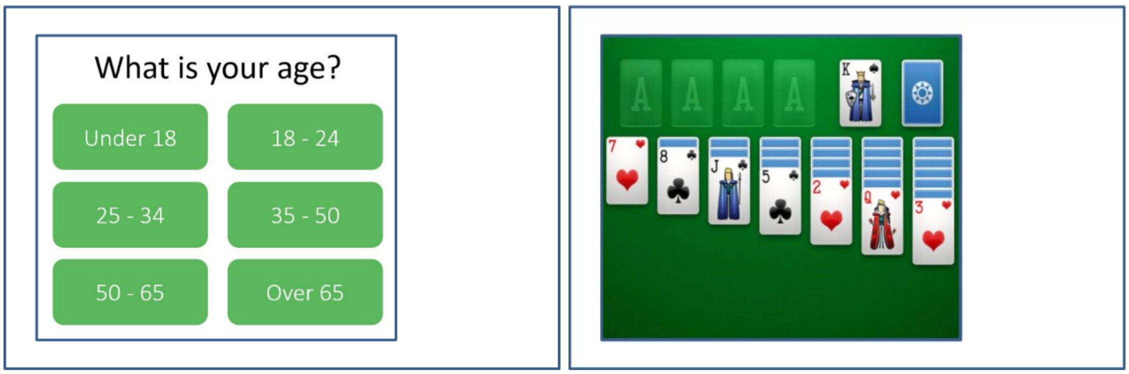


Figure 5. Two examples of the gesture interaction system's visual interface, with different interactive applications: a questionnaire and a card game. The application area is nested within the display and it may contain any application interface that supports simple touchscreen-like interaction. The area surrounding the application interface is dedicated for affordances and feedback of the gesture interface.

As this gesture system is designed in a generic manner to support many applications, its display is primarily devoted to the application or task with which the user is interacting. A designated application area within the visual interface encapsulates the application's interface, and its contents are specific to the application itself. It may be a game, or video chat window, or a set of menu options, or even an entire computer UI (such as a desktop or mobile operating system or application) wrapped within the gesture interface. This area of the interface has a one-to-one mapping to the 3D interaction space in which the user gestures (minus the z-axis, or depth component). This means that the user's hand within the gesture space will be represented by a cursor in the interaction/content area of the gesture interface, and this cursor corresponds directly to their hand position and movement projected onto the xy-plane. Additional affordances and feedback specific to the *application* (see Figure 4) may be shown in the application area as well.

The affordances and feedback for the gesture recognition system are displayed in a reserved space around the application area. This includes a wide area on one side of the interface (which can switch sides depending on the dominant hand of the user) where large, attention-grabbing feedback may be provided. This area can be on either side of the interface, and can in fact dynamically flip depending on the hand the user is gesturing with. This way, their raised arm does not obstruct their view of the gesture affordances and feedback. This area acts as a sort of feedback and affordance dashboard for the gesture interface. However, some gesture affordances and feedback may be placed in other areas of the interface, including overlaid on the application area. The form and placement of the affordances and feedback for both nested interfaces are described in the following section.

3.3. Visual Affordances and Feedback

As already discussed, the affordances and feedback designed and implemented for this gesture interface were motivated by the related work on gesture interfaces and by the framework developed based on fundamental interaction principles. The affordances and feedback in my gesture system can each be logically separated into two categories each. For affordances these are: *things the user can do* and *how to do them*. For feedback these are *acknowledgements of things the user did*, and *updates about the system status*. The relative importance of the affordances and feedback in these categories evaluated in the experiments described in the following chapters.

An additional consideration is that affordances and feedback are needed to support both the gesture interface as well as the application interface. However, as this is a general-purpose interface, the affordances and feedback for the application interface are not designed for a particular application. Instead, they are designed for a small set of actions that are common to all touch-based interactive applications, namely: selecting, moving, and releasing items or elements in the application interface. While the affordances and feedback of a particular application interface may be designed in conjunction with a gesture interface, this is not necessary for integration with the gesture interface. Poorly designed or implemented application affordances or feedback may, of course, negatively impact the interaction.

3.3.1. Gesture and Application Affordances

A total of six explicit affordances are included in this system, four indicating *things the user can do*, and two indicating *how to perform unfamiliar actions*. The possible actions the user needs to be aware of (supported by *can-do* affordances) are that they can: gesture with their hand, point at the application interface, select or grab items in the interface, and move and release items in the interface. The unfamiliar actions that users may need to learn how to perform (supported by *how-to* affordances) are: how to select an item, and how to release an item. Users also need to know how to move an item, but in keeping with the virtual touchscreen paradigm, a movable object follows the user's hand/cursor after it is selected and continues to do so until it is released. Due to this tight coupling with the cursor feedback, moving an item is considered among the application feedback described in the following subsection.

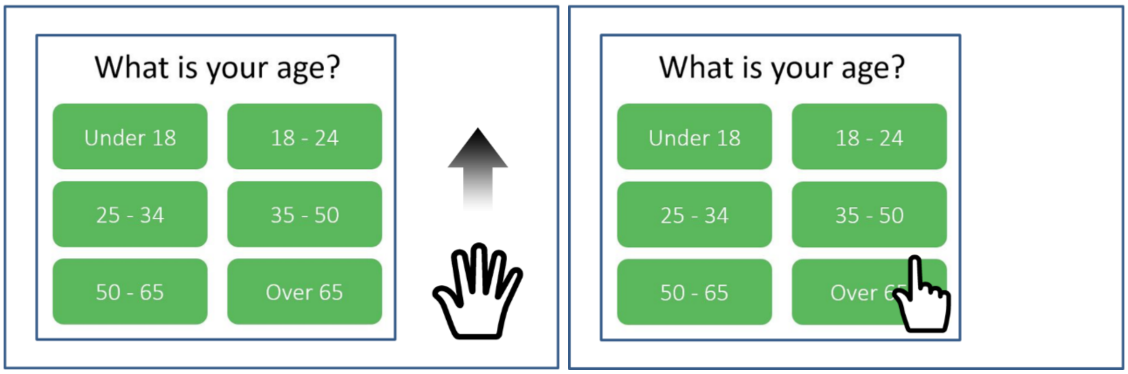


Figure 6. The gesture affordances indicating to the user that they can user their hand to gesture (left) and that they can point at the application interface to interact with it (right). Note that the first affordance is animated while the second is static. These are both *can-do* affordances.

The first affordance is for the gesture interface and it indicates to the user that the system supports hand gesture interaction, thus prompting them to raise their hand into view. It is visually represented by an animation of a hand icon being raised, which is shown at the side of the interface (see Figure 6, left). This is a gesture affordance in the category of things the user can do.

The second affordance is also for the gesture interface, and it indicates to the user that the system supports pointing gestures specifically, and they can point at the application display area to interact with it. Once the user has raised their hand into view, the interface shows that pointing is supported by displaying a static hand cursor in a familiar pointing pose in a corner of the interaction area (Figure 6, right). This is also a gesture affordance, and is also in the category of things the user can do. The icon is replaced by a small circular cursor upon pointing detection (i.e. when the user moves their hand into the 3D-bounded interaction space), and that cursor then follows the hand's motion.

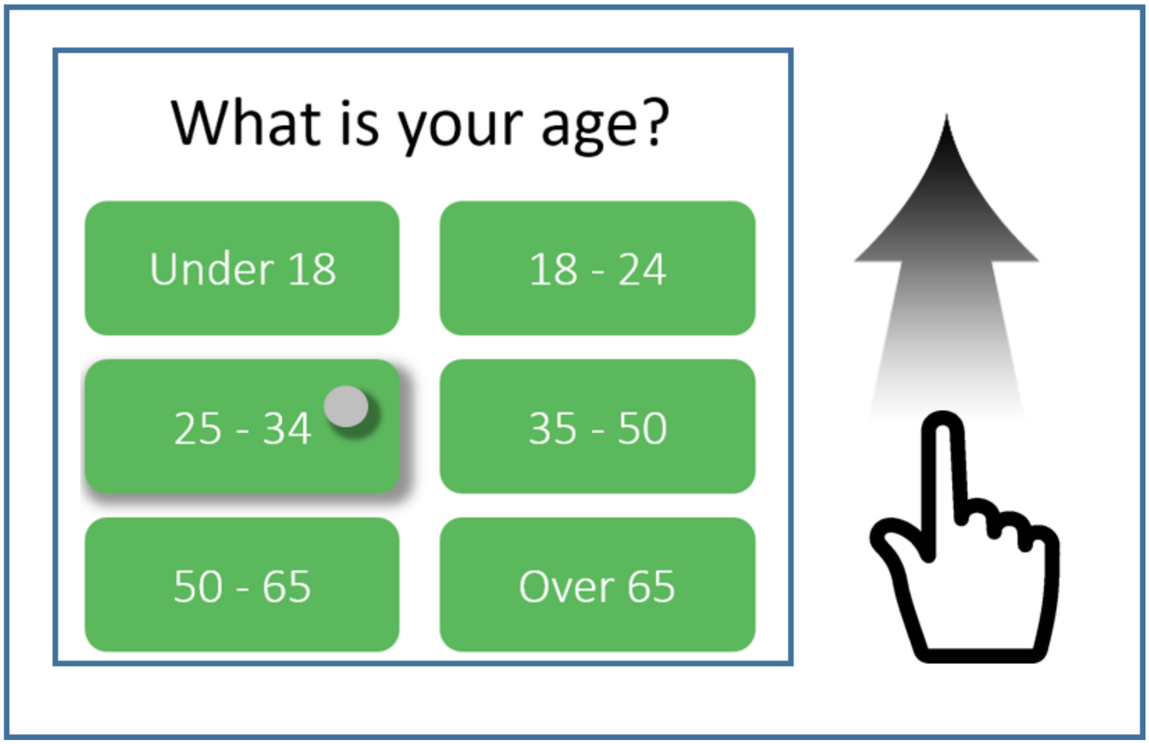


Figure 7. The application affordance indicating that an item in the interface can be selected (highlighted item at middle-left), and the gesture affordance indicating how to perform the corresponding tap-to-select gesture (animated hand icon at right). The first affordance is activated while the user's cursor is hovering over the item; it is a *can-do* affordances. The second affordance is continuously animated upon hovering, but after a delay; it is a *how-to* affordance.

Next is a pair of closely related affordances. The first of these is an application affordance indicating to the user that an item in the application interface can be selected; this is a *can-do* affordance. The second is a gesture affordance showing the user how to perform the corresponding tap-to-select gesture; this is a *how-to* affordance. Examples of both affordances are visualized in Figure 7. Not all elements of the application interface may necessarily be selected, and there are many different ways to indicate that a particular element *can* be. For example, passive affordances can be incorporated into the design of a selectable item, making it a well-defined and/or familiar object (like a button, dial, etc.) or giving it clear separation from the background (e.g. using color, texture, shadow, etc.). Or the same techniques may be applied in an active affordance – that is, an affordance triggered by some action or event (like a timer or a particular user action).

In the application interfaces tested with this gesture system, both passive and active affordance are used to indicate that an item can be selected. Selectable items are distinct and clearly separable objects in the interface, and if the user hovers over one, the application interface further confirms that it is selectable by highlighting the item in some way, within the application area. Highlighting an application element can be done in several ways, but in this system, the method selected was to apply a drop-shadow to selectable items when the user hovers on them. This provides further separation from the background, thus enhancing the perceivability of the affordance at the moment it is most relevant.

Note that this is a departure from the direct virtual touchscreen paradigm (physical touch interfaces don't generally know when a user is hovering their finger), and instead borrows from mouse- and pen-based interfaces. The reason for its inclusion in this interface is twofold. First, it takes advantage of what the gesture interaction modality offers (in this case, the ability to detect hovering) to further enhance the visibility of the interface to the user (in this case, providing an additional affordance). And second, when considered together with the how-to-select affordance, it provides for a clearer mapping between the actions supported by the application and the gestures supported by the gesture interface.

The gesture affordance to show how to perform a selection gesture takes the form of an animated “press” gesture template shown in the side area of the visual interface. This animation shows a hand icon moving up and shrinking as if moving forward/inward, to indicate to the user that a selection gesture is performed by moving their hand forward, toward the display and gesture sensor. This affordance appears on the screen together with the item-can-be-selected affordance to reinforce their connection, but after brief delay. The delay is in place for the benefit of users who’ve already learned the gesture and do not need to see [and possibly be distracted by] it every time they select an item. So a first time user who moves the cursor over a selectable item in the application interface will first see the item highlighted to indicate that it can be selected, and after a moment they will also see the tap-to-select gesture template indicating what physical gesture corresponds to the selection action in the application interface.

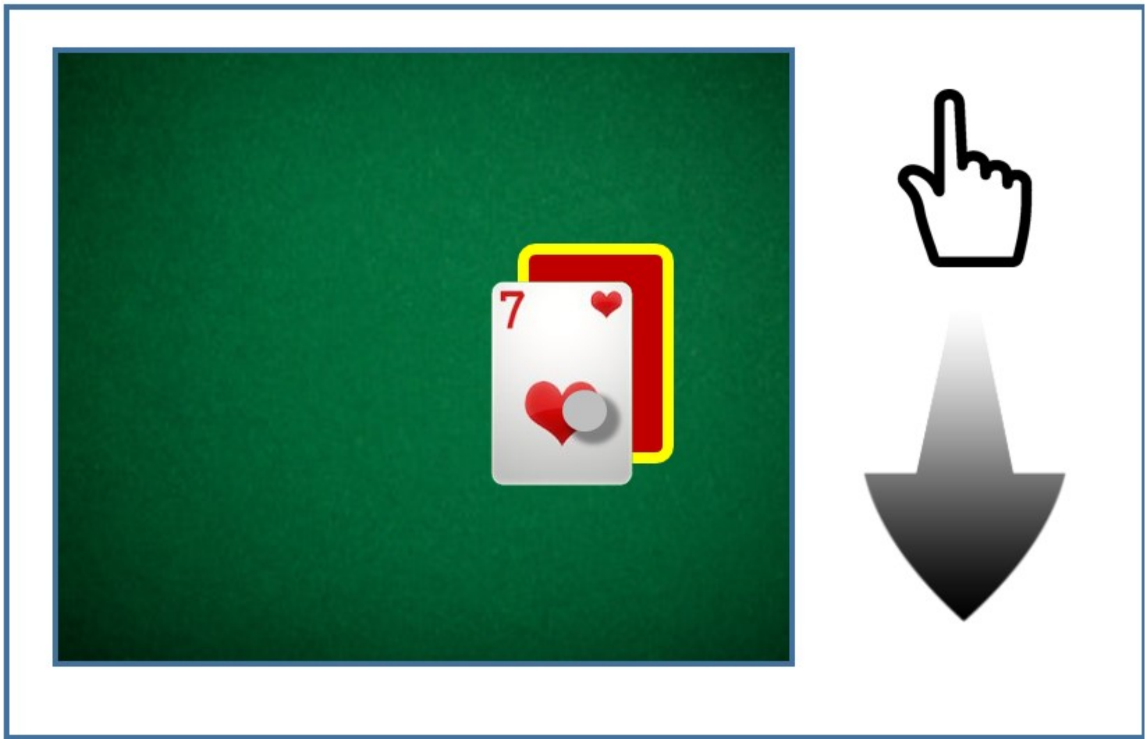


Figure 8. The application affordance indicating that an item in the interface can be dropped in a particular area (yellow-bordered rectangle at middle), and the gesture affordance indicating how to perform the corresponding withdraw-to-release gesture (animated hand icon at right). The first affordance is activated while the movable item is above the droppable area; it is a *can-do* affordances. The second affordance is continuously animated upon the same hovering, but after a delay; it is a *how-to* affordance.

Last is another pair of closely related affordances which act as the compliment to the previous two. The first is an application affordance indicating that an item that is held can be dropped; this is a *can-do* affordance. The second is a gesture affordance showing the user how to perform the corresponding withdraw-to-release gesture; this is a *how-to* affordance. Both affordances are visualized in Figure 8. As with selectable items, not all items can be held and moved, but any selectable item which is also movable will immediately be coupled to the cursor's movement, eliminating the need for explicit affordance telling the user that the item can be moved or how to move it. Once holding an item, users may intentionally or inadvertently drop it without the need for the can-be-dropped and how-to-drop affordances, but their inclusion completes the natural mapping to the touchscreen-like interfaces on which this interaction is based. And they become particularly useful for applications in which movable items can only be dropped in particular areas at particular times.

As with selectable items, there are numerous ways to indicate within an application interface that an item can be dropped; I again use highlighting – this time highlighting the area or goal in which the held item can be dropped by adding a colored border to it. The gesture affordance to show how to perform a selection gesture takes the form of an animated “withdraw” gesture template shown in the side area of the visual interface. This animation shows a hand icon moving down and growing as if moving backward/outward, to indicate to the user that a release gesture is performed by moving their hand back, away the display and gesture sensor.

3.3.2. Gesture and Application Feedback

Nine explicit types of feedback are included in this gesture interaction system, seven *acknowledging user actions* and two providing *system status* information not directly related to the action-reaction cycle. The gesture interface recognizes the following physical actions performed by the user and reacts to them by providing *acknowledgement* feedback: raising a hand into view, pointing at the application area of the interface, performing a tap-to-select gesture, and performing a withdraw-to-release gesture. Note that these are all physical actions interpreted by the gesture interface, but they may not always results in the triggering of a supported action or command in the application interface. The actions recognized by the application interface (and responded to by *acknowledgement* feedback within that interface) are: selecting an item, moving an item, and dropping an item. Additionally, the joint interface provides *system status* feedback when the user's hands are not visible to the gesture system, and when their hand is detected, but is outside the defined bounds of the 3D interaction space. Like the interface affordances, some of these distinct feedback types are closely paired.

The first pair of closely related feedback types are both gesture feedback and they concern the visibility of the user's hands. If the user's hands are not visible, the interface shows the "raise hand" gesture template – note that this reuses the affordance indicating "support for hand gesturing," and so looks the same as in Figure 6 (left). This type of feedback is in the category of *system status*, as it informs the user that the interaction cannot proceed until their hands are visible. If the user raises their hand, the interface acknowledges this by removing the "raise hand" gesture template, and displaying the pointing prompt (Figure 6, right) or the cursor (depending on whether hand is within the bounds of the interaction space). This type of feedback is in the category of *acknowledging user actions*.

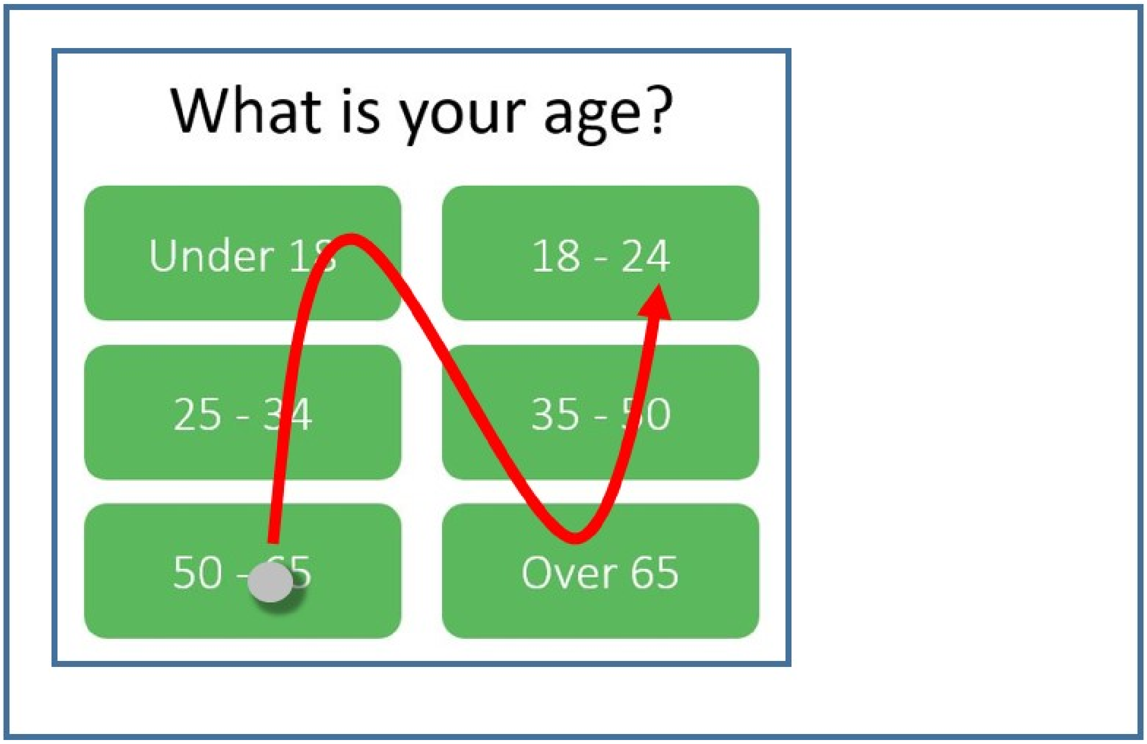


Figure 9. The gesture feedback corresponding to pointing within the bounds of the application interface: a cursor. The cursor is circular and sized to be relatively unobtrusive, but has a drop shadow to make it readily separable from the interface contents behind it. The red arrow indicates that the cursor can move freely within the application's interaction space. This feedback *acknowledges user actions*.

The following gesture feedback is the most elementary to the virtual touchscreen interaction, and is necessary to maintain the universality of pointing gestures: a cursor (shown in Figure 9). Pointing gestures are universal in part because people can see each other's hands and thus observe when they are pointing and where they are pointing, as well as see and feel the location of their own hands. A person who is pointing and a person for whose benefit they are pointing share a visual common ground; the first person knows that the other can see if and where they are pointing. In this interface, the position of the user's hand in the interaction space is representing directly in the interaction area of the visual interface by a circular cursor icon. The user thus knows that the system can "see" if and when they are pointing. This icon follows the user's hand in a one-to-one mapping, minus the z-axis (i.e. the position of their hand in the 3D interaction space orthogonally projected onto the xy-plane). This mapping remains the case for as long as the user's hand can be tracked, and as long as it remains within the bounds of the 3D interaction space. This type of feedback *acknowledges user action*.

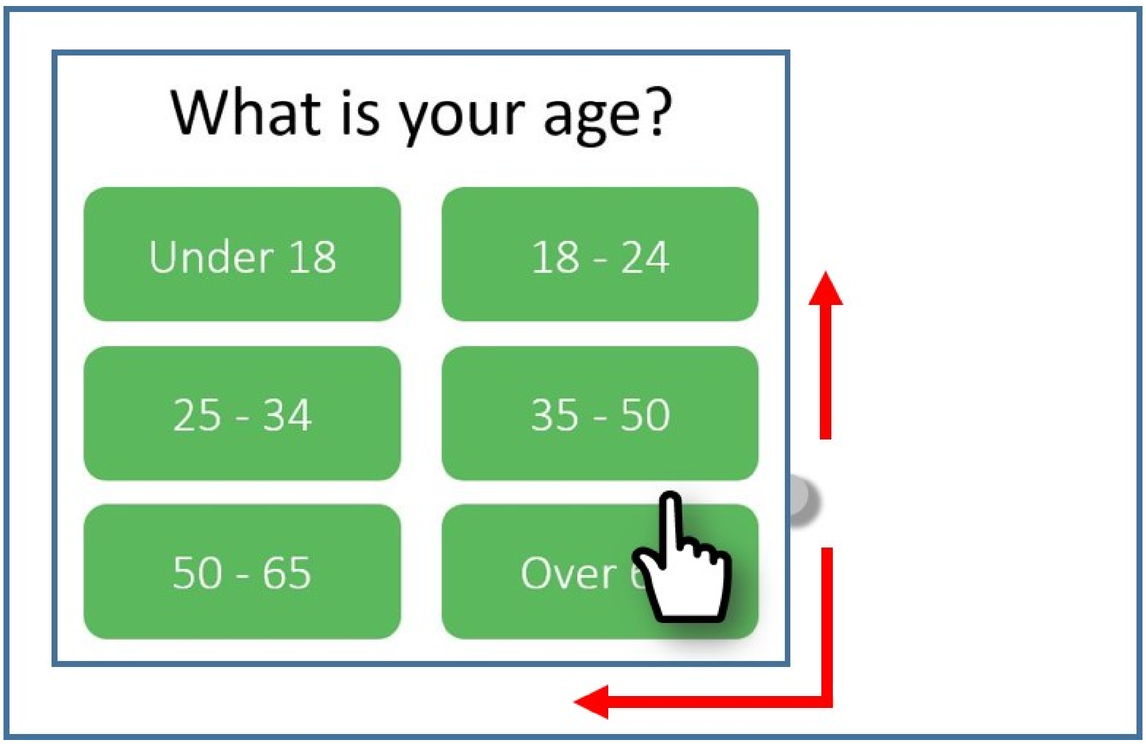


Figure 10. The gesture feedback indicating that the user's hand is outside the bounds of the interaction space. The pointing prompt re-appears and cursor is moved outside and behind the application space in the interface, to help guide to user back within bounds. The red arrows show how the out-of-bounds cursor follows the edges of the application space. This feedback is primarily *system status*.

The next type of gesture feedback is triggered when the user's hand is moved beyond the borders of the interaction space, and serves to inform the user of this fact. If the user moves their hand out of bounds, the cursor disappears and the pointing prompt returns (note again this is the same affordance that indicates support for pointing gestures). Also, if the hand is close to returning within the border, a cursor-like icon appears "behind" and just outside the border of the interaction area of the visual interface, following the movement of the user's hand (shown in Figure 10). This additional feedback serves to help guide the user back to within the interaction space. This type of feedback provides the user with *system status* information.

The remaining types of feedback work in pairs to react to the supported actions within the gesture interface (i.e. supported gestures) and to the supported actions or commands within the application interface. The gesture and application interfaces each provide their own reaction/acknowledgement feedback, but though they are closely related a detected gesture does not always yield a response from the application. All of the following feedback types are in the category of acknowledging user actions.

A pair of feedback types inform the user when they have performed a tap-to-select gesture (intentionally or not) and when that gesture resulted in successfully selecting an item in the application interface. If the user moves their hand beyond the virtual detection plane, the gesture interface lets them know it has detected a tap gesture by animating the cursor to briefly shrink, or appear to recess. Note that this will occur whether or not an item actually selected in the application interface – it can be thought of as analogous to the tactile contact of touching a touchscreen or to the audible and tactile "click" of pressing a button on a computer mouse, both of which also do not necessarily indicate a successful UI selection. If an item is actually selected, the application interface responds in a task-specific manner. This response may be to move on to a new screen, or to highlight the selected item, or something else. In general, the *contents* of the application interface change or are updated in response to successful UI actions and commands.

The next type of feedback indicates that the user is actively moving an item, which can be done for some, but not all selectable items. If a movable item was selected, the cursor remains small to indicate the item is being held and can be moved, and the interface couples the movement of the selected item with gesture cursor, thus moving them together. To ensure this acknowledgement feedback is always synchronized between the gesture interface and the application interface, only movable items can be held in this interface. So moving an item in the application interface elicits both gesture feedback (the shrunken cursor and its movement) and application feedback (the movement of the item).

The last types of feedback included in this interface inform the user when they have performed a withdraw-to-release gesture and when that gesture resulted in successfully dropping an item in the application interface. If a user withdraws their hand behind the virtual selection plane in the 3D interaction space, the interface interprets this action as an attempt to release an item being held (if there is one). This is acknowledged by animating the cursor to briefly grow, or appear to rise. This feedback mirrors the feedback for the "tap gesture," but in reverse. If the item was successfully dropped, this is indicated to the user by returning the cursor to its original size and decoupling its movement from the item (i.e. releasing the item). A particular situation in which attempting to drop an item may not be successful is for an application in which items can only be dropped in specific areas. In that case, the application interface may again respond in a task-specific manner, such as by snapping the item into place, or refusing to release the item from the cursor's hold. Here, the resizing cursor is gesture feedback, while the application's response (e.g. dropped item, snapping it into place, etc.) is application feedback.

3.4. Summary and Discussion of Gesture System

This chapter described how the insights gathered from the related work were developed into a preliminary framework for affordances and feedback in gesture-based interactions, then described the general-purpose gesture interface developed using that framework and a virtual touchscreen paradigm. The framework presents a system view of a gesture interaction, highlighting the distinct gesture and application interfaces, their relationship, and the role of affordances in each. The aim of this system is to better support novice users in a gesture-based interaction, and to study the effects of affordances and feedback on such an interaction.

The virtual touchscreen paradigm – chosen for its familiarity and flexibility – is implemented using three one-handed in-air gestures, together with their corresponding detection mechanisms and a unified visual interface. Pointing is the principle gesture and is favored for its universality; it is supplemented by two additional gestures: tap-to-select, and withdraw-to-release. The gestures are detected by and constrained with a fixed 3D interaction space and embedded virtual selection plane. The visual interface may integrate any touch- or mouse-supporting application interface, also provides the affordances and feedback for both the application interface and the intermediary gesture interface.

The affordances and feedback selected and implemented for this interface serve three primary purposes. First, they address two of the key interaction challenges identified the related work: guidance for unfamiliar users and maintenance of a closed-loop interaction. Second, they are a test case for the affordance and feedback framework introduces at the beginning of the chapter. And third, they are organized such that two distinct categories of affordances and two distinct categories of feedback can be discerned, thus allowing comparative evaluation of those categories. Those categories are: affordances indicating *what the user can do* vs. affordances indicating *how to do something*, and feedback *acknowledging user actions* vs feedback providing *system status*. A total of six types of affordances and nine types of feedback were designed and implemented for the gesture system.

It is the author's belief that the choices made in developing this interface – in particular the supported gestures, the touchscreen paradigm, and types of feedback and affordances that were included – make for a useful and interesting interaction, yet these choices are not trivial and many alternatives may have been tried instead. For example, the choice could have been made to support many other types of gestures. But the selections made were driven by the desire to have just enough complexity to support an interesting interaction, and to test and answer my research questions. In particular, the experiments described in the following chapter aim to answer whether affordances and feedback improve a gesture interaction with a novice in the way expected, as well as discover whether different types of affordances and feedback are more important for the success of such an interaction.

Furthermore, strong emphasis is placed in this work on the experience of novice users (especially those interacting with assistive robots) because it is the author's belief that current gesture interfaces do not do enough to support them. The majority of such systems require prior training or instruction for their users. This is impractical for a robot encountering a patient or victim for the first time. It would be asking a lot of a person in such a situation to read a set of instruction or complete a training course just to be able to perform even simple tasks, like stopping the robot if it is uncomfortably close. An ideal interaction should include mechanisms to guide – rather than instruct – users, and should allow them to easily recover from problems and errors. Carefully designed affordances and feedback are the best and most direct ways to support novice users, and improve ease of use for all types of users. And although gesture interfaces and interaction principles like affordances and feedback have been studied separately for decades, at present role and effects of these principles in gesture interactions remains poorly understood.

Two important features of a gesture interaction with an assistive robot that are not included in this system and the corresponding experiments are the robot itself, and additional interaction modalities, such as speech interpretation. In an ideal case, a gesture interface might be used in conjunction with speech recognition or some other interaction medium to provide a user with redundancy and choice. Here, the focus is only on gesture interaction to better answer the research questions posed. For much the same reason, a physical robot is not included in the experiments because it would add significant complexity and introduce new confounds and points of failure that distract from the target research problem.

4. EXPERIMENTAL DESIGN

The gesture interaction system described in the previous chapter was used in experiments described herein to answer two research questions. Q1: *How does including visual affordances and feedback impact the performance of a hand gesture interaction with a novice user*? And Q2: *What types of affordances and feedback are most important for such an interaction*? The experiments consist of a series of three between-subjects user studies evaluating six distinct study conditions (two conditions per study) regarding the inclusion and exclusion of different types of feedback and affordances. The applications used within the gesture system for this evaluation were two game-like interaction tasks, and evaluation was done using four objective task performance measures and 21 subjective user responses in six categories. Eight covariates were also analyzed, two of which were explicitly controlled for in the experiments.

4.1. The Gesture System and Interaction Conditions

The gesture system was used for three concurrent user studies, conducted in an office-like setting, together with a pair of puzzle-based interaction tasks. Although the system is motivated by a desire to support novice gesture interaction *with an assistive robot*, the research questions are broader than human-robot interaction. Gesture interaction may be supported by any type of computational machine with a human-facing interface, and the questions raised about the role of affordances and feedback in such interactions transcend the interaction mechanics of any particular machine.

For the purposes of the experiments, the assumptions and requirements imposed on the gesture interaction were: the machine must have a *visual display*, the interface must support *hand* gestures, and the machine must *visually* tracks the user's hands. These requirements are grounded in the findings of the related work. As Gibson and Norman state, affordances are mainly perceived visually [18, 46], so it makes sense to require a display for visualizing graphical affordances. Feedback can then also be presented graphically. The choice to support hand gestures (as opposed to full-body or other gesture types) was motivated by McNeill's and Kita's assertion that pointing with the hands is a universal gesture [26, 40]. And the requirement for *visual* hand detection (as opposed to gesture gloves or wearable IMUs) is due to the same reason that motivated the study of affordances and feedback: the desire to support novices in a "walk up and use" gesture interaction. Such an interaction would not allow for users to be outfitted with gesture hardware ahead of time.

The gesture system's visual interface and touchscreen-like interaction was used in these experiments with a standard computer display, though it may also be used on an assistive robot or other type of gesture-supporting machine. A physical robot was not used for this user study because it does not contribute to answering the core research questions, and instead adds complexity, confounding variables, and multiple points of failure. The computer display in these experiments may be considered as if it were the display on a stationary assistive robot, though other interaction effects may be seen when an embodied agent is present. For example, users may have higher expectations of system competency when interacting with an agent that can move as well as "see" and display information [52]. However, such effects are not studied here.

4.2. Research Hypotheses

A total of four research hypotheses were derived from the research questions, and each was evaluated in one of three user studies. Recall that my research questions are: Q1: *How does including visual affordances and feedback impact the performance of a hand gesture interaction with a novice user*? And Q2: *What types of affordances and feedback are most important for such an interaction*? Each of these questions yielded two hypotheses for testing.

4.2.1. Hypotheses from Research Question 1

For the question "How does including visual affordances and feedback impact the performance of a hand gesture interaction with a novice user?" the experiments tested two hypotheses. These hypotheses were expected to be easily supported and thus they were intended to serve as precursors to the more interesting and less predictable hypotheses from research question 2.

H1: *Including visual affordances and feedback will result in better interaction performance*. Because the users targeted in these experiments are all novices, a measurable learning curve was anticipated and significant number of interaction errors that affect their task performance. For example, users may not initially know that the gesture system can only see their hands within a certain range of positions, or may try to perform gestures that the system does not recognize. With visual affordances and feedback, it was expected that users will be able to quickly learn how the system expects them to interact, and be able to quickly identify and recover from problems to return to the interaction.

H2: *Including visual affordances and feedback will result in higher user satisfaction*. Affordances and feedback are explored in this work due to the expectation that they may improve ease of use, particularly for novice users. So similarly to the previous hypothesis, it was anticipated that users would be happier and more satisfied when the interaction includes affordances and feedback.

4.2.2. Hypotheses from Research Question 2

For the second research question "What types of affordances and feedback are most important for such an interaction?" two hypotheses were tested as well. The new nature of this evaluation meant that the predictions made by the hypotheses were much less certain than those for research question 1.

H3: *Affordances that tell a user how to do something in the interface are more important to the performance of a gesture interaction than affordances indicating what a user can do*. This hypothesis sought to determine whether one type of affordances is more important than another in terms of their effect on interaction performance. The two logical categories of affordances identified are those that indicate to a user (especially a novice one) *what they can do in an interaction*, and those that indicate them *how they can do something* (especially something unfamiliar or ambiguous). This hypothesis compared the impact of these categories of affordances directly to one another, and it was tentatively expected that the former would have a higher positive impact on interaction performance.

H4: *Feedback that indicates system status is more important to the performance of a gesture interaction than feedback acknowledging user actions*. Similarly to H3, this hypothesis sought to determine whether one type of feedback is more important than another in terms of their effect on interaction performance. Again, two logical categories were identified for interface feedback: those that *acknowledge an action taken by the user* (which must necessarily be detected and recognized by the system), and those that provide unsolicited information about *system status*. H4 compared the impact of these categories of feedback, and again the former was predicted to have a higher positive impact on interaction performance.

4.2.3. Discussion of Research Hypotheses

It was fully anticipated that hypotheses 1 and 2 would easily be demonstrated to be true (i.e. the null hypotheses for these will be rejected with statistical significance), so research interest focused on the validity of hypotheses 3 and 4. It would be very surprising indeed to find that novice users do *not* perform better (or prefer) when guidance and feedback are included in a gesture interface. So the original purpose of evaluating hypotheses 1 and 2 was to determine *to what extent* such preference and performance improvement may be observed. User study A evaluated both these hypotheses.

Hypotheses 3 and 4 were evaluated in user studies B and C, respectively. These hypotheses had much higher uncertainty than H1 and H2, as to the author's knowledge the evaluations they propose have not been studied in any domain to date. Determining the types of affordances and feedback that have the greatest impact on a gesture-based interaction was originally expected to be the most important part of this work, as it would provide clear guidance for the design of future gesture interfaces, as well as systems that incorporate them, including assistive robots.

4.3. The Interaction Tasks

To evaluate the research hypotheses, a series of user studies was run in which participants performed two scored interaction tasks using the gesture interface: a sorting task, and a maze task. Both of these tasks are game-like, visual in nature, and could be performed using the simple gestures defined and supported by the described gesture system. Additionally, both of these interaction tasks provided opportunities to present users with all of the included affordance and feedback types in a meaningful way. These affordance and feedback types could then be enabled or disabled per study condition to evaluate their effect on interaction performance.

Participants in the user studies performed both interaction tasks, with the order counterbalanced across participants to reduce ordering effects (e.g. due to learning effects and the inherent difficulty of the tasks, relative to one another). The dominant hand of the participants was also controlled for with counterbalancing. To avoid the effects of the users' hands obstructing their view of the interface, the interaction interface (minus the application area) was mirrored horizontally for left-handed users (see section 3.2.2). All users were incentivized to perform the tasks as quickly as they could, while minimizing errors (which clearly defined for each task). As such, users' performance in these two tasks was principally affected by the study conditions introduced, and could thus be used to evaluate the research hypotheses.

4.3.1. Interaction Task 1: Sorting

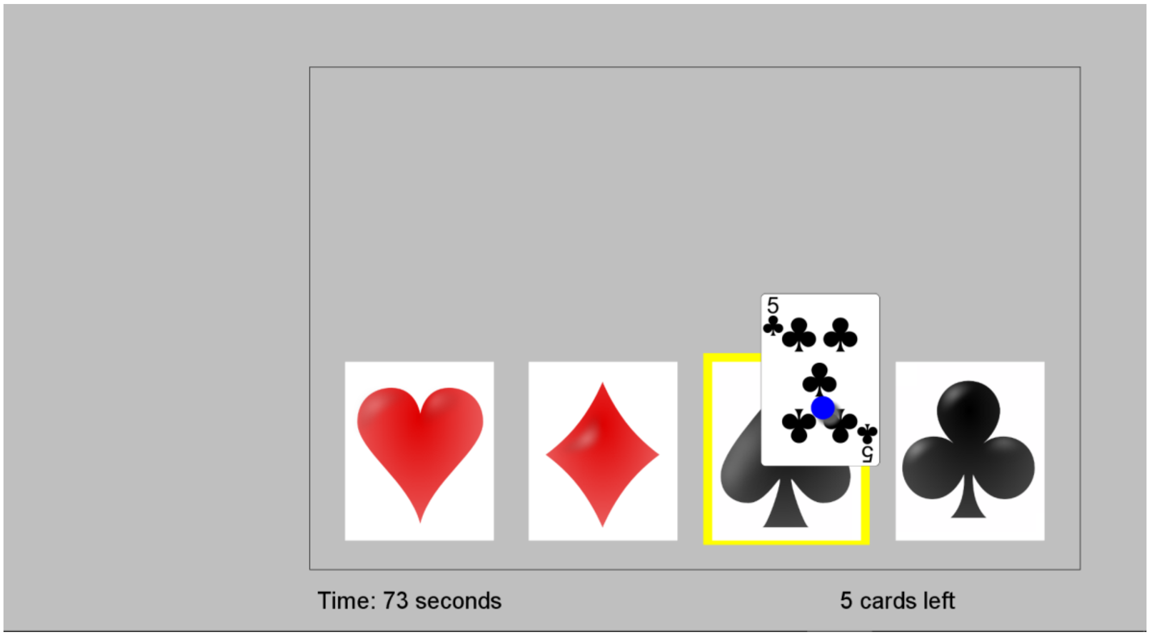


Figure 11. The sorting task as it appeared in the gesture interface used by the study participants. Here, the user is about to drop the five of clubs card into the sorting bin labeled spades, so this will count as a task mistake.

The first interaction task (shown in Figure 11) is a sorting one, in which users were presented with playing cards and had to sort them into bins. The types of bins were: suits (4 bins), color (2 bins), and face vs. number cards (2 bins). To maximize the opportunity for affordance and feedback generation, each card appeared at a random location in the interface, the type of bins to be sorted into were change randomly, and order of the bins was be randomized as well. The location of the bins remained consistent, however.

To complete the sorting task, users had to use the supported hand gestures to select and hold the card, move it across the interface to the appropriate bin, and release the card to drop it there. A new card then appeared to be sorted (possibly into different bin types). The cognitive effort of determining which bin to sort a card into may be considered minimal, so performance was affected primarily by the clarity of using the interface. Users were be asked to perform the task as quickly and accurately as they could until they'd correctly sorted 5 cards, which constituted one round. They were then given a break of 30 seconds, to reduce arm fatigue, then repeat the task for a total of 3 rounds (so that they sorted a total of 15 cards).

The users were timed for each round of 5 cards, and the interface counts a task error every time the user drops a card in the wrong bin (in which case a new card is drawn) or moves it beyond the interaction space (in that case the same card is reset to its starting position). The number of gestures detected was counted as well: every time the user's hand crossed the selection plane in either direction (i.e. every time they performed a tap or withdraw gesture, whether intentionally or not) was counted as a gesture attempt.

4.3.2. Interaction Task 2: Maze

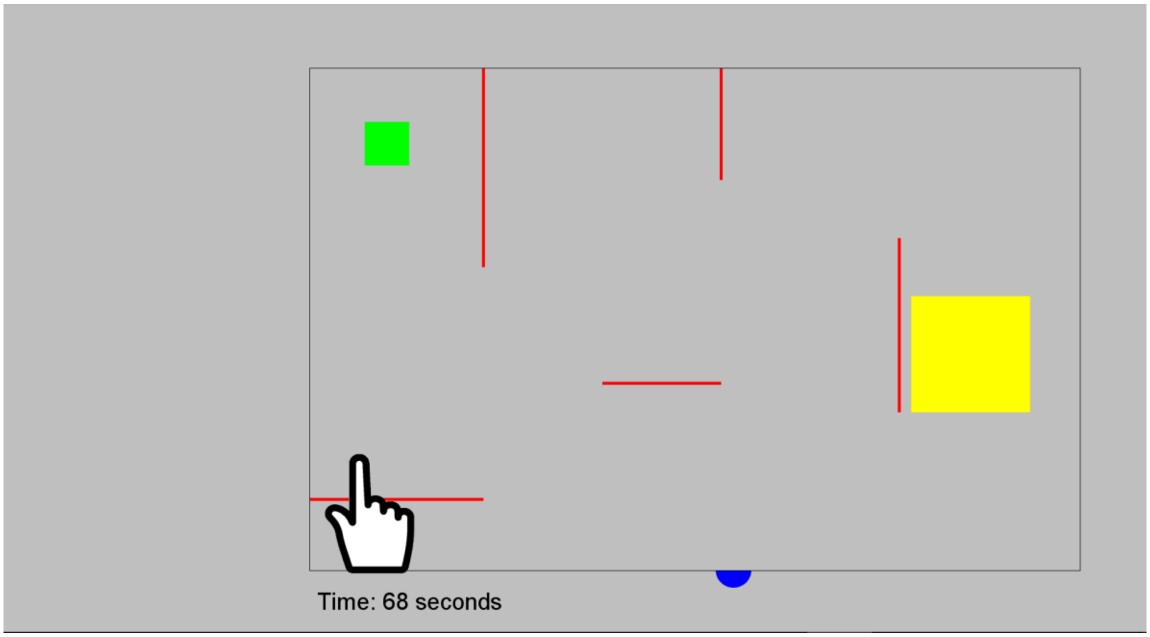


Figure 12. The maze task as it appeared in the gesture interface used by the study participants. The token to be moved is the small green square at the top-left, and the goal into which it should be dripped is the larger yellow square at the right. Here, the user's hand is outside the interaction bounds, so the out-of-bounds feedback is presented.

The second interaction task was a set of simple mazes (one is shown in Figure 12). In this task, the user had to move a token through an easy-to-solve maze to a goal. The principal challenge of this task was that allowing the token to touch the walls of the maze will reset it to the beginning. However, the purpose of this task is not to test coordination or puzzle-solving skill, so the paths in the maze were intentionally made very wide, and the mazes were static and manually fashioned to be very easy to solve.

To complete this task, users needed to use the supported gestures to select the token, move it through the maze to its goal (without touching the walls of the maze), and release the token to drop it there. Users were allowed to drop the item along the way and pick it up again to continue, without the item being reset to the start. When they completed a maze, they were given a break of 30 seconds (to reduce arm fatigue), and then the task was repeat with a new maze, for a total of 3 mazes. The 3 mazes were be defined statically (i.e. they will not be randomized) in order to ensure they are easily solvable.

Like in the sorting task, users were be timed in their completion of each round of this task (i.e. each of the three mazes). A task error was counted every time the user allowed the token to touch the maze walls (which also resets the icon to the beginning of the maze). Dropping the item within the walls of the maze to pause did not count as a task error, but did affect their completion time. Again, each detected tap and withdraw gesture (whether in the goal or not) was be counted as a gesture attempt.

For both interaction tasks, users were informed that the task is timed, and what constituted a task error, but they were not told that gesture attempts would be counted. The 90-second time limit was applied to each round to ensure that the study could proceed in a timely fashion. Users that could not complete a task round within this time limit were counted as taking the maximum possible time.

4.4. Evaluation Measures

Evaluation of the research hypotheses was done using objective, quantifiable performance measures, as well as subjective user feedback about their experience with the gesture system. Four objective measures are used to evaluate interaction performance in the two tasks: task completion time, number of task errors, number of gesture attempts, and the ratio of time that the user's hand is out-of-bounds or not visible. Subjective user feedback was collecting using a two Positive and Negative Affect Schedules (PANAS) – one asking about the user's mood during the gesture interaction, and the asking about their moon immediately after the interaction – as well as a series of 17 questions about their experience with the interface, and their perception of any feedback they observed during the interaction. Responses to questions were collected on a 5-point Likert scale, and qualitative participant responses were also solicited.

4.4.1. Objective Interaction Measures

Four objective measures were used to quantify and assess interaction performance, and they are closely related to the interaction tasks described in the previous section. They are: task completion time, number of task errors, number of gesture attempts, and the ratio of time that the user's hand is out-of-bounds or not visible. Task completion time is the primary measure of interaction performance because both interaction tasks have well-defined completion states, and users are incentivized to complete them quickly. However, to deter hasty or reckless task attempts, users were also incentivized to avoid making task errors. Task errors are defined according to the interaction task: for the sorting task, miss-sorting a card, or moving it out of the interaction area counted as an error; for the maze task, allowing the icon to touch the maze walls counted as an error. Note that the mazes included a fully-enclosing perimeter wall, so moving the icon out of the interaction area implicitly counted as an error there as well.

For both tasks, a gesture attempt was counted for each detected tap and withdraw gesture (whether or not it results in a successful task completion). The number of gesture attempts is a relevant measure of interaction performance because for gestures have no guaranteed response in the application interface. For example the user might try tapping on something that can't be selected. Unfamiliar users may perform more gesture attempts to accomplish a given task, and a mark of a high performance interaction may be accomplishing a task in a minimal number of gesture attempts. Appropriate affordances and feedback may effect a reduction in such attempts, by better informing users of what they can do, what they've done, and how the system is responding.

The out-of-range ratio measure was determined as the ratio of total task time during which the gesture system cannot see the user's hand, their hand is detected but is outside the interaction space, or the user attempts to interact with more than one hand (as this interface is designed for one-handed interaction at the moment). Users were informed prior to the interaction that single-hand gestures were expected, and they had to implicitly discover during the interaction that the hand detection method has limitations (e.g. the bounds of the interaction space). Because of these facts and the fact that users were incentivized to complete the interaction tasks quickly and efficiently, they were by proxy incentivized to keep a single hand visible (i.e. detectable) during the interaction. So a high out-of-range ratio is a proxy indicator of poor interaction performance.

4.4.2. Subjective Measures and Feedback

Subjective user responses were collected in qualitative and quantitative form using post-interaction surveys. The principal forms of user-solicited feedback used for analysis were an evaluation of positive and negative affect (using PANAS – the Positive and Negative Affect Schedule, by Watson et al. [74]) and a series of questions in the post-interaction survey about the user's experience during the interaction tasks. User responses to these questions was collected with a 5-point Likert scale, and could thus be analyzed numerically. These responses and the PANAS scores were the primary data source for evaluation of Hypothesis 2. In addition to questions with quantitative responses, the post-interaction survey also asked open-ended questions about the users' experience and thoughts on the gesture interaction. Appendix G contains the post-interaction survey participants were asked to answer.

4.4.3. Controlled and Uncontrolled Covariates

The pre-interaction surveys used the PANAS questionnaire to users' mood before the interaction, and the pre-survey also included a personality profile using questions from the International Personality Item Pool (IPIP, Oregon Research Institute) to assess the Big-Five Personality factors: Extraversion, Agreeableness, Conscientiousness, Emotional Stability, and Intellect/Imagination [20]. Note that mood and personality are distinct in that mood represents feelings and behaviors expressed over a short time period (up to several days or weeks), whereas personality represents feelings and behaviors expressed over long time periods (months to years). Questions were also included to collect demographic information (age, gender, occupation, etc.) and information about users' experience with computers, robots, and gesture recognition. The participants demographic and experience information, along with their assessed mood and personality were evaluated as covariates at the conclusion of the study to determine if any were confounding variables in the evaluation of the study hypotheses. The pre-interaction survey participants were asked to answer is found in Appendix F.

4.5. User Study

Evaluation of the four research hypotheses was done with a user study that is separated into a series of three between-subjects sub-studies, each testing a pair of study conditions. With the exception of the distinct study conditions they examine, the three sub-studies were identical in format and procedure. For each trial of the user studies, a single participant performed both interaction tasks using the gesture system described in the previous chapter, alone and without assistance or prior instruction. Their performance in the tasks, as well as their subjective responses about the interaction, were used to evaluate the affordances and feedback included in the condition of the study in which they participated.

Participants were recruited using printed flyers, as well as the Texas A&M University main email list (copies provided in Appendices C and D). This mailing list includes a diverse set of students, faculty, staff, and other people associated with the university. A total of 40 subjects completed the user studies and responded to the pre- and post-interaction questionnaires. Only 39 of these though had their interaction data successfully recorded. Participation was incentivized by a drawing for two $25 gift cards, for which all participants are eligible (provided that they attend their scheduled study trial). The interaction tasks required that subjects have fair to good eyesight, be able to sit for at least 10 minutes, and be able to read and understand English.

After recruitment and assignment of study conditions and task order (both of which were be counterbalanced across subjects), participants were scheduled into time blocks of 30 minutes each. Completion of both interaction tasks took up to 10 minutes per participant. Instructions to the participants, as well as pre- and post-interaction surveys took up to an addition 15 minutes to complete. And the system setup and restoration time for each participant was less than 1 minute. The remaining time was used as a buffer to allow for late participant arrivals, slower-than-typical interaction times, and to verify recordings and logs of interaction data. Participants who arrived too late for this accommodation were be asked to reschedule.

For all participants, the interaction involved sitting at a standard computer display in an office-like setting, with a depth sensor (the Intel/Creative Senz3D) that resembles a webcam facing them. Prior to the interaction, they were provided with an informed consent document (found in Appendix E) describing what they can expect from the study, and any questions they had were answered (provided that they didn't interfere with the study conditions or results). They were then instructed that the interface before them supports single-hand gesture interaction (but with no further details about the gesture system), and they were given a high-level description of the two interaction tasks. Finally, they will be instructed that they should perform the tasks as quickly as they can, and avoid task errors (which will be described for each task). To incentivize their performance, additional $5 gift cards were be awarded to top performances in each category.

This user study was approved by the Institutional Review Board (IRB) of Texas A&M University (TAMU IRB reference number IRB2015-0637D). The approved materials includes the study protocol, as described in this section, recruitment material (a flyer and an email prepared for distribution), a consent form for participants, and pre- and post-interaction questionnaires. All participant interaction data remains confidential and does not include (nor is connected in any way to) any personally identifiable information. Signed consent forms were immediately separated from all collected interaction data, and are kept in a secure location accessible only to research staff. Interaction data includes only: interaction performance measures, raw sensor data (in the form of depth maps), and participant questionnaire responses. The interaction data for each participant was anonymized, and was stored and referenced only by a unique code.

4.6. Sub-studies and Study Conditions

The overall user study consisted of three sub-studies, conducted concurrently: Study A evaluated hypotheses H1 and H2; Study B evaluated hypothesis H3; and Study C evaluated hypothesis H4. The three sub-studies used the same recruitment pool to enroll participants, though participants were each enrolled in a single sub-study, and they were not be told ahead of time which it was. From the participants' perspective, a single study was conducted, with no distinct sub-studies. The same gesture system was used for all three sub-studies, and participants in each completed both defined interaction tasks. Because the interaction tasks are both susceptible to learning effects, and it would be very difficult to control for task difficulty across conditions for a single user, all three of the sub-studies were between-subjects.

A total of six study conditions were evaluated in these experiments, exactly two of which are assigned to each of the three sub-studies, with no repetition of conditions. This allowed binary hypothesis testing to be done for each of the four research hypotheses, for direct and clear comparison. Objective interaction performance measures was compared for the pair of conditions in each sub-study (to evaluate hypotheses H1, H3, and H4), and subjective responses was compared as well for the first sub-study (to evaluate H2).

4.6.1. Study A: Conditions 1 and 2

Study A tested hypotheses H1 and H2 using a direct comparison of two study conditions; comparison was done with objective interaction performance measures and subjective participant responses. Recall that hypotheses H1 and H2 are: *Including visual affordances and feedback will result in better interaction performance*, and *Including visual affordances and feedback will result in higher user satisfaction*, respectively. Both hypotheses were tested by comparing the gesture system with and without the corresponding types of affordances and feedback; these modifications to the gesture system are organized into study conditions 1 and 2.

Condition 1: *No explicit affordances and only required and application feedback*. In this study condition, the gesture interface remained intact for participants to use in the interaction tasks. However the designed affordances and feedback methods were not included, with two exceptions: the cursor and the application feedback. The cursor is the minimal required feedback needed to allow the interaction to take place. This gesture interface is driven primarily by pointing gestures, which are universal in human-human interactions in part because of people know the other party can see their hands, and thus know when and where they are pointing. To maintain this universality in a human-machine interaction, the same real-time visibility must be supported by the other party. In this case, that means letting the user know where the system thinks they are pointing, which is done with a cursor. Additionally, application feedback remained, as it is implicit to the interaction task. So for example, if the user moved a card in the interaction space for the sorting task, the card *will* actually move.

Condition 2: All affordances and all feedback. In this study condition, all the affordance and feedback methods described the System chapter were included in the gesture interface for use by participants in both interaction tasks. This included a total of six types of affordances, and nine types of feedback. Hypotheses H1 and H2 predicted that this condition would outperform Condition 1 in the objective and subjective evaluation measures.

4.6.2. Study B: Conditions 3 and 4

Study B evaluated hypothesis H3 by comparing objective performance measures for two study conditions. Hypothesis H3 predicted that: *Affordances indicating how to do something in interface are more important to interaction performance than affordances indicating what the user can do*. To evaluate this hypothesis, interaction performance was compared in two study conditions (Conditions 3 and 4), each using a version of the gesture interface that includes only one or the other of these categories of affordances. Both interface versions included all the feedback of Condition 2, though.

Condition 3. *All feedback, but only affordances indicating what the user can do*. In this study condition, the only type of explicit affordance included in the gesture interface were those that a user may perceive as classical action potentials – that is, only clues telling the user what they can do in the interaction. The affordances that were included in this condition were: indications to the user that the interface supports hand gesture interaction, indications that it supports pointing gestures, indications that an item in the interface can be selected, and indications that an item can be moved.

Condition 4: *All feedback, but only affordances showing the user how to perform gestures*. In this study condition, the only type of explicit affordance included in the gesture interface are those serve to teach or show the user how to perform a [potentially unfamiliar] gesture. Two affordances are included in this category and therefore in this study condition: showing the user how to select an item with tap-to-select gesture, and showing the user how to release an item with a withdraw-to-release gesture. The affordances included in Study B's two study conditions form mutually exclusive sets.

4.6.3. Study C: Conditions 5 and 6

Study C is the final part of the overall user study, and evaluated hypothesis H4, again by comparing objective performance measures for two study conditions. Hypothesis H4 predicted that *Feedback indicating system status is more important to interaction performance than feedback acknowledging user actions*. Evaluation of this hypothesis was done by comparing interaction performance in Conditions 5 and 6. Each of these conditions included all affordance types, but only those feedback types in one or the other of the two feedback categories.

Condition 5: *All affordances, but only feedback acknowledging user actions*. In this study condition, the gesture interface the participants encountered included only visual feedback that acknowledges to users that the gesture system has detected and recognized a particular action they've performed. Four types of feedback were included in this category and thus formed part of the gesture interface in this study condition: feedback acknowledging that the user has raised their hand into view, feedback acknowledging that the user is pointing within the interaction space, acknowledgement that the user has performed a tap gesture, and acknowledgement that the user has performed a release gesture.

Condition 6: *All affordances, but only feedback indicating system status*. In this final study condition, the gesture interface included only feedback that provided the user with unprompted information about the status of the gesture system. Two types of feedback are included in this category and study condition: feedback indicating to the user that their hand is not visible to the system, and feedback indicating that their hand is outside the bounds of the interaction space. These are in contrast to the types of feedback included in Condition 5, and again the feedback included in the two conditions compared in this sub-study form mutually exclusive sets, however the two exceptions from Condition 1 apply here as well. Feedback that acknowledges pointing gestures performed by the user – a cursor – must still be included here, despite being outside of the defined category. And the application feedback, being an implicit part of the application interface, must also remain present.

4.7. Summary and Discussion of Experiments

The experiments described in this chapter were designed to answer two research questions about the effect of affordances and feedback on a gesture interaction, by testing four derived research hypotheses. The gesture system and interface described in chapter 3 was used in a 40-person user study in which participants used hand gestures to perform two game-like interaction tasks. Six distinct study conditions were designed to directly test the research hypotheses; the conditions were distinguished by the types of affordances and feedback they included or omitted. The study conditions were compared using objective measures of interaction task performance as well as subjective user responses about their experience in the interaction. The demographics, experience, and personality of study participants were assessed to search for possible covariates that may have influenced study results. The results of these experiments, and their analysis, are the subject of the following chapter.

The vision-based hand detection method described in Appendix B remained part of the gesture interaction system used in these experiments, but its performance was not formally evaluated within the scope of this dissertation. This is due in part to a commitment to the importance of formally understanding the role of affordances and feedback in a gesture interaction, rather than focusing on marginal algorithmic improvements. More practically though, there was no effective way to establish ground truth information during the user study interactions against which to compare hand detection performance. However, the sensor data collected during the user studies will be used after the completion of these experiments to evaluate and improve upon the hand detection method described. It was nonetheless included in the gesture system because the interaction conditions used for evaluation (namely users completely unfamiliar with and unprepared for the interaction) may have been problematic for simpler heuristic based methods.

5. EXPERIMENTAL RESULTS AND ANALYSIS

Hypotheses, Study and Condition		Affordance types		Feedback types	
		What can be done	*How to do something*	*Acknowledge user actions*	*System status*
Hypotheses 1&2 Sub-study A	A1			*	
	A2	✓	✓	✓	✓
Hypotheses 3 Sub-study B	B1	✓		✓	✓
	B2		✓	✓	✓
Hypothesis 4 Sub-study C	C1	✓	✓	✓	
	C3	✓	✓	*	✓

Table 2. Reference table showing the types of affordances and feedback included in each study condition. The (*) under feedback *acknowledging user actions* for conditions A1 and C3 indicates that though the intention is to omit this type of feedback, exceptions had to be made to include the cursor feedback and the implicit application feedback. So these conditions include only minimal feedback in this category. Sub-study A evaluated hypotheses 1 and 2, and Sub-studies B and C evaluated hypotheses 3 and 4, respectively.

This section presents the results of the experiments described in the previous section, and analyzes those results to assess the presented hypotheses and answer the core research questions of this dissertation. Additionally, covariate data that was collected in the user studies is analyzed and assessed for potential confounding effects on the experimental results. Table 2 provides a quick reference showing which affordances and feedback are included in each hypothesis and study condition.

All hypothesis testing was done with Student's unpaired t-tests with independent variances. And all reported p-values are for 2-tail tests for means. 1-tail tests would of course be twice as sensitive to differences and would result in p-values that are half of the values reported, suggesting greater statistical significance. However, reporting 1-tail p-values would ignore the possibility that the true effect is in the opposite direction of what is supposed – which is a very real possibility, as evidenced by some surprising results.

With 42 typical values per condition tested (7 subjects, 6 tasks each), the t-tests are robust to non-normally distributed data. Nonetheless, the hypothesis tests were re-run with normal-transformed data (using Box-Cox transformations with derived λ-parameter values -0.2006, -0.4241, 0.4667, and 0.3925 for the four measures, respectively) to determine whether any non-normality in the distribution resulted in artificially low p-values. The p-values for the normal-transformed data were lower than the original p-values for the majority of conditions and measures, and exceptions to this rule were very small (max p-value difference of 0.0031 for two primary measures). The conclusion with respect to the normality of the data distributions is that the true difference in means is at least as statistically significant as reported.

5.1. Performance Trends Across All Conditions

Before presenting results for the three sub-studies and the hypotheses they evaluate, an overview of average subject performance across all study conditions is presented here for each of the four measures used in these experiments. These results serve to provide an appropriate sense of scale and variance for the quantitative measures used to evaluate the actual study conditions.

Participants in all six study conditions each completed six interaction tasks in the experiments. Three tasks of one type (either the maze task, or the card-sorting task) were completed first, followed by three tasks of the other type. The starting task for each participant was counterbalanced to ensure balanced numbers of each task order per each study condition. One task may have been inherently more challenging than the other (as discussed in section 4.4.3 on covariates), but across all study conditions and task orders, participants on average improved through the course of their six tasks.

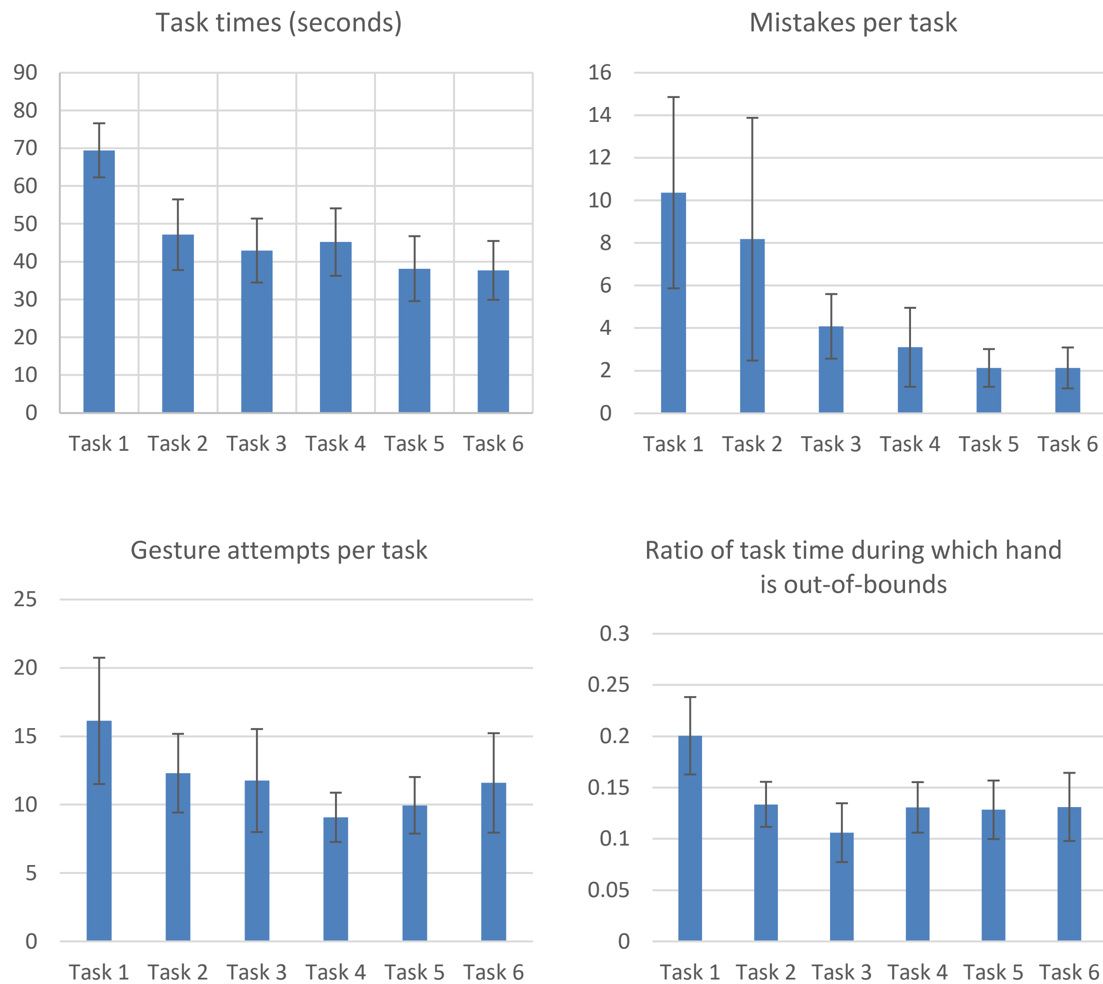


Figure 13. Experimental results across all study conditions, demonstrating performance in all four quantitative measures collected. Error bars denote 95% confidence intervals for each average value presented. Note that all measures were designed so that lower values indicate better performance.

The four bar charts in Figure 13 show average values for each task (1 – 6) for all participants, measured by each of the four quantitative measures described in section 4.4.1. Error bars indicate 95% confidence intervals for these averages. The trend of improvement over the course of the six tasks in evident in all four measures. In particular, it is clear that the most pronounced improvement is between task 1 and task 2. Thereafter, the improvement stagnates for most measures. This divide may occur after task 2 for the “Mistakes per task” measure, but the high uncertainty in average value for task 2 leaves this possibility in question.

The conclusion regarding these experiment-wide results is that there is indeed a separation between the participants’ performance on task 1 and on the remainder of the tasks thereafter. As this research is about interface feature effects on the performance of *novice* users (that is, users who have never before seen the interface nor have any description of preparation for it) it becomes important to evaluate “first-contact” performance as well as overall performance for each participant. Therefore, in the upcoming performance results assessing hypotheses 1, 3, and 4, results will be presented and analyzed for each participant overall, as well for their performance on task 1 only and on tasks 2 – 6 only. This distinction will help to see whether an observed data trend occurs immediately, or only after initial exploratory interaction.

5.2. Results for Hypothesis 1

This hypothesis compares “all affordances and feedback” to essentially “no affordances nor feedback” (only cursor feedback and implicit application feedback remain), and the clear expectation was that the condition with all affordances and feedback (A2) should soundly beat the condition without them (A1) in most, if not all measures. The actuality was therefore a significant surprise, as the exact opposite effect was seen.

5.2.1. Performance of Conditions A1 and A2 in All Tasks

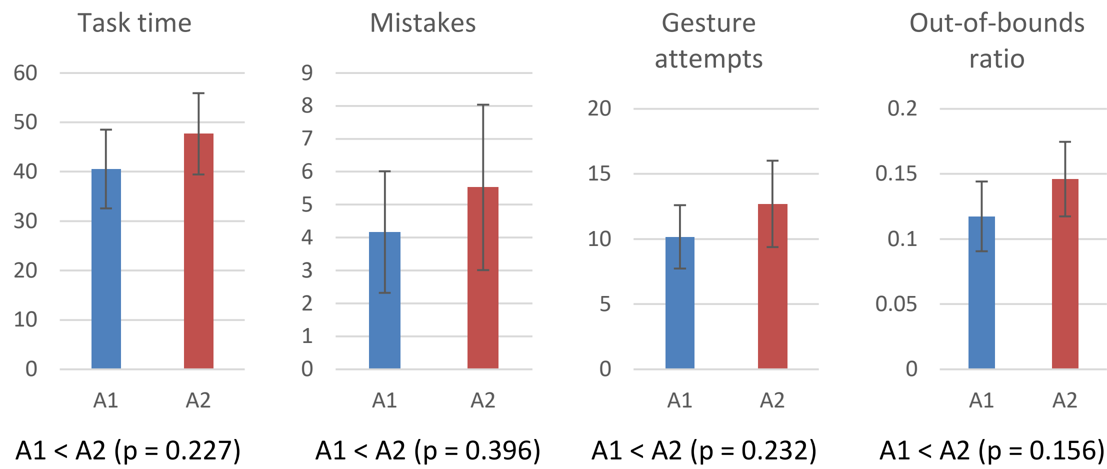


Figure 14. Performance results for hypothesis 1, comparing conditions A1 and A2 on all tasks. All values are averages per task. Error bars show 95% confidence intervals. Statistical p-values are for 2-tail unpaired Student's t-tests with independent variances.

As can be seen in Figure 14, in all four measures condition A1 (N = 7) outperformed condition A2 (N = 7). That is to say, users who interacted with an interface lacking all affordances and most feedback performed better in the two interaction tasks than users who interacted with the interface including all affordances and feedback. This is in stark opposition with the hypothesis and expected results. In fact, testing the *reverse* of the hypothesis (that is, test whether A1 < A2 for all measures), finds remarkably compelling statistical evidence that A1 did indeed achieve faster times, fewer mistakes and gesture attempts, and less time spent with hands out-of-bounds than A2.

These highly surprising results introduce an important new question: How is it possible that an interface with no affordances and only minimal feedback lead to better interaction task performance from complete novices than the same interface with all the designed affordances and feedback in place? How reliable are these results? The results are not statistically significant (using the α = .05 significance level chosen before the experiments were begun), yet they are highly compelling. To investigate the source of this surprising result, the results are now analyzed separately for the first task alone, and for the remaining tasks.

5.2.2. Performance of Conditions A1 and A2 in Task 1 only

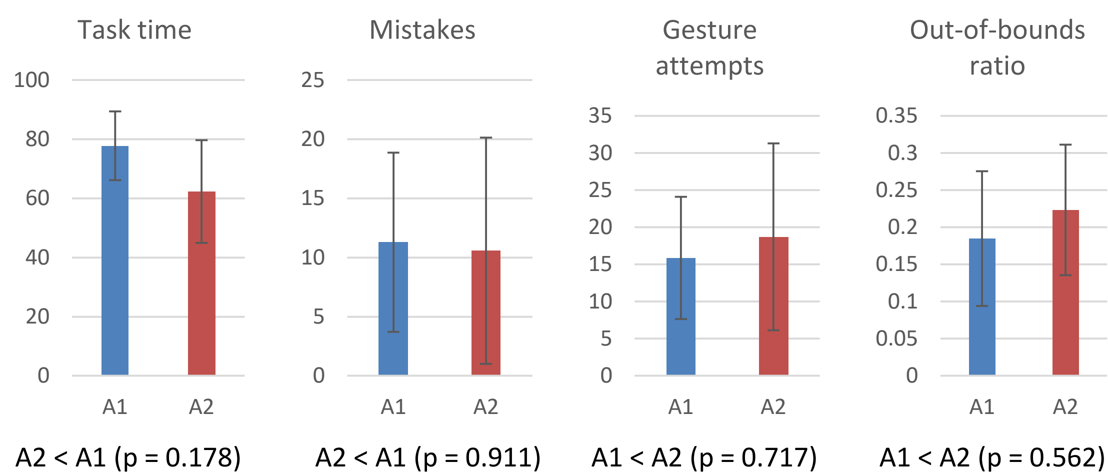


Figure 15. Performance results for conditions A1 and A2 in task 1 only, with 95% confidence interfals and 2-tail p-values for differences in means.

Looking at the results in the first task alone (Figure 15), there is much less significant differences for the last three measures, to the point that we cannot conclude any meaningful difference in the values for these study conditions. However, task time – which is the primary and most important measure – demonstrates an advantage for condition A2 (again, not statistically significant, but compelling nonetheless). This suggests that for task 1 (that is, the very first time the users saw the interface at all) the affordances and feedback were helpful enough to give an early edge to users that had them enabled in the interface with which they interacted. Participants with and without the affordances and feedback both make similar numbers of mistakes and gesture attempts on their first task, and they moved their hands out of bounds a similar amount as well.

5.2.3. Performance of Conditions A1 and A2 in Tasks 2 – 6 only

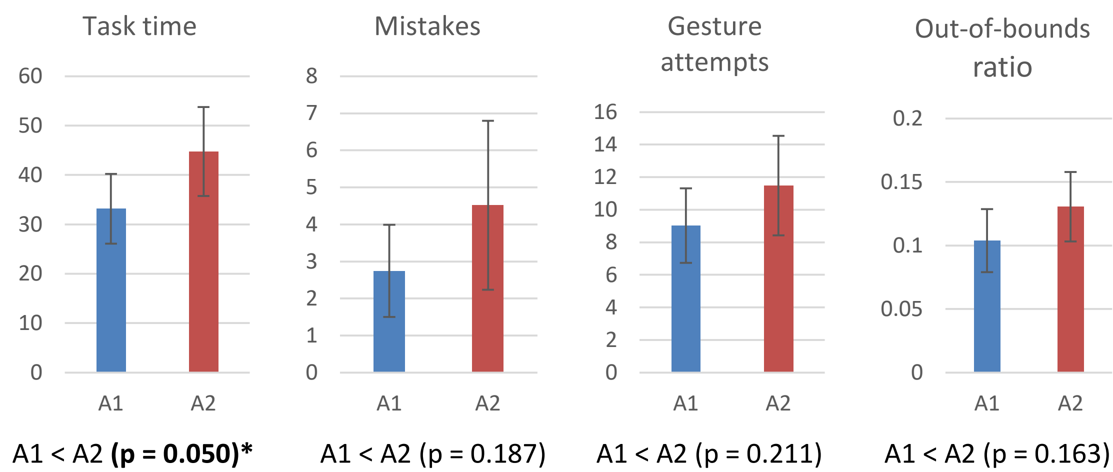


Figure 16. Performance results for conditions A1 and A2 in tasks 2 - 6 only, with 95% confidence interfals and 2-tail p-values for differences in means. Statistical significance (α = .05 level) is denoted in bold and with an asterisk.

Finally, when looking at the results of conditions A1 and A2 on tasks 2 – 6 only (Figure 16), the same performance advantage of condition A1 is present as with the overall results for this hypothesis. In fact, there is now statistical significance in the primary measure (task time). And confidence in the observed difference in the other three measures is improved over the results in all tasks.

This means that *after* participants got their first look at the interface in the first task, participants who interacted with all the affordances and feedback consistently took longer, made more mistakes, took more gesture attempts, and moved their hands out of bounds more often than those who had the "barebones" interface. So after even just the smallest amount of experience with the two tasks in this interface, novice users without access to the affordances and feedback designed for the interface managed to improve dramatically and even outperform users with all the affordances and feedback in their interface.

5.2.4. Summary of Results for Hypothesis 1

	All tasks				Task 1 only				Tasks 2 – 6 only			
	A1	A2	diff.	p	A1	A2	diff.	p	A1	A2	diff.	p
Task time (seconds)	40.57	47.68	7.11	0.227	77.73	62.28	15.45	0.178	33.14	44.76	11.62	0.050
	± 7.98	± 8.20			± 11.63	± 17.38			± 7.06	± 8.99		
Mistakes	4.17	5.52	1.36	0.396	11.29	10.57	0.71	0.911	2.74	4.51	1.77	0.187
	± 1.85	± 2.51			± 7.58	± 9.55			± 1.25	± 2.28		
Gesture attempts	10.17	12.69	2.52	0.232	15.86	18.71	2.86	0.717	9.03	11.49	2.46	0.211
	± 2.42	± 3.31			± 8.23	± 12.56			± 2.28	± 3.05		
Out-of-bounds	0.12	0.15	0.03	0.156	0.18	0.22	0.04	0.562	0.10	0.13	0.03	0.163
	± 0.03	± 0.03			± 0.09	± 0.09			± 0.02	± 0.03		

Table 3. Performance results for conditions A1 and A2 in all tasks, task 1 only, and tasks 2 - 6 only. Values are reported as means for the given condition and measure, with 95% confidence interfals. The lower (i.e. better) value in each condition-measure pair is highlighted in green. Any differences which tested statistically significant (α = .05 level) are highlighted in orange. All p-values are 2-tail.

Table 3 presents the numerical results for conditions A1 and A2 in the three sets of tasks for which the hypothesis was tested, according to the four performance measures collected. These are the values used to produce the charts in Figures Figure 14, Figure 15, and Figure 16. For all measures, a lower value indicates better performance. Though a statistical significant difference was found only in task time for tasks 2 – 6 (favoring condition A1), other measures concur with the finding that condition A1 performed better than A2, in the set of all tasks considered together, as well as in tasks 2 – 6 considered only.

On the surface, the poor performance of condition A2 compared to A1 seems to suggest that something about the affordances and feedback that were selected, designed and implemented in this gesture system were overall detrimental to novice users. Yet the affordances and feedback appear to have been helpful to users in the first task – that is, when they first encountered the interface. So these results again yield more questions: Why did the performance advantage flip after the first task? Are the results after task 1 due more to some disadvantage in condition A2 or to an advantage in condition A1? These questions are explored in chapter 6.

The conclusion for hypothesis 1 is that it *may* hold true for the very first interaction instance, but there is a *clear and complete reversal* of the hypothesis after users briefly familiarize themselves with the interface. Whether this is due to aspects specific to this interface (e.g. the choice of affordances and feedback included) or to external and more generalizable factors (e.g. learning effects and contradiction of mental models) cannot be assessed in this hypothesis alone, but is rather explored in conjunction with hypothesis 2 and then more fully in chapter 6, in light of all experimental results.

5.3. Results for Hypothesis 2

This hypothesis compares the effects of including "all affordances and feedback" (condition A2) to including "no affordances only minimal feedback" (condition A1, which has only cursor and implicit application feedback) on novice users' satisfaction with the gesture interface. In light of the strong negative results for hypothesis 1, the results of this hypothesis are also investigated to seek supportable answers for the questions raised about the poor objective performance of condition A2.

As with hypothesis 1, the expectation here was that condition A2 should yield higher user satisfaction than condition A1. But the subjective results collected again fail to support the hypothesis, instead favoring condition A1 slightly or else being inconclusive. However, the results do yield some insight into the cause of the inverted results for this and the previous hypotheses.

5.3.1. Positive and Negative Affect

The first and best-established manner which measured users' satisfaction and experience during their interaction with the gesture system was the Positive and Negative Affect Schedule (PANAS) developed by Watson et al. [74]. PANAS is a set of 20 words describing the affect (the mood and emotion) of a person. The words are each scored 1 – 5 by the user, and an aggregate value is taken of the scores for words describing positive and negative affect.

The result of the PANAS surveys is a score for positive affect and a score for negative affect. These two scores are not directly comparable to one another. Rather, the two scores are compared separately for different subjects and groups of subjects. Watson reported an average positive affect score for the general population of $\mu = 29.7$, with standard deviation $\sigma = 7.9$. The corresponding score for negative affect is $\mu = 14.8$, $\sigma = 5.4$. These values are for momentary affect (i.e. rated affect at a given moment). Participants in the user studies were given two PANAS surveys, both after the interaction, but one asking how they felt "during the interaction" and the other asking how they felt "at this moment" (i.e. several minutes after the interaction).

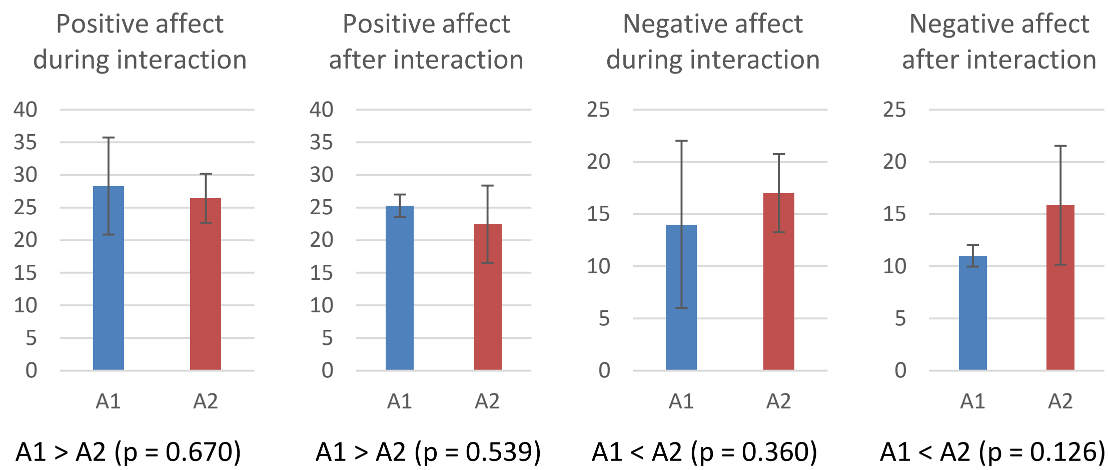


Figure 17. Positive and negative affect for conditions A1 and A2, during and after the interaction, with 95% confidence interfals and 2-tail p-values for differences in means. Higher values are better for positive affect and lower values are better for negative affect.

Figure 17 shows the positive and negative affect scores for participants in conditions A1 and A2, during the interaction, and after the interaction. Support for hypothesis 2 would correspond with higher positive affect scores and lower negative affect scores in condition A2. The results instead show a slight favoring of condition A1, which is more pronounced for negative affect than for positive affect. Although no statistical significance ($\alpha = .05$) was found in the observed differences, "negative affect after the interaction" was closest, with a 2-tail p-value of 0.126.

With regard to attempting to explain the results of hypothesis 1, the statistically negligible difference in positive affect for the two conditions indicates that neither the lack nor inclusion of affordances and feedback produce significantly increased positive emotions in the subjects. However, the apparent difference in negative affect *does* suggest that more negative emotions are caused by condition A2. This finding lends weight to the idea that the *particular* affordances and feedback included in condition A2 (that is, all of the ones designed for this gesture interface) may have been detrimental to user performance and satisfaction.

As a further evaluation, the affect scores achieved in these conditions are compared to the mean and standard deviation established by Watson for the general population. Focusing on affect during the interaction, the mean positive affect scores for conditions A1 and A2 correspond to -0.179 and -0.414 standard deviations from the population mean, while the mean negative affect scores correspond to -0.148σ and $+0.407\sigma$, respectively. These deviations demonstrate that condition A2 produced somewhat lower positive affect (34th percentile) and similarly higher negative affect (66th percentile) in its participants than the momentary means for the general population, suggesting that the interaction was somewhat disliked. Condition A1 placed in the 43rd percentile of positive affect and 44th percentile of negative affect. The low negative affect level in particular suggests that a baseline negative affect for this type of gesture interaction with a novice may actually be as low as or lower than the mean momentary negative affect for the general population. In other words, it is not a forgone conclusion that complete novices have to struggle with or be upset by a computer or robot interface which requires they use gestures to interact.

5.3.2. Subjective Questions about the Gesture Interface

Participants in both conditions were asked 11 questions about the gesture interface they used for the interaction tasks, with responses collected on 5-point Likert scales. Those questions and their responses are divided here into questions with either positive or negative valence, similarly to the positive and negative affect scores.

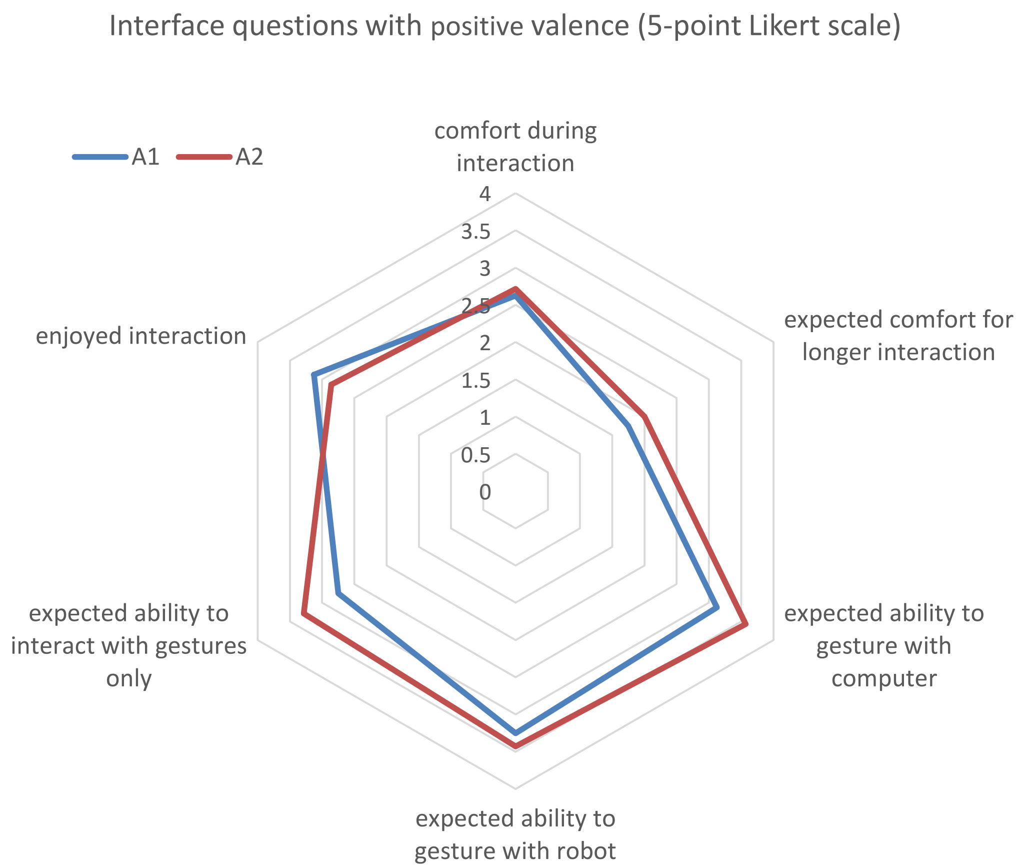


Figure 18. Average responses from condition A1 and A2 participants to positive-valence questions about experience with gesture interface, as recorded on 5-point Likert scales. Higher values are better.

Figures Figure 18 and Figure 19 show radar plots summarizing the responses to the interface questions by participants in the two study conditions. Note that these radar plots are used ONLY to provide a concise summary of the data, and are not meant to be interpreted for their area under the curve. For positive-valence questions, the affordance and feedback condition (A2) has a small advantage in most questions, with the notable exception of the question asking how much users enjoyed the interaction. However, none of the individual questions showed a statistically significant difference in average response. The closest question asked whether participants felt they'd be able to interact with a computer or robot using *gestures only*. Condition A2 had higher average scores on this question than condition A1 ($p = 0.381$), and this is also the only question in this category with an effect size greater than half a point (0.536).

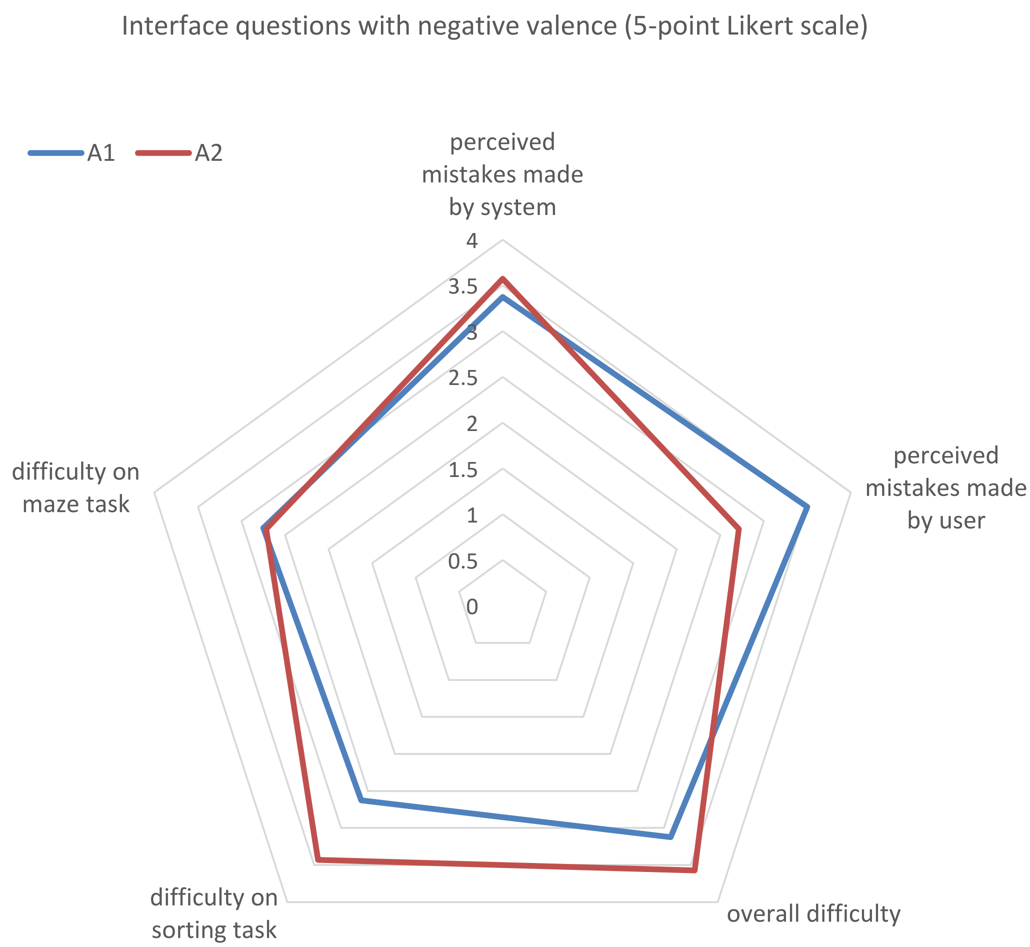


Figure 19. Average responses from condition A1 and A2 participants to negative-valence questions about experience with gesture interface, as recorded on 5-point Likert scales. Lower values are better.

The results on negative-valence questions are less consistent, with condition A2 scoring higher (that is, worse) than A1 on three questions and the opposite being true on 2 questions. Only two questions showed effect sizes larger than half a point, and they also had the lowest p-values in this category: *difficulty on the sorting task* (A1 < A2, $p = 0.205$, effect size 0.803), and *perceived mistakes made by the user* (A2 < A1, $p = 0.066$, effect size 0.786).

This means that users who interacted with the affordances and feedback felt that they made fewer mistakes than those who interacted without them, despite evidence to the contrary (as seen in the results to hypothesis 1). A possible reason for this is that perhaps the added transparency provided by the feedback made users feel that some mistakes they made were not actually mistakes, or at least not significant ones (that is to say, that because the gesture system "caught" the mistake, it wasn't a significant mistake after all).

As shown in section 5.6.1, the sorting task was inherently more difficult than the maze task, so any difficulty caused by the affordances and feedback in condition A2 may have been exacerbated by the task itself. This theory bears out by the large difference seen between the two conditions on perceived *difficulty completing the sorting task with gestures*, the negligible difference on *difficulty completing the maze task with gestures*, and the difference on *difficulty interacting with gestures overall*, which falls precisely between the other two.

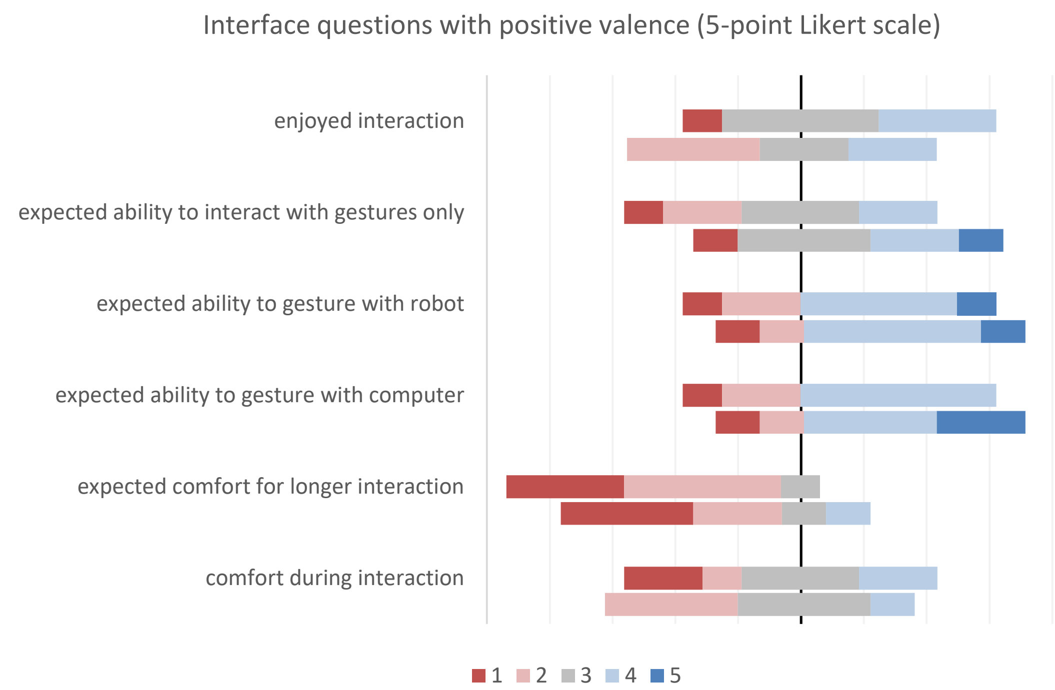


Figure 20. Distribution of responses from condition A1 and A2 participants to positive-valence questions about experience with the gesture interface, as recorded on 5-point Likert scales. Higher values are better. Each question has two bars; the top bar in each pair is the distribution of responses for condition A1; the bottom bar is for condition A2.

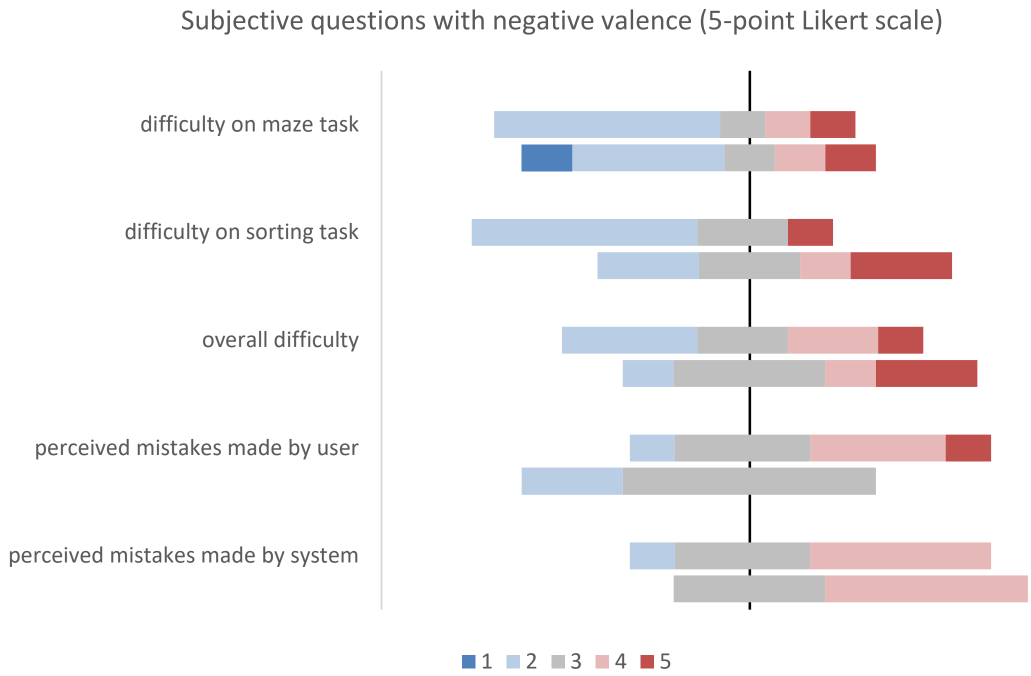


Figure 21. Distribution of responses from condition A1 and A2 participants to negative-valence questions about experience with the gesture interface, as recorded on 5-point Likert scales. Lower values are better. Each question has two bars; the top bar in each pair is the distribution of responses for condition A1; the bottom bar is for condition A2.

Figures Figure 20 and Figure 21 present alternative visualizations of the subject responses to these questions. These diverging stacked bar charts show the distribution of responses per each question, and highlight the inclusion or lack of certain responses to different questions which may be lost when simply looking at averages and even confidence intervals. For example, it is interesting to see that there were no neutral responses to the questions asking participants whether they feel they'd be able to interact using gestures with a computer interface and with a robot. The question somewhat polarized responses, but 50% or more of participants in both conditions responded with a 4 out of 5.

Also, for the question that was closest to statistical significance (*perceived mistakes made by user*, p = 0.066) condition A2 had no negative responses at all while half of the responses in condition A1 were negative. So not only did condition A2 appear to produce fewer perceived user mistakes, but in fact none of the respondents in that condition felt that they made many mistakes.

5.3.3. Subjective Questions about Interface "Feedback"

In addition to questions about their experience with the interface, participants in both conditions were also asked 6 questions about any "visual feedback" they noticed during the interaction. This would refer, of course, to both affordances and feedback, but the word "affordance" is not well known by the general public, whereas "feedback" is, so that term was used. Also, the questions do not specify any particular type or instance of affordances or feedback, but are rather used to assess all the affordances and feedback collectively. These questions too were divided into positive (4 questions) and negative-valence (2 questions).

Note that participants in condition A1 (without any affordances and almost any feedback) were asked these questions as well. Due to the somewhat ambiguous and open-ended term "visual feedback", those participants may have interpreted essentially any part of the interface as a type of "feedback". So these questions can be thought of as comparing the designed affordances and feedback in condition A2 to essentially the rest of the visual interface.

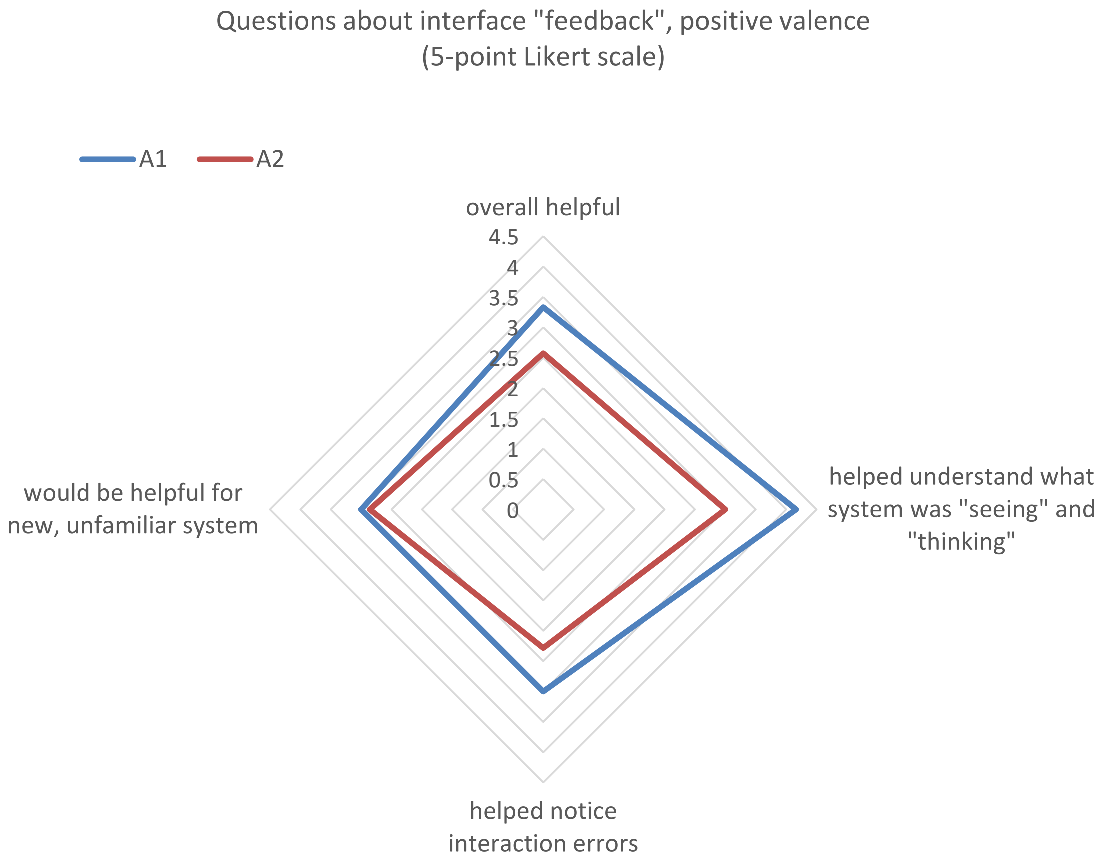


Figure 22. Average responses from condition A1 and A2 participants to positive-valence questions about observed "feedback" in the gesture interface, as recorded on 5-point Likert scales. Higher values are better.

Figure 22 summarizes the responses to the positive-valence questions about feedback, and makes it immediately clear that participants in condition A1 had a higher opinion of any "feedback" they noticed (which is of course ironic as this condition had no explicit affordances and only minimal explicit feedback). Again, the radar plot is used only to succinctly summarize the data, not to imply that the "area under the curve" should be interpreted.

These questions all concerned the helpfulness of the “feedback” and three of the questions had results worth considering (all favoring condition A1): “overall helpful” had a p-value of 0.199 (effect size 0.762), “helped notice interaction errors” had a p-value of 0.329 (effect size 0.714), and “helped understand what system was seeing and thinking” had a p-value of 0.118 (effect size 1.167). The last of these is the most interesting, as condition A2 managed a neutral average score (exactly 3), yet condition A1 beat this score by the largest margin of this set of questions. This is particularly surprising, as one of the purposes of interaction feedback is to provide transparency to the user – which is exactly what this question is asking.

It may be that the “media equation” [52] may be the cause of this surprising result. The media equation suggests that people tend to ascribe greater affect and capability to systems and artifacts than they actually have, especially if the system or artifact is personified or responsive to input. Participants in this study interacted with a responsive interface and they may have assumed deeper meaning in any action or response made by the system (even incidental ones). They might reasonably have been interpreted any such actions as “feedback” provided to clarify the system’s intent or status. Conversely, the lower average score by participants in condition A2 may be due to disillusion when the actual feedback provided by the gesture system breaks the assumptions they made about the state of the system.

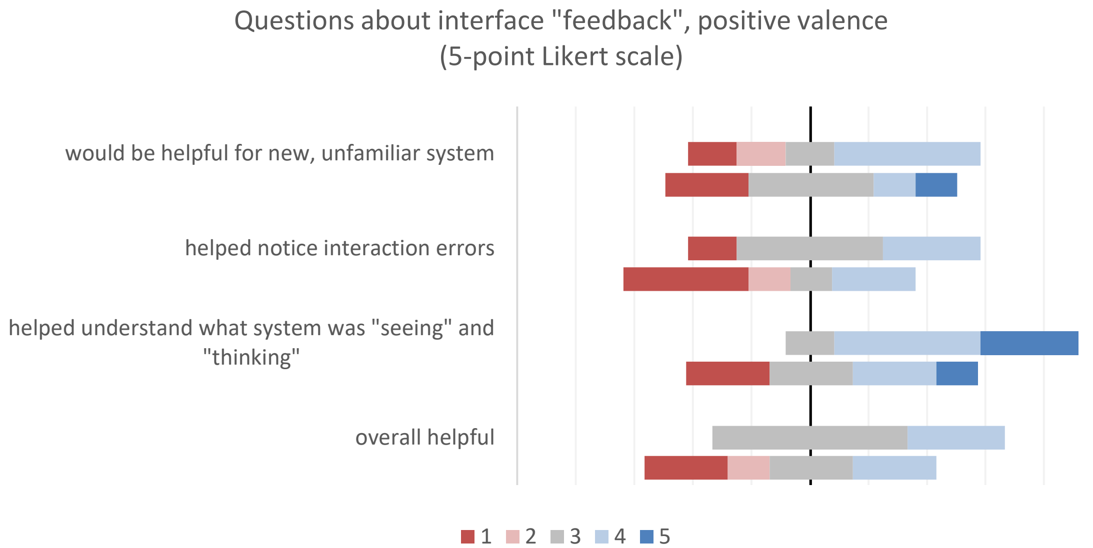


Figure 23. Distribution of responses from condition A1 and A2 participants to positive-valence questions about observed "feedback" in the gesture interface, as recorded on 5-point Likert scales. Higher values are better. Each question has two bars; the top bar in each pair is the distribution of responses for condition A1; the bottom bar is for condition A2.

Looking at the distribution of responses to these questions in Figure 23, there is significant skewing of responses to the questions about whether the "feedback" was helpful overall to the users and whether it helped them understand what the system was seeing and thinking. A portion of condition A2 respondents to both questions found the feedback "not helpful at all" in these respects while not a single respondent in condition A1 rated the feedback's helpfulness as less than neutral.

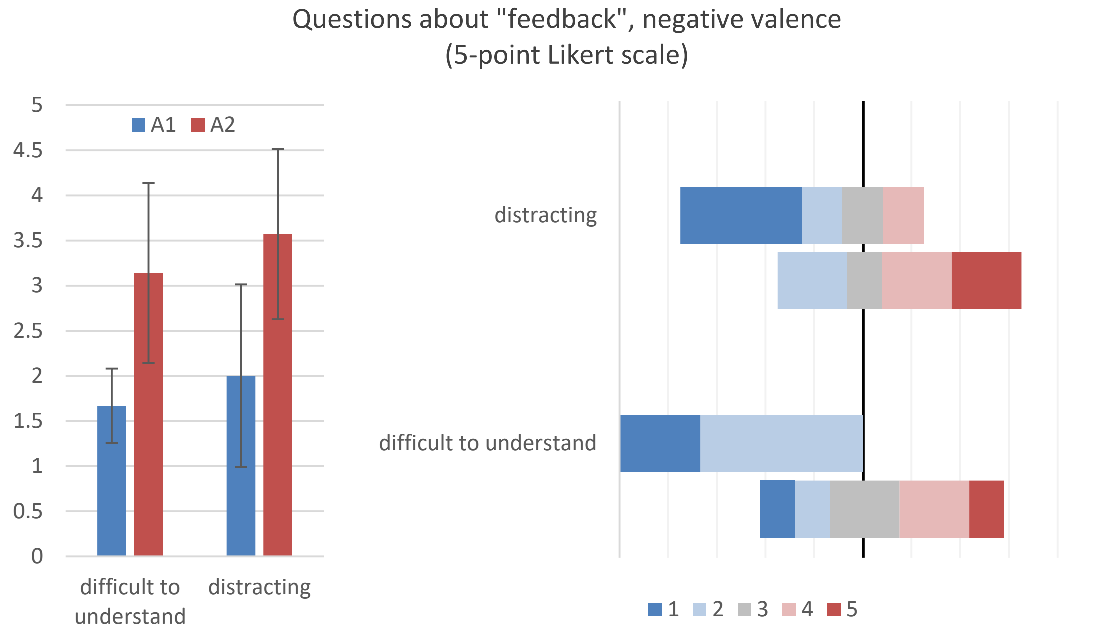


Figure 24. Average (left) and distribution (right) of responses from condition A1 and A2 participants to negative-valence questions about observed "feedback" in the gesture interface, as recorded on 5-point Likert scales. Lower values are better. For chart of response averages, 95% confidence interfals are provided. For chart of response distribution, each question has two bars; the top bar in each pair is the distribution of responses for condition A1; the bottom bar is for condition A2.

Finally, the responses (averages and distributions) to the two negative-valence questions about interface "feedback" are shown in Figure 24. Condition A1 has more favorable responses than A2 for both questions, and its advantage is statistically significant for both as well. We can conclude that the "feedback" in condition A1 is less "difficult to understand" than that in condition A2 ($p = 0.028$, effect size 1.476) and less "distracting" as well ($p = 0.048$, effect size 1.571).

However, the distribution of responses shows that these results are due much more to the "feedback" in condition A1 *not* being distracting/difficult to understand, rather than that in A2 being particularly distracting/difficult to understand. And because we know that the interface in condition A1 had several fewer visual elements than the one in condition A2 (due to the missing affordances and feedback), it makes sense that A1 might appear less distracting and difficult to understand to a novice user. Nonetheless, the greater than neutral values for condition A2 in both questions show that the affordance and feedback implementations were not as seamless as intended from the participants' point of view. Furthermore, the benchmark response values set by condition A1 in these questions – despite the total unfamiliarity of the interface to users – demonstrate that consistently clear and unobtrusive gesture interfaces are possible even for complete novices.

5.3.4. Summary of Results for Hypothesis 2

The evaluation of hypothesis 2 was done with an established measure of positive and negative affect (PANAS) during and after the interaction, as well as positive- and negative-valence questions about participants' subjective experience with the interface as a whole and with the "feedback" present in it. Table 4 shows the numerical results for all these subjective measures.

The PANAS results for the two study conditions were inconclusive with respect to difference in positive affect, but demonstrated lower negative affect for condition A1, especially so after the interaction was complete. And when compared to established general population means, the low negative affect for condition A1 provides a promising baseline for frustration-free gesture interaction with novice users.

The subjective questions about the interface as a whole showed that for positive-valence questions the two study conditions were statistically indistinguishable (though with a very slight apparent edge in favor of condition A2). The interface questions with negative valence showed interesting results in two questions, favoring opposite conditions: participants in condition A2 perceived a greater *difficulty of completing the sorting task with gestures* than those in condition A1, but they also *felt that they made fewer mistakes* in both tasks than participants in A1.

	A1		A2		diff.	p
↑ Positive affect during interaction	28.29	± 7.44	26.43	± 3.75	1.86	0.670
↑ Positive affect after interaction	25.29	± 8.02	22.43	± 3.75	2.86	0.539
↓ Negative affect during interaction	14.00	± 1.71	17.00	± 5.94	3.00	0.360
↓ Negative affect after interaction	11.00	± 1.05	15.86	± 5.70	4.86	0.126
↑ comfort during interaction	2.63	± 0.82	2.71	± 0.56	0.09	0.867
↑ expected comfort for longer interaction	1.75	± 0.49	2.00	± 0.86	0.25	0.616
↑ expected ability to gesture with computer	3.13	± 0.86	3.57	± 1.12	0.45	0.541
↑ expected ability to gesture with robot	3.25	± 0.96	3.43	± 1.04	0.18	0.808
↑ expected ability to interact with gestures only	2.75	± 0.72	3.29	± 0.93	0.54	0.381
↑ enjoyed interaction	3.13	± 0.69	2.86	± 0.67	0.27	0.595
↓ perceived mistakes made by system	3.38	± 0.52	3.57	± 0.40	0.20	0.573
↓ perceived mistakes made by user	3.50	± 0.64	2.71	± 0.36	0.79	0.066
↓ overall difficulty	3.13	± 0.78	3.57	± 0.84	0.45	0.459
↓ difficulty on sorting task	2.63	± 0.74	3.43	± 0.94	0.80	0.205
↓ difficulty on maze task	2.75	± 0.81	2.71	± 1.02	0.04	0.957
↑ overall helpful	3.33	± 0.41	2.57	± 0.94	0.76	0.199
↑ helped understand what system was "seeing" and "thinking"	4.17	± 0.60	3.00	± 1.13	1.17	0.118
↑ helped notice interaction errors	3.00	± 0.88	2.29	± 1.02	0.71	0.330
↑ would be helpful for new, unfamiliar system	3.00	± 1.01	2.86	± 1.08	0.14	0.855
↓ difficult to understand	1.67	± 0.41	3.14	± 1.00	1.48	0.028
↓ distracting	2.00	± 1.01	3.57	± 0.94	1.57	0.048

Table 4. Subjective participant responses for conditions A1 and A2. Values are reported as means for the given condition and measure, with 95% confidence intervals. Favorable values (high or low) for each item are denoted by arrows on the left. The more favorable value in each condition-measure pair is highlighted in green. Statistically significant differences (α = .05 level) are highlighted in orange. Almost statistically significant differences (p < 0.1) are highlighted in blue. All p-values are 2-tail.

In the subjective questions on perceived "feedback" in the interfaces, condition A2 was soundly beaten by A1 on both positive- and negative-valence questions. Although one might have expected that scores for condition A1 would be mostly neutral (since there are no added affordances and feedback apart from the pointing cursor), it surprisingly scored firmly favorable scores on three of the questions (one positive-valence and two negative). The results of the two negative-valence questions are best explained by the simpler and therefore less distracting interface in condition A1. However the "media equation" [52] may be at play in A1's high score on the positive-valence question *helped understand what system was seeing and thinking*.

Returning to the question of why condition A1 performed better than condition A2 in hypothesis 1, the findings relating to hypothesis 2 do in fact shed some light on the matter. First, users perceived that the "feedback" in condition A2 was more distracting and more difficult to understand than that in condition A1. Second, the very fact that users perceived any feedback at all in condition A1 (let alone rated it favorably) may indicate that their mental model of the interaction assumed greater capability to the system than it truly had, especially in terms of its responses and feedback. This idea is further supported by the fact that the same users felt that this "feedback" they perceived helped them understand the present state of the system. So when the system in condition A2 displayed the actual affordances and feedback that were designed for the interaction, they may have contradicted users' high-capability – and possibly very specific – mental models of the interaction, leaving them scrambling to reinterpret the feedback and thus distracting them from the time-sensitive task before them. In short: the performance disparity was due to both internal (distracting/confusing affordances and feedback) and external (media-equation-like expectations, and broken mental models).

Although there is inconsistency in these subjective scores, it is clear that as a whole they do not support hypothesis 2. The affordances and feedback in this interface did not yield measurably higher user satisfaction.

5.4. Results for Hypothesis 3

This hypothesis was tested in sub-study B, comparing the effects of including only affordances which indicate *what can be done* (Condition B1) in the interface, and only affordances indicating *how to do something* (Condition B2). Both conditions B1 and B2 included all feedback in their interfaces, to focus observed effects on the difference in included affordances. The expectation was that condition B2 should result in better interaction performance than condition B1. The results demonstrate support for this hypothesis in the two primary measures, in particular after initial interaction.

5.4.1. Performance of Conditions B1 and B2 in All Tasks

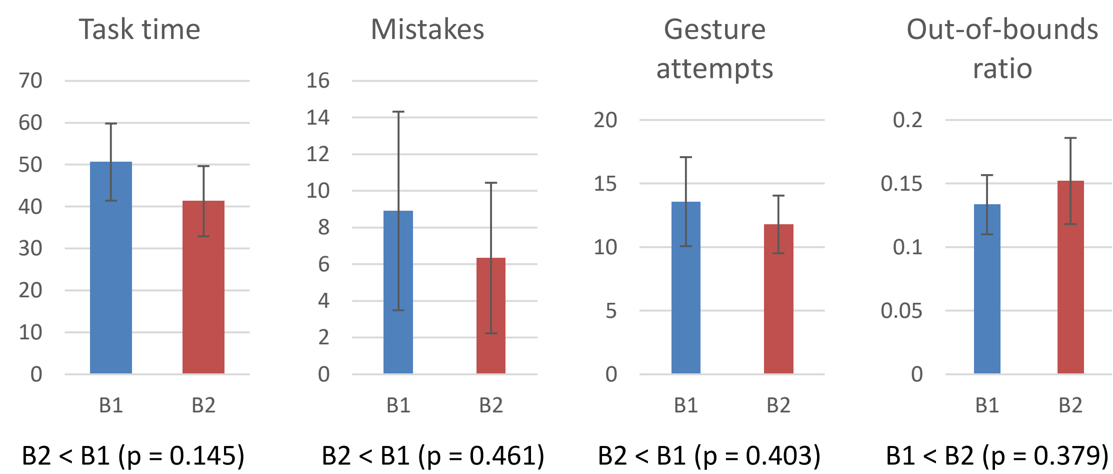


Figure 25. Performance results for hypothesis 3, comparing conditions B1 and B2 on all tasks. Error bars show 95% confidence interfals. Statistical p-values are for 2-tail unpaired Students' t-tests with independent variances.

Figure 25 shows the measured performance results for participants in study conditions B1 (N = 7) and B2 (N = 6). The first three measures show support for the hypothesis that affordances indicating how to do something are more important than affordances indicating what can be done. Interestingly, the ratio of time that the users moved their hands out of bounds is at odds with the other measures. However, there is large variance in some of the measures, especially mistakes, and none of the observed effects demonstrate statistical significance (α = .05 level) so as with hypothesis 1, the results are now broken up to see the effects in task 1 alone, and the remainder of the tasks.

5.4.2. Performance of Conditions B1 and B2 in Task 1 only

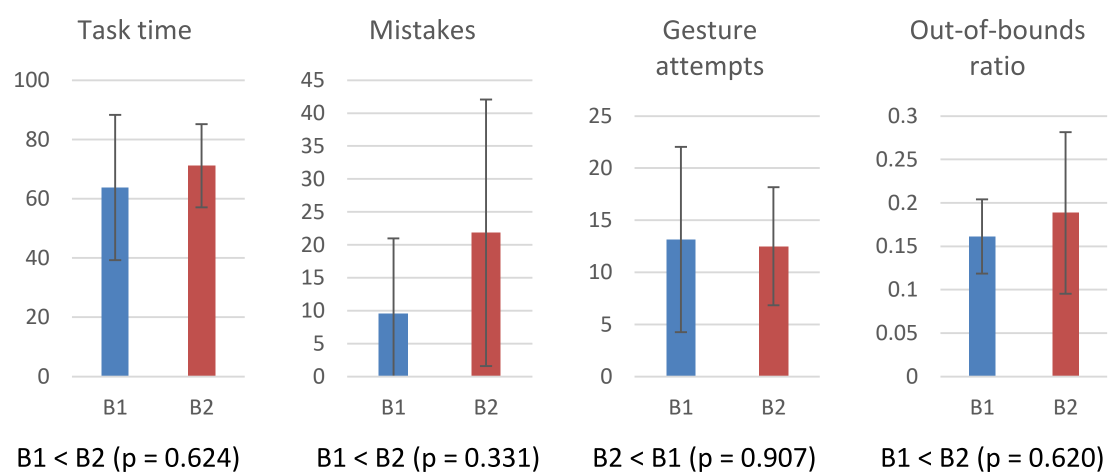


Figure 26. Performance results for conditions B1 and B2 in task 1 only, with 95% confidence interfals and 2-tail p-values for differences in means.

In the first task in sub-study B, the overall advantage seen by condition B2 in the two primary measures is lost and may even have been reversed (Figure 26). The reversal is not conclusive though, given the much lower confidences in the true condition means, due to the smaller sample sizes. However, if true, the reversal on the first task would make sense, as condition B1includes affordances indicating *what a user can do* in the interface, which would believable be more useful at first contact with the interface. In any case, it cannot be said that the *how to do something* affordances in condition B2 provide a performance advantage in the very first task.

5.4.3. Performance of Conditions B1 and B2 in Tasks 2 – 6 only

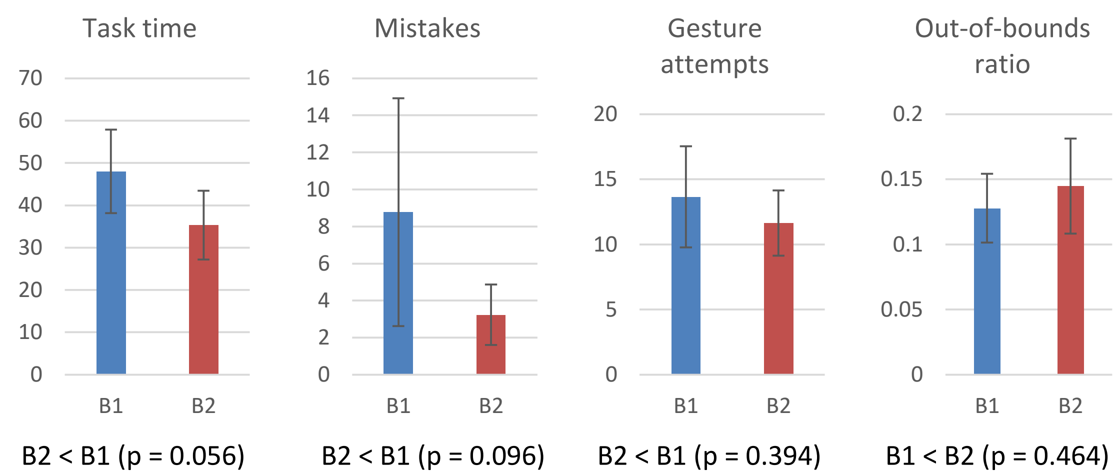


Figure 27. Performance results for conditions B1 and B2 in tasks 2 – 6 only, with 95% confidence interfals and 2-tail p-values for differences in means.

After users had their first interaction with the interface in task 1, the overall trend seen in all the tasks is once again evident in the remaining tasks (Figure 27), in agreement with hypothesis 3. In the first two (and most important) measures, there is now stronger statistical support for the hypothesis that the affordances in condition B2 produce better interaction performance than those in condition B1. The results in the other two measures are less certain though, and once again the out-of-bounds measure disagrees with the overall results.

These results tell us that after initial contact with the interface, the affordances telling users *how to do something* are more important to the performance of the gesture interaction than those telling them *what they can do*. Specifically, study participants whose interface included only the *how to* affordances completed the interaction tasks faster and with fewer mistakes than those whose interface included only the *what can be done* affordances.

5.4.4. Summary of Results for Hypothesis 3

	All tasks				Task 1 only				Tasks 2 – 6 only			
	B1	B2	diff.	p	B1	B2	diff.	p	B1	B2	diff.	p
Task time (seconds)	50.64 ± 9.21	41.30 ± 8.36	9.34	0.145	63.80 ± 24.51	71.13 ± 14.08	7.33	0.624	48.01 ± 9.84	35.33 ± 8.12	12.68	0.056
Mistakes	8.90 ± 5.41	6.33 ± 4.11	2.57	0.461	9.57 ± 11.39	21.83 ± 20.25	12.26	0.331	8.77 ± 6.15	3.23 ± 1.63	5.54	0.096
Gesture attempts	13.57 ± 3.51	11.78 ± 2.27	1.79	0.403	13.14 ± 8.89	12.50 ± 5.67	0.64	0.907	13.66 ± 3.87	11.63 ± 2.51	2.02	0.394
Out-of-bounds	0.13 ± 0.02	0.15 ± 0.03	0.02	0.379	0.16 ± 0.04	0.19 ± 0.09	0.03	0.620	0.13 ± 0.03	0.14 ± 0.04	0.02	0.464

Table 5. Performance results for conditions B1 and B2 in all tasks, task 1 only, and tasks 2 - 6 only. Values are reported as means for the given condition and measure, with 95% confidence interfals. The lower (i.e. better) value in each condition-measure pair is highlighted in green. No condition differences tested statistically significant (α = .05 level). Differences which tested nearly statistically significant (p < 0.1) are highlighted in blue. All p-values are 2-tail.

Table 5 presents the numerical results for hypothesis 3 in four performance measures collected. Conditions B1 and B2 are compared in three sets of tasks: all tasks, task 1 only, and tasks 2 – 6 only. These values were used to produce Figures Figure 25, Figure 26, and Figure 27 in the preceding subsections. These results show support for the hypothesis that affordances indicating *how to do something* (condition B2) are more important to interaction performance than those indicating *what can be done* (condition B1. This conclusion is especially evident in the two primary measures (task time and number of mistakes) in tasks 2 – 6, where the advantage of condition B2 over B1 is almost – but not quite – statistically significant at the $\alpha = .05$ level.

In task 1, the advantage of condition B2 disappears, and it is possible that B1 may be favored. This would be a reasonable conclusion as affordances telling a user *what they can do* in an interface (condition B1) may be immediately more helpful than those telling them *how to do something* (particularly as a novice user might not yet know what that something is). The results show that participants in the two conditions made very similar numbers of gesture attempts in task 1, yet those in condition B2 made many more mistakes (more than twice as many as B1). This leads be to believe that participants in condition B2 didn't know *when* they could perform certain actions in the interface (e.g. when they could drop an item) and this underperformed relative to condition B1.

In the remaining tasks (2 – 6), condition B2 held a decisive advantage over B1, especially in task time (where it was 36% faster than B1 on average) and number of mistakes (less than half the average number for condition B1). This means that after the first task, the *how to do something* affordances resulted in a large performance improvement for novice users of this interface.

The conclusion for hypothesis 3 is it is generally supported overall, and is strongly supported after the initial interaction (especially in the time and number of mistakes measures). So affordances that indicate *how to do something* in an interface are more important to interaction performance than affordances that indicate *what can be done*. For the initial interaction, the results are less conclusive, though it appears that the opposite effect is present.

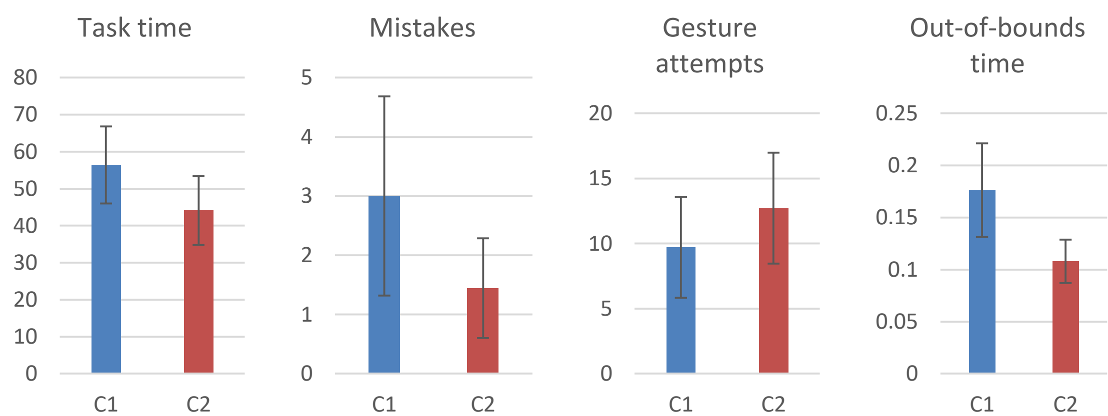
5.5. Results for Hypothesis 4

Hypothesis 4 was tested in sub-study C, which compared the effects of including only feedback which acknowledged user actions (Condition C1) in the interface, and including only feedback which provided system status warnings (Condition C2). Conditions C1 and C2 both had all the affordances in place, to focus any performance differences on the included and excluded feedback. Also, as with condition A1 in sub-study A, condition C2 included the required cursor feedback, even though it falls into the feedback category for C1. The expectation was that condition C2 should achieve higher interaction performance than condition C1. The results demonstrated strong support for this hypothesis in three out of four measures.

5.5.1. Performance of Conditions C1 and C2 in All Tasks

The average performance results for participants in conditions C1 (N = 6) and C2 (N = 6) across all interactions tasks are shown in Figure 28. The results show strong support in three measures (including one that is statistically significant) for the hypothesis that *system status* feedback (condition C2) is more important to the gesture interaction than feedback *acknowledging user actions* (condition C1). The number of gesture attempts favors condition C1 though.

Participants who were only provided interface feedback about *system status* completed the interaction tasks faster, with fewer mistakes, and they moved their hands out-of-bounds less often than participants who were provided with only feedback *acknowledging their actions*. The advantage of condition C2 over C1 in the out-of-bounce measure was statistically significant (α = .05 level) across all tasks. This strong result on the last measure is not surprising, as the system status feedback for this interface consisted of warning users when their hands are outside the bounds of the interaction space, or not visible to the gesture sensor. Nonetheless, that benefit translated into stronger performance in the two primary measures as well.



C2 < C1 (p = 0.089) C2 < C1 (p = 0.111) C1 < C2 (p = 0.311) C2 < C1 **(p = 0.010)***

Figure 28. Performance results for hypothesis 4, comparing conditions C1 and C2 on all tasks. Error bars show 95% confidence interfals. Statistical p-values are for 2-tail unpaired Students' t-tests with independent variances. Statistical significance (α = .05 level) is denoted in bold and with an asterisk.

Regarding the number of gesture attempts per task, the higher performance (i.e. fewer attempts) seen in condition C1 is like due to the specific types of acknowledgement feedback as well. Specifically, condition C1 included feedback indicating when the user has performed a "tap" or "release" gesture. So users without that feedback may have made more inadvertent gestures while completing the tasks.

What is particularly telling about these results is that one secondary measure (out-of-bounds ratio) strongly captured the performance effects of the system status feedback while another (gesture attempts) captured the performance effects of the user-acknowledgement feedback, yet it was the system status feedback which correlated with the results of the two primary measures. This may be further support for the hypothesis that system status feedback is more important.

5.5.2. Performance of Conditions C1 and C2 in Task 1 only

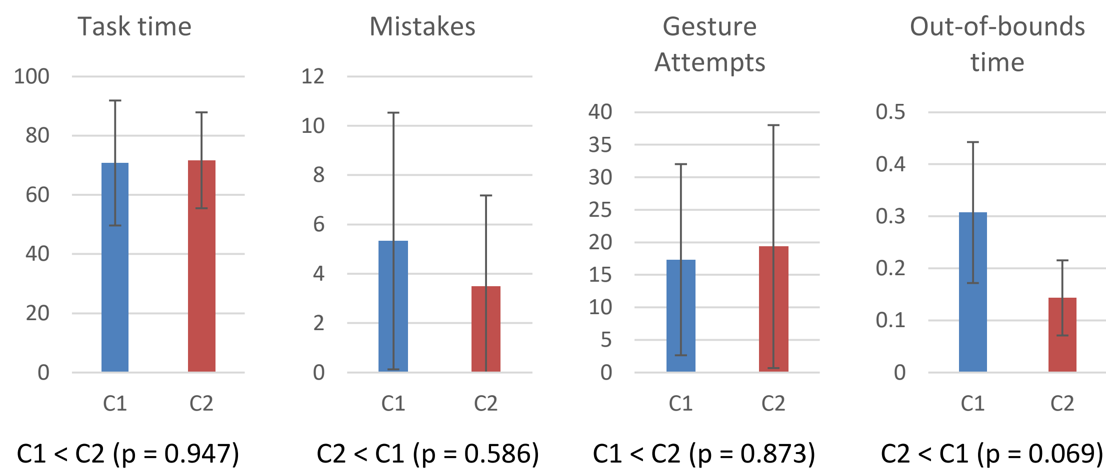
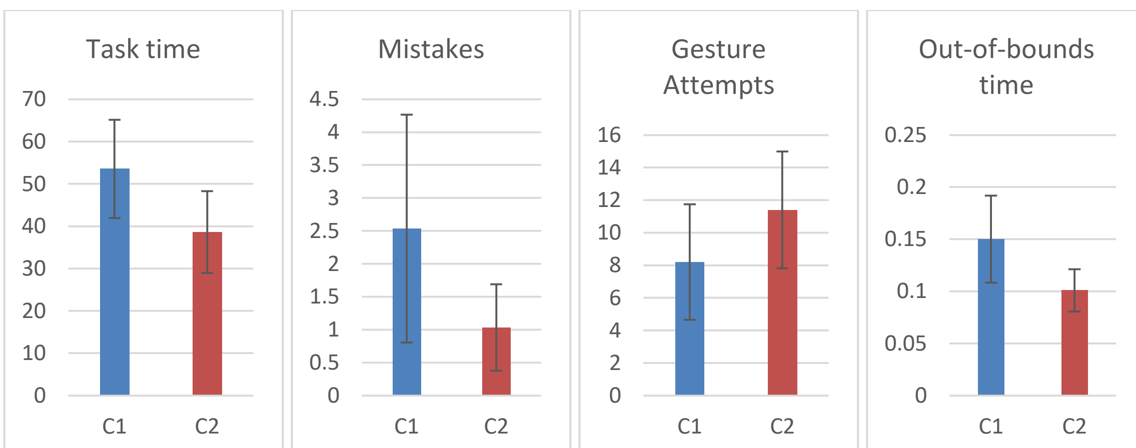


Figure 29. Performance results for conditions C1 and C2 in task 1 only, with 95% confidence interfals and 2-tail p-values for differences in means.

Figure 29, showing the performance results for conditions C1 and C2 in task 1 only, shows that the speed advantage condition C2 held when considering all tasks was actually not present in the first task. The lead in number of mistakes also largely disappears here, and the number of gesture mistakes is now indistinguishable between the two conditions as well. However, the out-of-bounds measure not only favored condition C2 from the very start of the interaction, but the effect size is particularly large (the out-of-bounds ratio for C2 is less than half that for C1) and the confidence in the mean difference is very high from the start (though not quite statistically significant).

The dead heat in time taken on the first task indicates that the system status feedback was not particularly impactful at the very start of the interaction. Similarly, the close results in gesture attempts (and the broad variance for both conditions) suggests that the feedback acknowledging user actions – in particular "tap" and "release" gesture attempts, in the case of this interaction – may not lead to measureable performance differences at the beginning.

5.5.3. Performance of Conditions C1 and C2 in Tasks 2 – 6 only



C2 < C1 (p = 0.057) | C2 < C1 (p = 0.120) | C1 < C2 (p = 0.218) | C2 < C1 **(p = 0.045)***

Figure 30. Performance results for conditions C1 and C2 in task 2 – 6 only, with 95% confidence interfals and 2-tail p-values for differences in means. Statistical significance (α = .05 level) is denoted in bold and with an asterisk.

Finally, looking at the results for conditions C1 and C2 in the remaining tasks (Figure 30), the measures once again agree with the overall trends for the six tasks considered together. Specifically, the task time, number of mistakes and out-of-bounds measures all strongly favor condition C2, thus supporting hypothesis 4. However, the number of gesture attempts again favors condition C1 instead.

Relative to all the tasks considered together, the results in tasks 2 – 6 show stronger statistical confidence in the mean difference for the first three tasks, but again none quite reaches statistical significance (α = .05 level). On the other hand, the out-of-bounds measure is again statistically significant, but its corresponding p-value is now much closer to the borderline case.

5.5.4. Summary of Results for Hypothesis 4

	All tasks				Task 1 only				Tasks 2 – 6 only			
	C1	C2	diff.	p	C1	C2	diff.	p	C1	C2	diff.	p
Task time (seconds)	56.42 ± 10.39	44.11 ± 9.35	12.31	0.089	70.77 ± 21.08	71.70 ± 16.20	0.93	0.947	53.55 ± 11.60	38.60 ± 9.66	14.95	0.057
Mistakes	3.00 ± 1.68	1.44 ± 0.84	1.56	0.111	5.33 ± 5.20	3.50 ± 3.68	1.83	0.586	2.53 ± 1.73	1.03 ± 0.65	1.50	0.120
Gesture attempts	9.72 ± 3.89	12.72 ± 4.26	3.00	0.311	17.33 ± 14.69	19.33 ± 18.68	2.00	0.873	8.20 ± 3.55	11.40 ± 3.58	3.20	0.218
Out-of-bounds	0.18 ± 0.04	0.11 ± 0.02	0.07	0.010	0.31 ± 0.14	0.14 ± 0.07	0.16	0.069	0.15 ± 0.04	0.10 ± 0.02	0.05	0.045

Table 6. Performance results for conditions C1 and C2 in all tasks, task 1 only, and tasks 2 - 6 only. Values are reported as means for the given condition and measure, with 95% confidence interfals. The lower (i.e. better) value in each condition-measure pair is highlighted in green. Statistically significant differences (α = .05 level) are highlighted in orange. Mean difference which are almost statistically significant ($p < 0.1$) are highlighted in blue. All p-values are 2-tail.

The numerical results for the measurements taken to test hypothesis 4 are presented in Table 6. The values in this table are the same that were used to produce Figures Figure 28, Figure 29, and Figure 30. The results together show that the measures of number of mistakes, number of gesture attempts, and out-of-bounce ratio consistently favored conditions C2, C1, and C2, respectively. Meanwhile, the task time measure went from not measuring either condition in the first task to favoring condition C2 in the remaining tasks and in all tasks considered together.

The out-of-bounds measure was strongly correlated with the feedback included in condition C2, and thus it remained the most consistent discriminator between the two conditions in all the task sets considered. The effect size seen for this condition was accordingly large: 64%, 114%, and 50% improvements by condition C2 over C1 in the three task sets that were evaluated.

Interestingly, the out-of-bounds effect size is largest in the very first task, though this is more due to condition C1's particularly poor initial performance than to special early achievement by condition C2. In fact, condition C1 halved its average out-of-bounds ratio after the first task. This means that even without the status feedback participants were able to learn the interaction bounds over time. Nonetheless, condition C2 outperformed C1 by significant margins across the board in this measure, so the status feedback undeniably resulted in better performance in this regard.

The sizeable advantage held by condition C1 in the number of gesture attempts (31% effect size across all tasks) is best explained by correlation as well. The feedback which acknowledged user actions in condition C1 included visual cues notifying the user when "tap" and "release" gestures were detected, so it stands to reason that the lack of these in condition C2 may have led to an increased number of inadvertent gestures performed, as those users would not necessarily know when they performed a gesture by accident.

Significantly, the *system status* feedback in faster overall task times (and especially so after the first task) and fewer mistakes despite a consistently higher number of task attempts. As the number of gesture attempts dropped substantially with the inclusion of *acknowledgement* feedback, yet did not correlate with the primary measures (time and mistakes), this serves as further support that system status feedback is more impactful on gesture performance.

The conclusion for hypothesis 4 is that it is strongly supported both overall and after initial interaction, but is only partially supported at first. This means that feedback that provides *system status* is more important to gesture performance than feedback which *acknowledges user actions*. However, the second type of feedback is important to other measurable aspects of an interaction – in this case, the number of gesture attempts.

5.6. Analysis of Covariates

In addition to the four objective performances measures and the subjective responses collected from participants after their interaction with the gesture system, demographic information was also collected about the participants and information about their experience with robots, video games, and other gesture interfaces to test for effects of these covariates on the interaction performance results. This section presents the analysis of those covariates and discusses their effects on the hypotheses results. The analysis is limited to the measured gesture performance in the two primary measures (task time and number of mistakes)

5.6.1. Controlled Covariates

Two binary-valued covariates were explicitly controlled for in the experiments by counterbalancing in the study condition pairs: the starting task for the interaction (either maze of sorting), and the dominant hand of the user (left or right). It was not known before the experiments began whether one task type was inherently more challenging than the other, and what implications that might have on the experimental results. So the starting task was alternated between subjects, ensuring that balanced numbers of each occurred in each study condition pair. At most a 1-person difference was allowed. Similarly, it was not known whether the interaction would be affected by the dominant hand of the user. As an attempt to reduce the effects of this, the interface was flipped horizontally for left-handed participants (to prevent their raised hand from obscuring their view of the affordances and feedback at the side of the interface). Nonetheless, these participants were also assigned to study conditions such that balanced numbers occurred in each condition pair.

Figure 31 shows the differences in interaction performance between left-handed ($N = 4$) and right-handed ($N = 35$) participants and between participants who started with the maze task ($N = 21$) and those who started with the sorting task ($N = 18$). Each covariate value pair was evaluated for difference in task time and number of mistakes per task, across all six study conditions and with all six tasks considered together. The results demonstrate clear and strong separation in performance for both covariates: left-handed participants and participants who started with the maze task performed much better than their respective counterparts. In the case of dominant hand, this effect was present despite my efforts to make the interface equally accessible for left- and right-handed users.

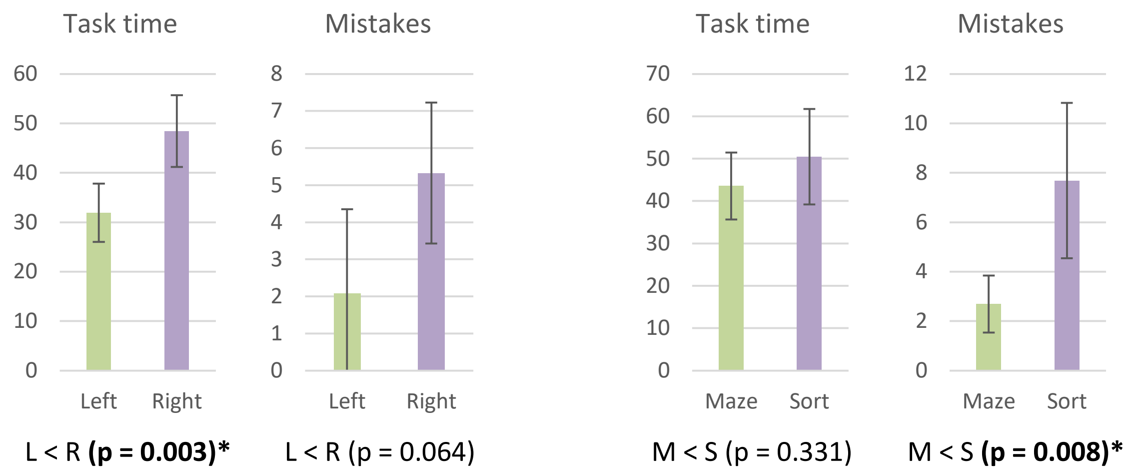


Figure 31. Performance results for the controlled covariates dominant hand (left measure pair) and starting task (right measure pair) across all study conditions, and for all tasks. Error bars denote 95% confidence interfals, and p-values are for 2-tail unpaired t-tests for differences in means with independent varainces. Statistical significance (α = .05 level) is denoted in bold and with an asterisk.

The difference is especially evident in both measures for dominant hand, and in number of mistakes for starting task. That is to say: participants who started with the maze task made much fewer mistakes overall (statistically significant difference at α = .05 level) and left-handed participants made much fewer mistakes and also completed the tasks faster (statistically significant difference at α = .05 level). However, as both of these covariates were counterbalanced for each sub-study prior to the start of each interaction (discussed at end of this section), these performance disparities did not influence the measured performance results in the different study conditions, nor the conclusions reached about the experiments' hypotheses.

5.6.2. Uncontrolled Covariates about Demographics and Experience

Prior each user's participation in the study, they answered demographic and experience-related questions that were later compiled into data for four binary-valued covariates and two continuous-valued covariate that could be analyzed. These covariates were not controlled for in the study design. In other words, different study conditions may have had different numbers of participants for different covariate categories. And as there was no a priori guarantee that the covariate categories were equally-performing in the performance data measured, a combination of covariate correlated with the dependent variables and unbalanced sub-study condition pairs for that covariate may have resulted in a confounding variable that throws the as-given study results into question. This analysis now seeks to identify any such covariates and determine whether they had a confounding effect on the experimental results

The four binary-valued covariates were evaluated in the same manner as the controlled covariates (that is, with two-condition hypothesis tests for differences in task time and number of mistakes). These covariates were: gender, prior experience with video games, prior experience with robots, and prior experience with other gesture interfaces. The values for these variables are male/female for gender, and yes/no for the remainder. The experience questions also asked participants about frequency of use (response choices were *never*, *once*, *a few times*, *regularly*, and *very often*). However, for the purposes of this evaluation, any experience was counted as prior experience. A more granular analysis of different frequencies of use did not find significant differences to the following results.

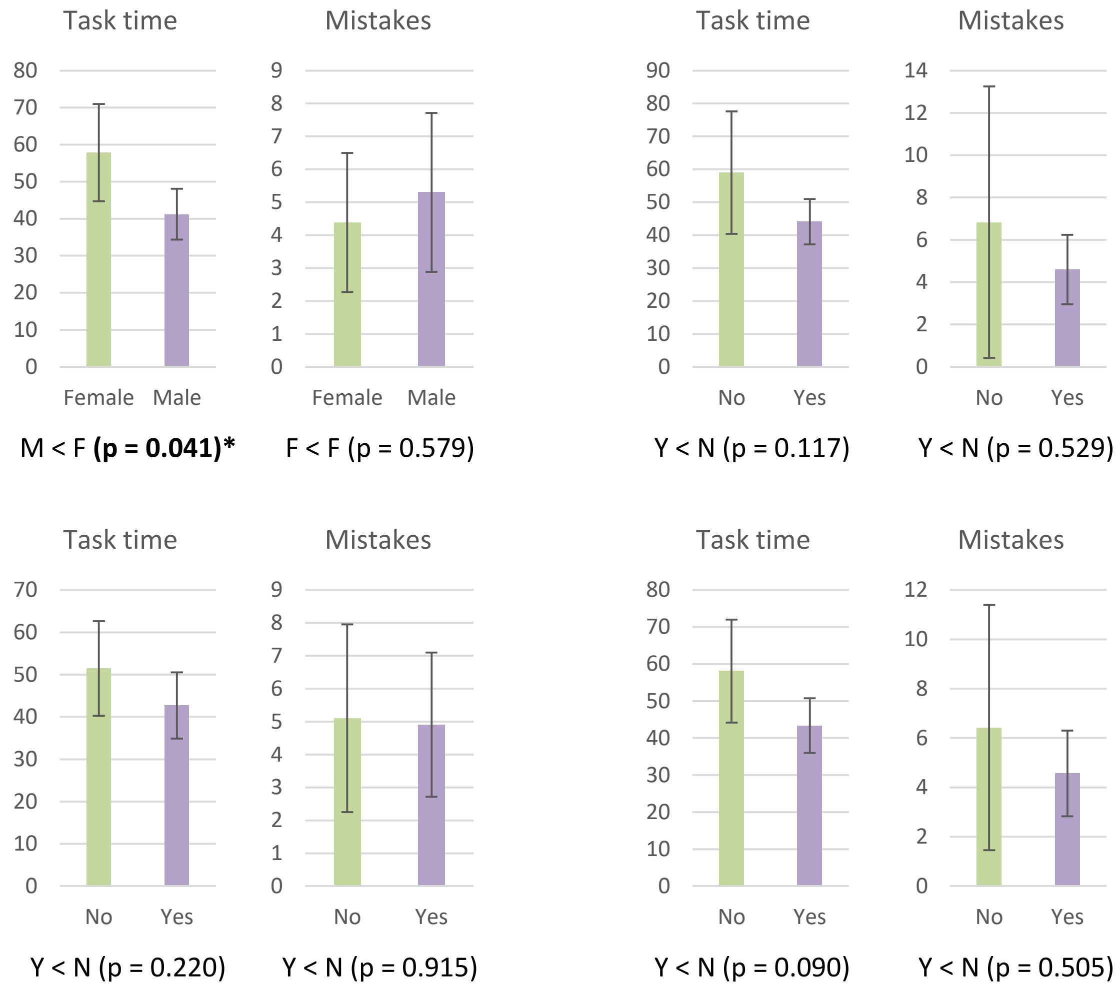


Figure 32. Performance results for uncontrolled demographic and experience-related covariates: gender (top-left), video games (top-right), robots (bottom-left), and gesture interfaces (bottom-right). Values are averages across all study conditions, and for all tasks. Error bars denote 95% confidence interfals, and p-values are for 2-tail unpaired t-tests for differences in means with independent varainces. Statistical significance (α = .05 level) is denoted in bold and with an asterisk.

Figure 32 shows the performance results for these four covariates, across all study conditions, and for all tasks. The first observation is that all four covariates impacted task time much more than the number of mistakes made. Second, a statistically significant difference (α = .05 level) was found between genders for average task time: men completed tasks much faster than women. Finally, the three experience covariates all demonstrated compelling evidence that prior experience with video games, robots, and other gesture interfaces correlate with better performance in this gesture interface. However, none of those covariates showed a statistically significant mean difference in either primary performance measure.

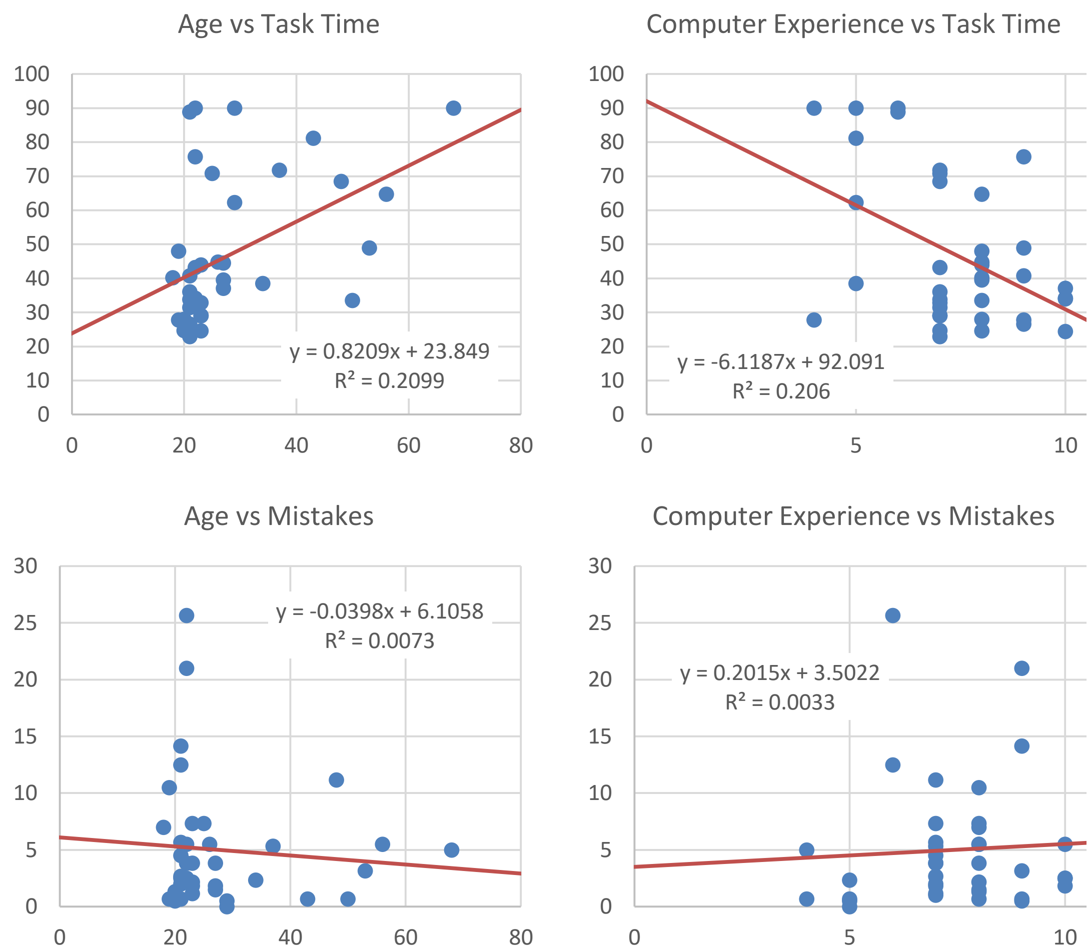


Figure 33. Linear regressions between the covariates age and self-reported level of computer experience (on a 10-point scale) and the performance measures task time and number of mistakes. Goodness-of-fit metric R^2 is provided for all refressions. All averages are per task, evaluated on all study conditions and for all tasks completed.

Two continuous-valued covariates were collected: age and self-rated computer experience (on a 10-point scale). These were evaluated using linear regression against the two primary task measures (time and number of mistakes). Figure 33 shows the regression results, including R^2 values for the lines of best fit. As can be seen, there is no appreciable correlation between the covariates and the average number of mistakes per task. And though there is a pronounced slope to the regression lines for the two covariates vs task time, the plots show that the lines of best fit are largely influenced by outliers, as the core data clusters around age 22 and computer experience level 7 do not show significant correlation. This verdict is reinforced by the low R^2 values for the two regressions.

Finally, covariates about participants mood prior to the interaction were collected via the same PANAS scale that was used to evaluate hypothesis 2, and personality factors along five axes were evaluated as well using questions from the International Personality Item Pool [20] and the Five-Factors scale by L. R. Goldberg [19]. Similar regressions were performed on all seven of these covariates for both task time and number of mistakes, but no strong correlations were found (the largest R^2 value in the set was 0.138). The results for these regressions are presented in Appendix H.

The conclusion then is that among these uncontrolled covariates, only gender demonstrates significant correlation with a dependent variable (task time), while experience with video games, robots and other gesture interfaces show potential correlation with task time, but not significantly so. Age and self-reported level of computer experience did not correlate strongly with either performance measure, nor did any of the mood or personality covariates collected.

5.6.3. Summary of Covariates and Effects on Experimental Results

		Task time		diff.	p	Mistakes		diff.	p
Dominant hand	Left	31.94	± 5.90	16.50	0.003	2.08	± 2.27	3.25	0.064
	Right	48.44	± 7.24			5.33	± 1.90		
Starting task	Maze	43.57	± 7.89	6.89	0.333	2.69	± 1.16	4.99	0.008
	Sort	50.46	± 11.24			7.69	± 3.14		
Gender	Female	57.82	± 13.13	16.61	0.041	4.38	± 2.11	0.92	0.579
	Male	41.21	± 6.85			5.30	± 2.41		
Played video games	No	59.03	± 18.59	14.97	0.177	6.83	± 6.41	2.24	0.529
	Yes	44.06	± 6.90			4.59	± 1.64		
Used robot	No	51.44	± 11.17	11.70	0.220	5.10	± 2.86	0.20	0.915
	Yes	42.73	± 7.82			4.90	± 2.19		
Used gesture interface	No	58.05	± 13.90	14.69	0.090	6.43	± 4.98	1.86	0.505
	Yes	43.36	± 7.35			4.57	± 1.74		

Table 7. Performance results for binary-valued covariates, as measured in two primary measures: task time and mistakes. Values are means for the given covariate and measure, across all study conditions and all tasks, with 95% confidence interfals. The lower (i.e. better) value in each covariate-measure pair is highlighted in green. Statistically significant differences (α = .05 level) are highlighted in orange. Mean difference which are almost statistically significant ($p < 0.1$) are highlighted in blue. All p-values are 2-tail.

Table 7 contains the numerical results for the six binary-valued covariates that were tested. These are the same values that were used to produce Figures Figure 31 and Figure 32. As can be seen, the two controlled covariates (dominant hand and starting task) and one uncontrolled covariate produces statistically significant performance differences (α = .05 level) in either task time or number of mistakes. The covariate corresponding to whether a participant had previously used a gesture interface produced a difference in task time that is almost statistically significant.

	Task time			Mistakes		
	Slope	Intercept	R^2	Slope	Intercept	R^2
Age	0.821	23.849	0.210	-0.040	6.106	0.007
Computer experience	-6.119	92.091	0.206	0.202	3.502	0.003

Table 8. Linear regression results for continuous-valued covariates versus two primary measures: task time and mistakes. Values are means for lines of best fit, with goodness-of-fit measured by R^2.

Table 8 contains the regression results for the two continuous-value covariates that were evaluated. These are the same values as for the regression plots in Figure 33. Neither covariate is strongly correlated with either performance measure.

		Sub-study A		Sub-study B		Sub-study C	
		A1	A2	B1	B2	C1	C2
Dominant hand	Left	1	1	0	0	1	1
	Right	6	6	7	6	5	5
Starting task	Maze	4	4	4	3	3	3
	Sort	3	3	3	3	3	3
Gender	Female	1	2	2	2	3	3
	Male	6	5	5	4	3	3
Play video games	No	1	0	1	1	3	1
	Yes	6	7	6	5	3	5
Used robot	No	2	1	5	4	4	2
	Yes	5	6	2	2	2	4
Used gesture interface	No	3	1	1	2	1	1
	Yes	4	6	6	4	5	5

Table 9. Distribution of study participants by study condition (independent measure) and six binary-valued covariates. Covariates which produced statistically significant effects (α = .05 level) in either of the two primary measures (dependent variables task time and mistakes) are highlighted in orange. Covariates which produced nearly statistically significant effects ($p < 0.1$) in the same are highlighted in blue. Sub-study condition pairs for which unbalanced numbers of participants fell into a covariate category are highlighted in red for that covariate category.

Finally, Table 9 presents the distribution of study participants by covariate and study condition, highlighting covariates that correlated strongly with the study results and sub-studies for which there was an unbalanced number of participants for a given covariate. An unbalanced covariate-sub-study pair is one where there is a greater than 1-person misbalance across the two sub-study conditions. This table can then be used to identify potential confounding effects due to the covariates.

The first observation is that sub-study C was unbalanced with respect to the number of participants who have played video games or have used a robot. However, as these covariates did not produce statistically significant ($\alpha = .05$ level) nor nearly statistically significant ($p < 0.1$) effects on the primary performance measures. So it cannot be said that these misbalances affected the experimental results for that sub-study.

Sub-studies A and B were both unbalanced with respect to the numbers of participants who have previously used a gesture interface. Specifically, Sub-study B was unbalanced only with respect to participants who *had* used a gesture interface, while sub-study A was unbalanced with respect to those who *had not* used one as well. This covariate produced a nearly statistically significant ($p < 0.1$) effect on task completion time, specifically favoring users who *had* previously used such an interface. This means that the results of these sub-studies may have been confounded by the effects of the uncontrolled covariate. Recalling that the evaluation of hypothesis 1 in sub-study A found condition A1 to outperform condition A2, a closer look at the unbalanced study conditions here finds that the misbalance is actually in condition A2's favor. This means the covariate worked *against* the sub-study's findings, rather than helped fulfil them.

So this covariate did not confound the results for sub-study A (evaluating hypothesis 1), and may actually have dampened the strength of the results instead. The covariate misbalance for sub-study B (evaluating hypothesis 3) similarly favored the opposite result of what was reported, so there again the covariate did not confound the results.

5.7. Summary of Experimental Results

Three sub-studies evaluated four hypotheses about the inclusion of affordances and feedback in the gesture system, and their results overturned the first of the hypothesis, were inconclusive for the second, and supported the third and fourth. Three of the hypothesis (H1, H3, and H4) concerned interaction performance, and were evaluated using four objective performance measures on a set of interactive tasks: task time, number of mistakes, number of gesture attempts, and time out-of-bounds. These performance measures are listed in order of importance, with the first two being considered the primary measures. A total of 21 subjective user responses in six categories were used to evaluate hypothesis H2, which concerned user satisfaction with the gesture system.

The first hypothesis – that including visual affordances and feedback in the gesture interface would result in improved interaction performance – was surprisingly overturned by the experimental results of sub-study A, as they demonstrated an opposite effect overall. In the first task, the study condition with all affordances and feedback (condition A2) *did* seem to produce faster task completion times (19.9% faster, $p = 0.178$) than the condition without them (condition A1). But in the remaining tasks, condition A1 was conclusively faster than A2 on average (26% faster, $p = 0.050$), and also seems to have produced fewer mistakes (39% fewer, $p = 0.187$), fewer gesture attempts (21.4% fewer, $p = 0.211$), and less time out-of-bounds (23% less, $p = 0.163$).

The second hypothesis – that including visual affordances and feedback in the gesture interface would result in higher user satisfaction – yielded inconclusive results in sub-study A. Of the 21 user responses gathered, 2 responses favored condition A1 (no affordances, only minimal feedback), 1 seems to have favored condition A2 (all affordances, all feedback), and 18 measures had average differences between the two conditions that were not statistically distinguishable at any standard degree of certainly ($p > 0.1$, including 10 measures with $p > 0.5$). Study participants who interacted with all affordances and all feedback (condition A2) perceived that they make fewer mistakes during the interaction (0.79 point difference on 5-point Likert scale, $p = 0.066$) than those in condition A1, but they also found the interface more distracting (1.48 point difference, $p = 0.028$) and more difficult to understand (1.57 point difference, $p = 0.048$).

The third hypothesis – that affordances indicating how to perform gestures are more important to a gesture interaction than those indicating what actions can be done – was confirmed overall by the experimental results of sub-study B. An interface including only *how-to* affordances (condition B2) yielded 18.4% faster task completion overall ($p = 0.145$) than an otherwise identical interface including only *can-do* affordances instead (condition B1). In the very first task the difference was not present, but thereafter condition B2's advantage in task time became more pronounced (26.4% faster in tasks 2 – 6, $p = 0.056$), and it also resulted in 63.2% fewer mistakes than condition B1 ($p = 0.096$).

The fourth and last hypothesis – that a feedback providing unsolicited system status is more important to a gesture interaction than feedback acknowledging user actions – was confirmed overall by the experimental results of sub-study C. Study condition C2, containing only feedback with *system status*, resulted in 21.8% faster task completion ($p = 0.089$), 52% fewer mistakes ($p = 0.111$), and 38.9% less time out-of-bounds ($p = 0.010$) than condition C1, which contained only *acknowledgement* feedback but was otherwise identical to condition C2. Again, the difference in task time and number of mistakes was not present in the first task, but in the remaining tasks 2 – 6, the time advantage became 27.9% ($p = 0.057$), and the advantage in number of mistakes was 59.3% ($p = 0.120$).

Six binary-valued and nine continuous-valued covariates were analyzed to determine whether any of them may have confounded the results of the experiments, and none were found to have done so. Two of the binary-valued covariates (dominant hand and starting task) were controlled for by counterbalancing during the experimental trials. Of the remaining four (gender, and prior experience with: video games, robots, and gesture interfaces), two covariates demonstrated a significant or potentially significant difference in task time: gender (28.7% difference, $p = 0.041$) and prior experience with gesture interfaces (25.3% difference, $p = 0.090$). However, the distribution of participants in these covariates was not misbalanced for any of the sub-studies in a way that may have been responsible for the experimental results observed. The continuous-valued covariates were not strongly correlated with the observed results either; the strongest correlations were for age ($R^2 = 0.210$) and self-reported level of computer experience ($R^2 = 0.206$).

6. DISCUSSION AND UPDATED INTERACTION FRAMEWORK

This chapter discusses the experimental results, with particular emphasis on the surprising overturning of hypothesis 1, and provides an updated version of the gesture interaction interface framework introduced in section 3.1. The discussed includes an exploration of possible explanations for the overturning of hypothesis 1, an evaluation of the success of the interface and experiments themselves, and insights gathered with respect to the framework and general gesture interfaces. The framework, though not directly evaluated by the experiments, is then updated in light of the insights gathered.

6.1. Discussion of Experimental Results

This section distills and interprets the experimental results to produce clear findings and recommendations for the design of gesture interfaces and integration of affordances and feedback in them. It also explores potential explanations for the surprising results for the first hypothesis, and discusses the successes and shortcomings of the gesture interface and experiments. To aid in discussion of the different study conditions and their experimental results, Table 2 in chapter 2 may be referenced to see which affordances and feedback are included in each study condition.

6.1.1. The Surprising Reversal of Hypothesis 1

The most surprising finding from these experiments was that including all the affordances and feedback in the gesture interface (Condition A2) resulted in worse interaction performance than omitting them (Condition A1), completely contradicting hypothesis 1. The all-affordance, all-feedback interface did actually appear to yield much faster task completion times in a user's very first interaction with the interface (15.5 second average difference, p = 0.178), but after this brief initial experience, the no-affordances, minimal-feedback interface gained a large and statistically significantly lead (11.6 second average difference, p = 0.05) in the remaining task attempts.

A hint to the cause for this surprising may instead be hidden within the performance *improvement* for each condition after the first task. Users in condition A2 started out well compared to condition A1, but thereafter they only reduced their task completion time by 17.5 seconds or 28%. Users in condition A1, on the other hand, reduced their task completion time by 44.6 seconds, or 57%. So the real result is that omitting these affordances and feedback somehow yielded a much larger improvement in the primary measure after the first task. A similar, but less pronounced effect was seen in the reduction of task mistakes: 8.6 fewer mistakes after the first task (76% reduction) for condition A1 vs 6.1 fewer mistakes after the first task (57% reduction) for condition A2

The question then is: why did users without the affordances and feedback improve so much more than those with them? And if all it takes to achieve high interaction performance is fumbling around at first, then are affordances and feedback even necessary? To answer these questions, the performance results for all six study conditions needs to be looked at, as well as the subjective responses collected to evaluate hypothesis 2. From this information, three potentially explanatory features of the interaction are evaluated.

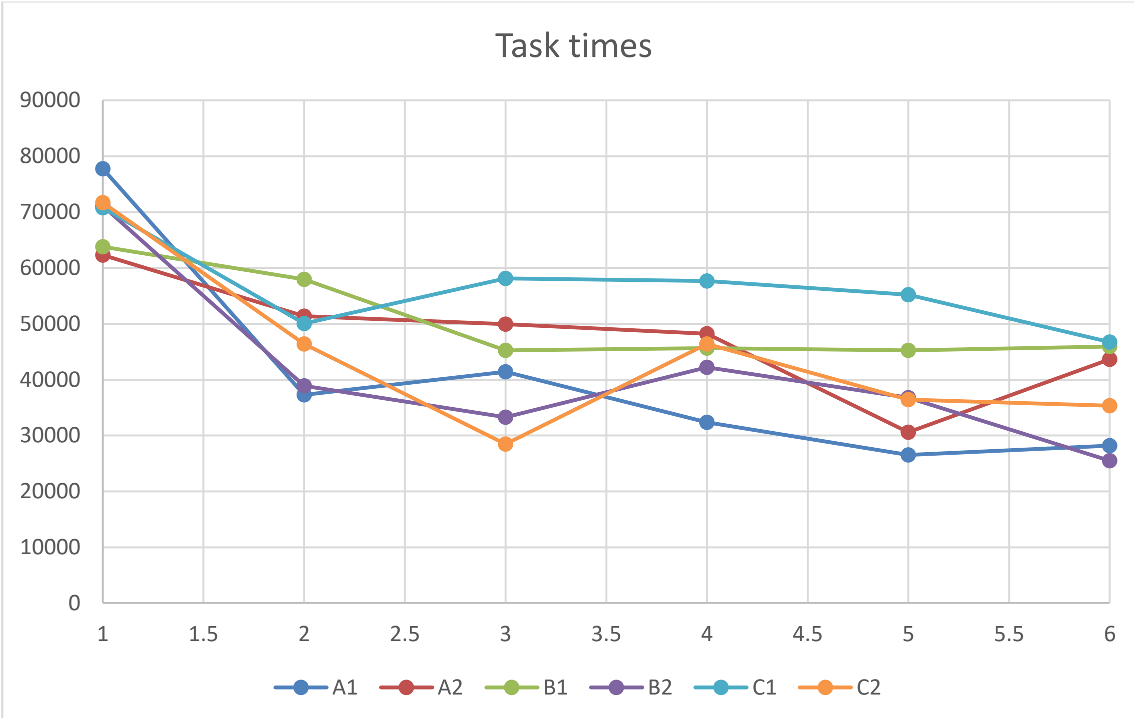


Figure 34. Comparison of average task times for all conditions.

First, comparing the performance results of all the study conditions (see Tables Table 3, Table 5, Table 6, and Figure 34 above) shows that Condition A2 did not perform particularly poorly, but rather Condition A1 performed exceptionally well, especially in task time. Looking again at improvement across tasks, it is found that A1 was actually the *slowest* of all conditions in the first task, yet the *fastest* in the remaining tasks and overall. And only condition B2 had a greater time improvement between tasks 1 and 2. Furthermore, condition A2 was actually the fastest in the first task.

Second, according to written and verbal comments provided by participants, the most challenging part of the interaction was figuring out how to perform the gestures which correspond to the select and drop actions in the two application interfaces. This difficulty was expressed by participants in all study conditions, and was especially problematic in the first task. In fact, several participants confessed that they made it past the first task thanks only to luck and extensive experimentation. Yet by the second task, participants in condition A1 seem to have already learned the gestures, as evidenced by the dramatic improvement in all performance measures (improvement of 52%, 51%, 80%, and 23% for the four measures, in order). Only condition B2 showed similar improvement in the two primary measures, yet that was almost certainly because it contains affordances showing the user how to perform the gestures in question.

Third, given the surprising flip of results after task 1, it may be supposed that that users who did not have the full affordances and feedback in their interface may have experimented more during the first task and thus learned to use the interface "the hard way," by trial and error. They might then have instilled more deeply the features they learned about the interface (e.g. how to perform the gestures, the fact that there are bounds to the interaction space, where those bounds are, etc.) and thus achieved better performance in later tasks. However, the results in the other three measures do not support this ad hoc hypothesis. There was no meaningful difference on the first task in number of mistakes or gesture attempts, nor in the ratio of time spent with hands outside the bounds of the interaction space.

So the surprising result of hypothesis 1 is not due to poor performance by condition A2, but rather exceptional performance by condition A1. The users in condition A1 somehow overcame the challenge of learning two specialized gestures without any gesture affordances. And those users did not experiment with the interface to a greater extent than users of the other study conditions. Finally, the covariates evaluated in the previous chapter did not show evidence that any had a confounding effect on the results for hypothesis 1.

Having exhausted directly measureable evidence that may help explain the unusually strong performance of condition A1, a speculative explanation is proposed. The lack of affordances and feedback in the interface may have encouraged users to focus on the application interface and the task at hand rather than try to make sense of the gesture interface. The lack of active application affordances may not be a significant loss, as the elements of the application interface have passive affordances as well (e.g. distinct shape, color, etc.). And the implicit application feedback remains too, along with the cursor feedback. The users of this interface might not have been aware of what gestures are supported and how to perform them, but they also did not have to incorporate those gestures into their conceptual model of the interface. Certainly they rated the interface as less confusing and distracting than users who interacted with gesture affordances and feedback.

Possibly the cognitive load to a true novice user of interpreting the joint gesture- and application-interface offset the benefits of the included affordances and feedback. If users interacting with Condition A1 only had to directly consider the application interface, they might in fact have been exploring the gesture mechanism in the first task and thus learned the appropriate gestures that correspond to application actions they wish to perform. Meanwhile, users in the other conditions may have similarly been gesturing and testing to flesh out their conceptual model of the joint interface. If this was the case, the increased gesture attempts, mistakes, and out-of-bounds ratios of the cognitively burdened users would have masked the exploration done by the users in Condition A1.

6.1.2. Successes and Shortcomings of the Interface and Experiments

The aim of this gesture interface was to support general-purpose interaction for truly novice users, using a virtual touchscreen paradigm, and incorporating affordances and feedback that can be evaluated experimentally. To that end, this work has been a success. However, there is certainly room to improve upon the approach used. Before discussing what might have been done differently in hindsight, some of the achievements and contributions are discussed.

To the author's knowledge, this work presents the first framework describing the role of affordances and feedback in a joint gesture and application interface. This framework compliments Norman's stages of interaction model, focusing on the system's perspective rather than the user's. The distinction made between the gesture interface and the application interface clarifies the relationship between the two, and highlights the parts of the interaction that require the may support of affordances and feedback.

The general-purpose interface uses the metaphor of a virtual touchscreen to support a somewhat more familiar interaction for novices, and to support the integration of generic touch-based applications. With the use of a 3D virtual interaction space and a selection plane, this paradigm also supports a discoverable mapping between gestures and application interface actions. This implementation also incorporates some mouse- and pen-like interaction as well to better take advantage of the physical gesturing modality while remaining close to the original metaphor.

Another small, but important contribution was the finding that novice users are not particularly upset by trying to use a gesture interface to actually achieve something (e.g. a game-like task) in a timed setting – though they may not be particularly happy about it, either. This may not sound like a high praise for the gesture interface, but it must be considering together with the fact that many of the artifacts people interact with on a daily basis cause frustration and confusion [46], especially when encountering something new. So for a gesture interface with no instructions to allow complete novices to perform as well as they did, and not make them upset is indeed a worthwhile achievement.

This work is also to the author's knowledge the first formal evaluation of purposefully designed affordances and feedback in a gesture interface. This evaluation considered different logical categories of affordances and feedback, and was done in a 40-person user study comprising only of true novice users. The size of this study is large by the standards of the typical gesture interface evaluation (as in the Related Work chapter), and the targeting of true novices is uncommon as well. Participants were given no advance instruction nor training, no practice time, and they were exposed to the full interface from the very start of the interaction. Both the application tasks and the physical interaction mechanisms were new to the participants, yet the vast majority were able to quickly learn to use the interface.

And finally, the results of the evaluation – though surprising for the first hypotheses – yielded clear findings and recommendations about the types of affordances and feedback that are more important to a gesture application. These findings are discussed in the following subsection.

As to the shortcomings of this work, the first is that the collected experimental data failed to concretely explain the negative result for hypothesis 1. The sub-studies and their conditions were setup under the assumption that hypotheses 1 and 2 would be easily and clearly supported, and focused much more on the evaluation of hypotheses 3 and 4. As such, additional measures which may have helped explain those results were not available.

According to participants, the gesture affordances indicating how to perform tap and withdraw gestures were unclear, and thus these gestures remained challenging to learn. This difficulty may have been compounded by the choice to use a static selection plane to detect the gestures. Although the selection plane – and the related choice to track only the centroid of the user's hand – simplified the gesture detection task, many (perhaps even most) of the study participants attempted several other tapping gestures before understanding how to use the one implemented.

Also, the effectiveness of the gesture feedback for detected tap and release gestures may have been hampered by the scale of the cursor response. Participants indicated that it was not clear when they had performed a gesture, even though the appropriate gesture feedback was enabled for their study condition. In designing this feedback this work focused on consistency within the application interface, but apparently at the cost of visibility. A larger or otherwise more visible form of feedback would have been more effective.

6.1.3. Lessons for Gesture Interfaces

One thing these experiments made clear was that different types of affordances and feedback do not have equal importance to a gesture interface. In particular, the experimental results confirmed the hypotheses that *system status* feedback and affordances indicating *how to do something* are more important in a gesture than *acknowledgement* feedback and affordances indicating *what can be done*, respectively. This means that if interface space or design time is at a premium, or if the inclusion of some particular affordance or feedback interferes with another, a clear choice can be made about which affordance or feedback to give precedence to. In fact, as Sub-study A shows, after the initial interaction, including all forms of affordances and feedback may actually hurt interaction performance – even to the point of being outperformed by an interface without any affordances and feedback.

A significant consequence of this finding is that it may be possible to integrate an existing application interface within a gesture interface without needing to design and add additional application affordances. The passive affordances inherent to the application may suffice, and in any case the gesture affordances indicating how to perform specific gestures may meet the affordance needs of the interaction. This means that it may be much easier to support many applications within a gesture interface without significant modification.

However, an important caveat to this preference is that for truly first-time users of a gesture interface, the relative importance of the types of affordances and feedback is much less clear, or even reversed at the very start of the interaction. Again, as demonstrated by the results of Sub-study A, at the very start of an interaction with a novice user, the best performance can be achieved by including both types of affordances and both types of feedback.

Another issue made apparent by this work is that the mapping between the gestures supported in the gesture interface and the actions supported in the application interface needs more than affordances to be fully realized. It is not enough that a user be informed what gestures are supported and how to perform them, nor that they be given the same information about supported actions within the application. They need to be confident that when they perform a gesture, it will map to the action they intended within the application. My choice to use a cursor and virtual touchscreen paradigm leverages constraints and familiarity to achieve the mapping, but the constraints themselves are virtual and are only visualized by status or acknowledgement feedback indicating when the user's hand has moved out of bounds or past the selection plane.

Hidden constraints such as these are problematic for two reasons: they break with the principle of interface visibility (e.g. a user cannot see when their hand is close to crossing the selection plane or moving out of bounds at the front or rear of the interaction space), and it places knowledge about the interface in the user's head rather than in the world (e.g. the user must recall rather than recognize where the bounds of the interaction are), which increases the cognitive load of using the interface. An alternative approach would be to leverage feedforward in the gesture interface to aid in its mapping to the application interface. For example, an affordance for a particular supported gesture can indicate what action it corresponds to in the application interface.

Finally, it is the author's suspicion that affordances and feedback for an intermediary interface (e.g. a gesture interface) may impose an unintentional cognitive load on novice users, no matter how well designed and implemented. The application interface should be the focus of the user's attention, so any gesture affordances or feedback would divert the user's attention, especially if unexpected. If the user's conceptual model of the interaction contained only the application interface at the start, introducing gesture affordances and feedback as a later time may require them to update or reevaluate their conceptual model, potentially introducing confusion and frustration.

6.2. Updated Framework for Gesture-based Interfaces

In light of the experimental results, the preliminary interaction framework introduced in section 3.1 is updated here with two important changes. The original framework gave equal emphasis to all affordances and feedback (in the gesture interface as well as the application interface), but the experimental results demonstrated that this may reduce the effectiveness of the interface. The original also does not consider learning effects on the interaction, nor does it distinguish between novice users and experienced ones. The experiments showed that even very brief experience can dramatically change the way users interpret and respond to affordances and feedback. And finally, the original framework did not provide guidance about how to help users form a consistent conceptual model of a gesture interaction.

The first change is to make clear within the framework the relative importance of the different types of affordances and feedback, so that appropriate effort may be devoted to their design and implementation. The experimental results showed that affordances showing a user how to perform an action are more important to an interface than affordances showing them what actions they can do; so the former type of affordance should be given precedence. The same is true for system status feedback versus action acknowledgement feedback. This change requires that both affordance types (*can-do* and *how-to*, for both the gesture affordances and the application affordances) be separated in the visualization of the interaction framework.

Second, and most important, a simplified, yet still accurate conceptual model of the gesture interaction needs to be suggested to the user. Interface users should certainly not have to concern themselves with affordances, acknowledgement, etc., but they do need to be aware of the gesture system between them and the application. Without the gesture system in their mental model, users may fail to understand, for example, that a gesture they perform may be successfully detected, yet not trigger the application action they intended. Or that they are attempting to select the correct item in an application interface, but are performing the selection gesture wrong.

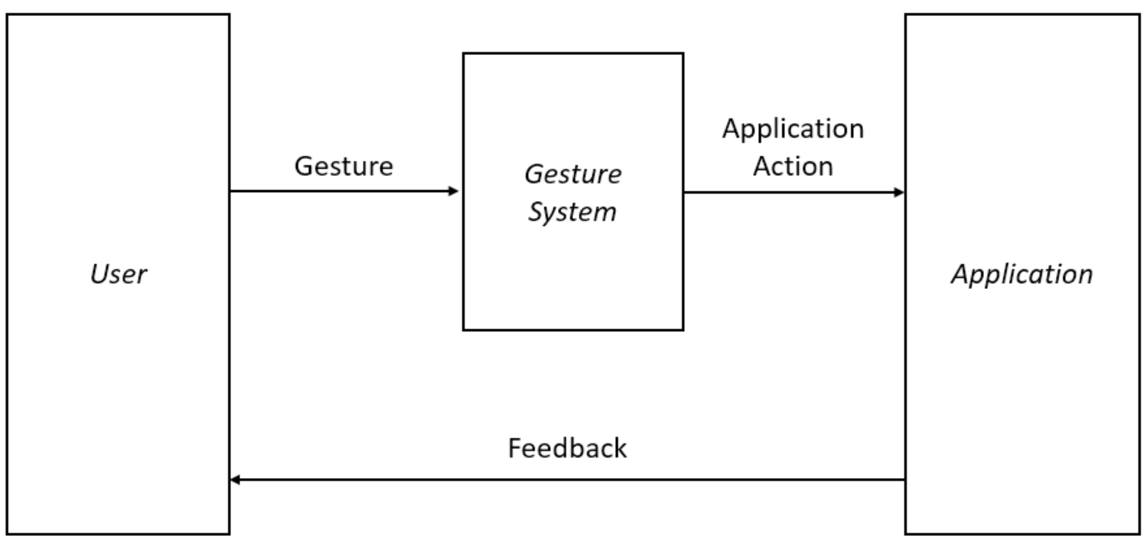


Figure 35. A simplified conceptual model of the gesture interaction, for the user. The key aspect of this admittedly simple model is that still provides a clear mapping between the gestures the user can perform and the actions the application supports.

The conceptual model that is suggested to the user should help them map their gesture attempts to the actions supported by the application interface (Figure 35). And it can be suggested most clearly to the user by synchronizing the gesture affordances with the application affordances, and the gesture feedback with the application feedback. Showing a user an action supported by an application interface together with the corresponding gesture supported by the gesture interface tells them that the action and gesture are closely related, but distinct. When they perform a supported gesture which triggers a supported action in the application, simultaneous acknowledgement feedback from the gesture interface and the application interface makes it clear to the user that the gesture triggered the action, and that the mapping between the two is supported by the gesture system.

This completed framework (Figure 36) is intended as a guide to design affordances and feedback for a gesture interaction in a prioritized manner, and also to design gesture interactions that suggest a meaningful conceptual model of the interaction to the user. The synchronization of gesture and application affordances, and or gesture and application feedback is denoted by the green and orange bars joining the gesture affordance and application affordance arrow, as well as the gesture feedback and application feedback arrows. This clear coupling indicates that the pair should be considered together and reminds the designed that their synchronization is an essential part of the interface. The relative importance of different categories of affordances and feedback is shown using bold versus italic font faces for the labels. Thus, if the interface designer chooses to create a particularly spare interface, they can clearly see which affordances and feedback are still worth implementing.

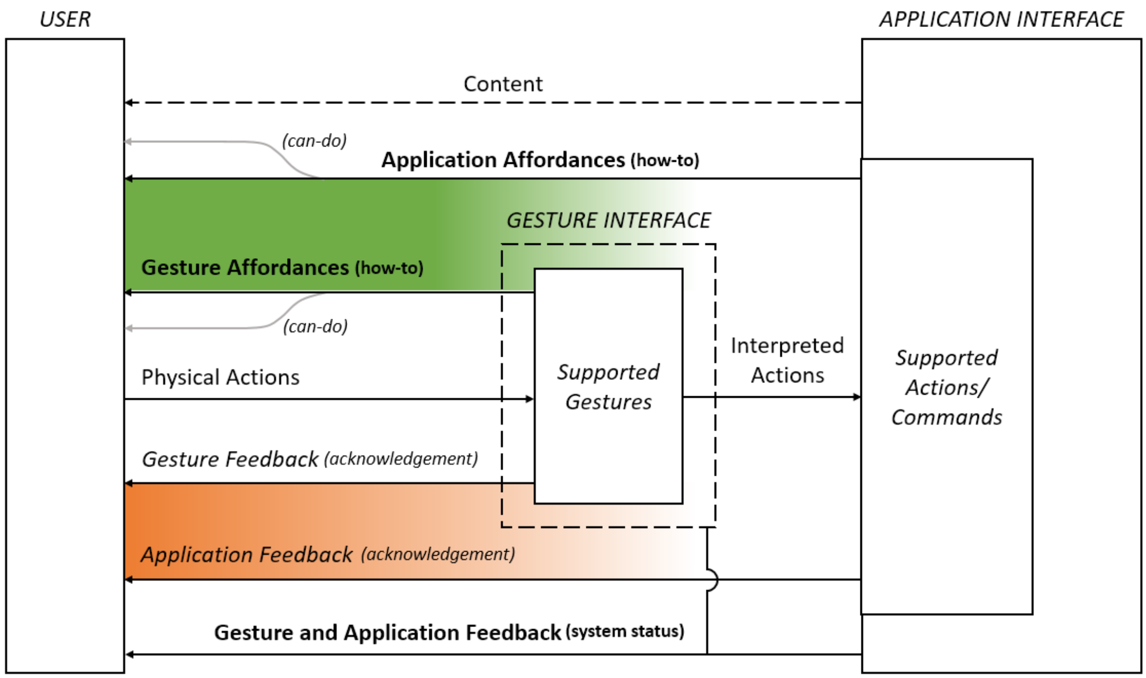


Figure 36. Updated gesture interaction framework. The four categories of affordances and feedback are now made clear, and the ones more important to the interaction (according to the experimental results) are presented in bold, whereas their less-significant counterparts are shown in italics. The green and orange bars denote the synchronization of gesture and application affordances, as well as gesture and application feedback.

7. SUMMARY AND CONCLUSIONS

This chapter concludes this work with a summary of the complete dissertation, a discussion of several new research questions introduced by this work and directions for future work to explore those questions, and a compilation of the conclusions and contributions made by this work.

7.1. Summary of Dissertation

This work was motivated by the need for intuitive interaction with assistive robots, and the challenges that arise when attempting to meet that need with a gesture interface. To start a gesture interaction, the user needs to first know that gesturing is supported, then *what* gestures are supported, *how* to perform them, and *what might result* from them. Then, to maintain the interaction, the user need to know whether they are being understood by the gesture interface, and they need to know when something has gone wrong. And all of this information must be relayed to a novice user during the interaction itself, as there is no opportunity for formal training or practice, and there is no pre-established warnings or feedback that they would know to expect. Furthermore, the interface must accommodate differences among different users with respect to their preferred gestures and assumptions about the interaction. Typical gesture systems, such as those discussed in section 2.1 do not support novice users well, as they tend to use complex gesture sets and require training or practice prior to their use.

The core approach of this work was to use affordances, feedback, and universal gestures to develop a gesture interaction supporting novice users. From the related work in section 2.2.2, this work identified two logical categories of affordances (those indicating *what a user can do* in an interface, and those indicating *how to do it*) and two logical categories of feedback (*acknowledgement of user actions*, and unprompted *system status* updates) that may be evaluated with respect to one another. It was found in the cognitive psychology literature that pointing is a universal gesture (section 2.3), but that the different manners of pointing are not themselves universal. And besides, a single gesture – universal or not – does not make for a very use interesting or useful interaction.

A universal pointing gesture was therefore supplemented with two additional gestures ("tap-to-select" and "withdraw-to-release") and the three were used together to support a virtual touchscreen paradigm for interaction (section 3.2.1). This approach benefits from the familiarity, proven ease-of-use, and generality of touch input; but the physical affordances and feedback inherent in a touch interaction are lost in translation when moving to an intangible in-air experience. So solve this problem, the bounds, direction, and scale of the virtual touch interaction was constrained to an oriented 3D interaction space, fixed with respect to the display and gesture sensor. A cursor on the display follows the user's hand – tracked only by its centroid for robustness to different pointing types – as it moves within the interaction space, in a one-to-one mapping with the xy-plane of the 3D space. And within this interaction space, a virtual selection plane serves to robustly detect the two complimentary gestures. When the user moves their hand forward past the selection plane, a "tap-to-select" gesture is detected, and when they then move their hand back behind the plane a "withdraw-to-release" gesture is detected.

To formalize the design process for the gesture interface created for this work, section 3.1 describes the development of a novel framework for the integration of affordances and feedback in a gesture interaction. This framework was designed to complement Don Norman's seven-stage model of an interaction, focusing on the system's perspective where Norman had focused on the user's. The framework distinguishes between the gesture interface and the application interface in a gesture-based interaction, and it clarifies the relationship between the two and the affordances and feedback that each need to support interaction with them. The affordances and feedback considered by the framework are in all four logical categories identified previously in the related work.

Using the interaction framework and the virtual touchscreen paradigm, a general-purpose gesture interaction system was designed and implemented which supports integration of arbitrary interactive applications. The intent is that any application that supports touch-, pen-, or mouse-based interaction should be able to integrate into the gesture system. The visual interface of the system (described in section 3.2.2) contains a generic application space surrounded by a sort of dashboard area where gesture affordances and feedback may be presented to the user. Section 3.3 explains in detail the six affordances (four in the *what* category and two in the *how* category) and nine types of feedback (seven *acknowledging user actions* and two providing unsolicited *system status* updates) that were designed and integrated into the visual interface, following the guidelines of the interaction framework.

A set of user studies was designed to evaluate two research questions using the gesture system: *How does including visual affordances and feedback impact the performance of a hand gesture interaction with a novice user?* and *What types of affordances and feedback are most important for such an interaction?* In section 4.2, four hypotheses were derived from these questions, the first two comparing an all-affordances, all-feedback interface to a "barebones" one, the third comparing *what can be done* affordances to *how to do something* affordances, and the fourth comparing *acknowledgement* feedback to *system status* feedback. Two applications, described in section 4.3, were developed for participants to use within the gesture system – a card sorting task and a simple maze task – each with a clear objective and a time limit to complete it. Section 4.6 describes the six study conditions that were created and grouped into three pairwise sub-studies. And evaluation was done using four objective task performance measures, and 21 subjective user responses in six categories. These evaluation measures are described in section 4.4, along with the covariates that were collected and analyzed as well. A total of 40 participants of diverse ages and backgrounds participated in the user studies.

The results of the experiments included two major surprises: section 5.2 shows how participants who interacted with a "full" interface (all-affordances and all-feedback) performed *worse* than those who used a "barebones" interface (no affordances and only minimal feedback), and section 5.3 shows that they were *not* more satisfied with the interaction, as had been predicted. However, the "full" interface did yield higher performance when participants first started the interaction (section 5.2.2), suggesting that the affordances and feedback were initially helpful, but thereafter users of the "barebones" interface improved much more quickly (section 5.2.3). As predicted by the third hypothesis, affordances indicating *how to do something* were shown in section 5.4 to be more important to interaction performance than those indicating *what can be done*. And similarly, as predicted by the fourth hypothesis, feedback providing unsolicited *system status* information was shown in section 5.5 to be more important to interaction performance than feedback *acknowledging user actions*.

Section 6.1.1 explored potential explanations for the surprising result of hypothesis 1, but failed to find one that could be directly supported by the experimental data. That section concluded with a speculative theory that the result may be due to the reduced cognitive load of not having to interpret the gesture interface. Sections 6.1.2 and 6.1.3 provided important insights distilled from the experimental results with respect to designing gesture interfaces: different types of affordances and feedback have inherently different degrees of importance; existing applications supporting touch or mouse input may be integrated in a gesture interface without the need to add specialized affordances and feedback to the application's interface; the helpfulness of affordances and feedback to a novice user may change dramatically after their initial interaction; and the mapping between supported gestures and supported application actions cannot be fulfilled by affordances alone – feedforward and constraints may be incorporated, but care must be taken to avoid hidden constraints.

Finally, in light of the experimental results, two updates are made to the gesture interaction framework in section 6.2. The first change makes clear in a graphical manner the relative importance of different types of affordances and feedback in the joint gesture-application interface the framework describes. In particular, and in accordance with the results summarized in sections 5.4.4 and 5.5.4, the framework gives precedent to affordances indicating how gestures are performed in the interface and to unsolicited feedback about the joint system's status. The second change proposes a simplified but accurate conceptual model of the interaction that is made discoverable to users via synchronization of "how-to" affordances and "acknowledgement" feedback in the gesture interface and the application interface.

7.2. New Questions and Future Work

This dissertation answered questions about the nature and effect of affordances and feedback within a gesture-based interaction, but it also produced several new questions along the way. For example: *How do individual affordances and feedback instances compare to one another*? The experiments compared two *categories* of affordances, each with several instances or forms of affordances (and similarly for feedback). However, it was beyond the scope of this work to compare individual forms to one another. A more important question is: *How do the experimental results generalize to gesture interactions with more complex gesture sets*? Although complex gesture sets were explicitly avoided here (in favor of intuitive pointing-based gestures), they may be beneficial in some interactions. In particular, would similar results be observed if the problematic selection plane mechanism for gesture detection were replaced by more granular finger gestures?

The unexpectedly high interaction performance by participants in Condition A1 (no affordances and only minimal feedback) presents several questions that were not able to be answered satisfactorily. The most important question, of course, is why did participants in this study condition perform so well and improve so quickly compared to other participants? A theory is proposed that lowered cognitive load and early exploration contributed to the performance, but the validity of this theory is itself is an important question to be answered.

Another question is how might other types of visual (e.g. flashing colors, more lifelike gesture templates, etc.) and nonvisual (tones, speech, haptics) feedback compare to the feedback implemented in this work? And what about physical affordances (e.g. physical frame or grid to identify the interaction area)? Or extending the gesture system also support speech, or touch?

An important finding from the experiments is that affordances and feedback have different effects at the start of an interaction versus later as the user becomes more familiar. An important follow-up questions is whether this effect would still have been seen if all users had the same interface for the first task (e.g. the “barebones” interface from condition A1), but then switched to different conditions in remaining the tasks? Similarly, it remains to be seen whether including affordances and feedback at the start of an interaction then removing them would yield improved interaction performance as well.

These questions are all worthy of further investigation, however the immediate focus of future work beyond this dissertation will be on extending the supported gestures and forms of gesturing, making the gesture detection mechanisms more sensitive to small (e.g. finger-based) gestures, and further exploring the integration of the gesture system with traditional mouse-, pen- and touch-based applications. Focus will remain on interaction with novice users, including limited-mobility users, such as the old and infirm. It is a goal of the future work extending from this dissertation that if a person can so much as move a finger, they should be able to have a meaningful gesture-based interaction. Finally, focus will also return to the computer vision component of a gesture interaction, to leverage some of the recent advancements in this area into a richer and much more robust interaction.

7.3. Conclusions

The intellectual merits of this dissertation are summarized in three novel contributions: 1) a general-purpose gesture interface built on a virtual touchscreen paradigm and designed to support novice users, 2) an in-depth evaluation of four categories of affordances and feedback in this interface, and 3) a set of findings and guidelines to inform the design of future gesture interfaces, centered on a novel framework for interaction design and the role of affordances and feedback in gesture interaction.

The experimental results in work produced three key finding and conclusions. First, though the inclusion of affordances and feedback in a gesture interface produces improved interaction performance for novices on an *initial* interaction, thereafter the omission of those explicit affordances and feedback yields surprisingly higher performance from novices. Second, affordances that tell a user *how* to perform gestures and other interface actions are more important to a gesture interaction than affordances that tell them *what* actions they can do. Third, feedback that provides unsolicited system status updates requiring the user action is more important to a gesture interaction than feedback acknowledging the user's actions.

Three additional conclusions were reached concerning the design of gesture interfaces for novices and the integration of affordances and feedback in them. First, interactive gesture systems consist of two distinct interfaces – the gesture interface and the application interface – whose affordance and feedback needs must be evaluated separately. Second, it may be possible to successfully integrate existing touch-, pen- and mouse-based application interfaces in a virtual touchscreen gesture interface like the one described in this work, potentially without the need for additional explicit affordances and feedback. However, well-designed and prioritized affordances and feedback may improve interaction performance, at least at first. Third, it should be a design priority to make clear the mapping between the gesture interface and the application interface it supports (in particular, between the gestures supported by the former and the actions supported by the latter), and to guide users to form an accurate conceptual model of the interaction. This may be accomplished by appropriate affordances and constraints, and in particular by synchronizing the relevant affordances and feedback present in the two interfaces.

This dissertation directly impacts the HCI and HRI research communities, and with the continued growth of gesture interaction with computers and robots, it may become an important motivation and reference for best design practices for new gesture systems. This work also has an important broader societal impact, as it may influence how future assistive robots are designed to interact with members of the general population –

particularly novice users – in various settings, including search and rescue, healthcare and elderly care. As described in the motivation for this work, these future users of all degrees of familiarity will need to be able to start a gesture-based interaction and then maintain it, and they will be guided to do so by well-designed interaction affordances and feedback.

REFERENCES

1. J.A. Adams. "Critical considerations for human-robot interface development". in *2002 AAAI Fall Symposium*. 2002.

2. J.K. Aggarwal and Q. Cai. "Human motion analysis: A review". in *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*. 1997.

3. O. Bau and W.E. Mackay. "OctoPocus: a dynamic guide for learning gesture-based command sets". in *Proceedings of the 21st annual ACM symposium on User interface software and technology*. 2008.

4. M. Beaudouin-Lafon. "Designing interaction, not interfaces". in *Proceedings of the working conference on Advanced visual interfaces*. 2004.

5. M. Becker, E. Kefalea, et al., "GripSee: A gesture-controlled robot for object perception and manipulation". *Autonomous Robots*, 1999. vol. 6, issue 2, p. 203-221.

6. A. Berenguer, J. Goncalves, et al., "Are Smartphones Ubiquitous?: An in-depth survey of smartphone adoption by seniors". *IEEE Consumer Electronics Magazine*, 2017. vol. 6, issue 1, p. 104-110.

7. R.P. Bonasso, E. Huber, and D. Kortenkamp. "Recognizing and interpreting gestures within the context of an intelligent robot control architecture". in *AAAI Fall Symposium on Embodied Language and Action*. 1995.

8. A. Bragdon, R. Zeleznik, et al. "GestureBar: improving the approachability of gesture-based interfaces". in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009.

9. G. Butterworth, "Pointing is the royal road to language for babies". *Pointing: Where language, culture, and cognition meet*, 2003. vol., issue, p. 9-33.

10. S. Calinon and A. Billard. "Stochastic gesture production and recognition model for a humanoid robot". in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*. 2004.

11. K. Cheng and M. Takatsuka. "Estimating virtual touchscreen for fingertip interaction with large displays". in *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*. 2006.

12. R. Cipolla and N.J. Hollinghurst, "Human-robot interface by pointing with uncalibrated stereo vision". *Image and Vision Computing*, 1996. vol. 14, issue 3, p. 171-178.

13. N. Diolaiti and C. Melchiorri. "Teleoperation of a mobile robot through haptic feedback". in *Haptic Virtual Environments and Their Applications, IEEE International Workshop 2002 HAVE*. 2002.

14. T. Djajadiningrat, K. Overbeeke, and S. Wensveen. "But how, Donald, tell us how?: on the creation of meaning in interaction design through feedforward and inherent feedback". in *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques*. 2002.

15. T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots". *Robotics and autonomous systems*, 2003. vol. 42, issue 3, p. 143-166.

16. D. Freeman, H. Benko, et al. "ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures". in *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. 2009.

17. W.W. Gaver. "Technology affordances". in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1991.

18. J.J. Gibson, *The ecological approach to visual perception: classic edition*. 2014: Psychology Press.

19. L.R. Goldberg, "An alternative" description of personality": the big-five factor structure". *Journal of personality and social psychology*, 1990. vol. 59, issue 6, p. 1216.

20. L.R. Goldberg, J.A. Johnson, et al., "The international personality item pool and the future of public-domain personality measures". *Journal of Research in personality*, 2006. vol. 40, issue 1, p. 84-96.

21. M.A. Goodrich and A.C. Schultz, "Human-robot interaction: a survey". *Foundations and trends in human-computer interaction*, 2007. vol. 1, issue 3, p. 203-275.

22. A. Green, H. Hüttenrauch, et al. "User centered design for intelligent service robots". in *Robot and Human Interactive Communication, 2000. RO-MAN 2000. Proceedings. 9th IEEE International Workshop on*. 2000.

23. K. Imagawa, S. Lu, and S. Igi. "Color-based hands tracking system for sign language recognition". in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*. 1998.

24. P. Jing and G. Ye-peng, "Human-computer interaction using pointing gesture based on an adaptive virtual touch screen". *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 2013. vol. 6, issue 4, p. 81-92.

25. M.J. Jones and J.M. Rehg, "Statistical color models with application to skin detection". *International Journal of Computer Vision*, 2002. vol. 46, issue 1, p. 81-96.

26. S. Kita, *Pointing: Where language, culture, and cognition meet*. 2003: Psychology Press.

27. R. Kjeldsen and J. Kender. "Finding skin in color images". in *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*. 1996.

28. M. Kolsch and M. Turk. "Robust hand detection". in *Automatic Face and Gesture Recognition*. 2004.

29. D. Kortenkamp, E. Huber, and R.P. Bonasso. "Recognizing and interpreting gestures on a mobile robot". in *Proceedings of the National Conference on Artificial Intelligence*. 1996.

30. S. Kratz and R. Ballagas. "Unravelling seams: improving mobile gesture recognition with visual feedback techniques". in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009.

31. S. Kristoffersen and F. Ljungberg. "Designing interaction styles for a mobile use context". in *International Symposium on Handheld and Ubiquitous Computing*. 1999.

32. E. Lin, A. Cassidy, et al. "Hand tracking using spatial gesture modeling and visual feedback for a virtual DJ system". in *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*. 2002.

33. G. Lintern, "An affordance-based perspective on human-machine interface design". *Ecological Psychology*, 2000. vol. 12, issue 1, p. 65-69.

34. K.E. MacLean. "Designing with haptic feedback". in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*. 2000.

35. J.R. Maier and G.M. Fadel, "Affordance based design: a relational theory for design". *Research in Engineering Design*, 2009. vol. 20, issue 1, p. 13-27.

36. A. Malima, E. Ozgur, and M. Çetin. "A fast algorithm for vision-based hand gesture recognition for robot control". in *Signal Processing and Communications Applications, 2006 IEEE 14th*. 2006.

37. N. Masataka, "From index-finger extension to index-finger pointing: Ontogenesis of pointing in preverbal infants". *Pointing: Where language, culture, and cognition meet*, 2003. vol., issue, p. 69-84.

38. C. Matuszek, L. Bo, et al. "Learning from Unscripted Deictic Gesture and Language for Human-Robot Interactions". in *AAAI*. 2014.

39. J. McGrenere and W. Ho. "Affordances: Clarifying and evolving a concept". in *Graphics interface*. 2000.

40. D. McNeill, *Language and gesture*. Vol. 2. 2000: Cambridge University Press.

41. S. Mitra and T. Acharya, "Gesture recognition: A survey". *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 2007. vol. 37, issue 3, p. 311-324.

42. J. Moeller and A. Kerne. "ZeroTouch: an optical multi-touch and free-air interaction architecture". in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2012.

43. C. Nass, J. Steuer, and E.R. Tauber. "Computers are social actors". in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1994.

44. D.A. Norman, "Cognitive engineering". *User centered system design: New perspectives on human-computer interaction*, 1986. vol. 3161, issue.

45. D.A. Norman, "Affordance, conventions, and design". *interactions*, 1999. vol. 6, issue 3, p. 38-43.

46. D.A. Norman, *The design of everyday things: Revised and expanded edition*. 2013: Basic books.

47. A.M. Okamura, "Methods for haptic feedback in teleoperated robot-assisted surgery". *Industrial Robot: An International Journal*, 2004. vol. 31, issue 6, p. 499-508.

48. E.-J. Ong and R. Bowden. "A boosted classifier tree for hand shape detection". in *Automatic Face and Gesture Recognition*. 2004.

49. V. Pavlovic, R. Sharma, and T.S. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1997. vol. 19, issue 7, p. 677-695.

50. K. Qian, J. Niu, and H. Yang, "Developing a Gesture Based Remote Human-Robot Interaction System Using Kinect". *International Journal of Smart Home*, 2013. vol. 7, issue 4.

51. D.G.S. Rao, D.D. Bhattacharya, et al., "Dual Sensor Based Gesture Robot Control Using Minimal Hardware System". *International Journal of Scientific and Research Publications*, 2013. vol. 3, issue 5.

52. B. Reeves and C. Nass, *The Media Equation: How people treat computers, television, and new media like real people and places*. 1996: CSLI Publications and Cambridge university press.

53. O. Rogalla, M. Ehrenmann, et al. "Using gesture and speech control for commanding a robot assistant". in *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*. 2002.

54. A. Sauppé and B. Mutlu. "Robot deictics: How gesture and context shape referential communication". in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*. 2014.

55. B. Shneiderman, *Designing the user interface: strategies for effective human-computer interaction*. 2010: Pearson Education India.

56. S.L. Smith and J.N. Mosier, *Guidelines for designing user interface software*. 1986: Mitre Corporation Bedford, MA.

57. C.X. Soto, Murphy, Robin R. "Boosted Cascades with Depth Images for Hand Detection". in *Robotics: Science and Systems Conference (RSS)*. 2016. Ann Arbor, Michigan, USA.

58. S. Strachan, P. Eslambolchilar, et al. "GpsTunes: controlling navigation via audio feedback". in *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*. 2005.

59. J. Suarez and R.R. Murphy. "Hand gesture recognition with depth images: A review". in *RO-MAN, 2012 IEEE*. 2012.

60. N.T. Thinh, N.T.V. Tuyen, and D.T. Son, "Gait of Quadruped Robot and Interaction Based on Gesture Recognition". *Journal of Automation and Control Engineering Vol*, 2016. vol. 4, issue 1.

61. M. Tosas and B. Li. "Virtual touch screen for mixed reality". in *International Workshop on Computer Vision in Human-Computer Interaction*. 2004.

62. J. Triesch and C. Von Der Malsburg. "A gesture interface for human-robot-interaction". in *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. 1998.

63. D. Uebersax, J. Gall, et al. "Real-time sign language letter and word recognition from depth data". in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. 2011.

64. M. Van den Bergh, D. Carton, et al. "Real-time 3D hand gesture interaction with a robot for understanding directions from humans". in *RO-MAN, 2011 IEEE*. 2011.

65. M. Van den Bergh and L. Van Gool. "Combining RGB and ToF cameras for real-time 3D hand gesture interaction". in *Applications of Computer Vision (WACV), 2011 IEEE Workshop on*. 2011.

66. J. Vermeulen, K. Luyten, et al. "Crossing the bridge over Norman's Gulf of Execution: revealing feedforward's true identity". in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2013.

67. V. Vezhnevets, V. Sazonov, and A. Andreeva. "A survey on pixel-based skin color detection techniques". in *Proc. Graphicon*. 2003.

68. P. Viola and M. Jones. "Rapid object detection using a boosted cascade of simple features". in *Computer Vision and Pattern Recognition (CVPR)*. 2001.

69. D. Vyas, C.M. Chisalita, and G.C. Van Der Veer. "Affordance in interaction". in *Proceedings of the 13th Eurpoean conference on Cognitive ergonomics: trust and control in complex socio-technical systems*. 2006.

70. L.S. Vygotsky, "The genesis of higher mental functions". *The concept of activity in Soviet psychology*, 1981. vol., issue, p. 144-188.

71. J.P. Wachs, M. Kölsch, et al., "Vision-based hand-gesture applications". *Communications of the ACM*, 2011. vol. 54, issue 2, p. 60-71.

72. S. Waldherr, R. Romero, and S. Thrun, "A gesture based interface for human-robot interaction". *Autonomous Robots*, 2000. vol. 9, issue 2, p. 151-173.

73. J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: virtual touch screen in the air using RF signals". *ACM SIGCOMM Computer Communication Review*, 2015. vol. 44, issue 4, p. 235-246.

74. D. Watson, L.A. Clark, and A. Tellegen, "Development and validation of brief measures of positive and negative affect: the PANAS scales". *Journal of personality and social psychology*, 1988. vol. 54, issue 6, p. 1063.

75. A. Webb, A. Kerne, et al. "Choreographic buttons: promoting social interaction through human movement and clear affordances". in *Proceedings of the 14th ACM international conference on Multimedia*. 2006.

76. R. Wen, W.-L. Tay, et al., "Hand gesture guided robot-assisted surgery based on a direct augmented reality interface". *Computer methods and programs in biomedicine*, 2014. vol. 116, issue 2, p. 68-80.

77. S.A. Wensveen, J.P. Djajadiningrat, and C. Overbeeke. "Interaction frogger: a design framework to couple action and function through feedback and feedforward". in *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*. 2004.

78. A.D. Wilson. "TouchLight: an imaging touch screen and display for gesture-based interaction". in *Proceedings of the 6th international conference on Multimodal interfaces*. 2004.

79. Y. Wu and T.S. Huang, "Vision-based gesture recognition: A review", in *Gesture-based communication in human-computer interaction*. 1999, Springer. p. 103-115.

80. D. Xu, X. Wu, et al., "Online dynamic gesture recognition for human robot interaction". *Journal of Intelligent & Robotic Systems*, 2015. vol. 77, issue 3-4, p. 583-596.

APPENDIX A

GESTURE INTERACTION WITH ASSISTIVE ROBOTS

Assistive robots are selected as a motivation for this work because of their frequent interaction with novice users, and the particular challenges they pose for such interaction (as well as those that generalize to other applications of gesture interfaces). These robots are embodied agents designed to provide physical, mental, and social support to people such as a medical patients or trapped victims [15, 21]. They are social in nature, and their design places high emphasis on user interaction [1, 22]. Their interfaces therefore must favor ease of use, with preference generally given to touch, speech, and gesture input. These modalities are sometimes said to support “natural user interaction” [53, 64], meaning that their use comes naturally to most people, or is otherwise intuitive or easy to learn.

Importantly, no natural user interface is appropriate for all situations. Gesturing supported by hand-held or wearable controllers or sensors may enable precise input, but such an interface is only feasible if the user can be provided with the controller or fitted with the sensors ahead of time – which is not generally the case for assistive robots. Touch-screens are highly familiar due to the explosive growth of smartphones, tablets, and similar devices over the last decade, but they require that a user be able to reach the screen, which is restrictive for some settings (e.g. healthcare, search and rescue). Natural language processing and visual gesture recognition are both contact-free and require no advance setup, but both rely on imperfect recognition that is highly dependent on environmental conditions (e.g. noise, lighting, occlusions). Both are also subject to regional and cultural constraints (e.g. speech has different languages and dialects, and most hand gestures have distinct meanings to different cultures [40]).

However, gesture input has the important advantage that some simple gestures are general enough to be understood consistently by large populations, and pointing in particular is used and understood universally [26, 40]. These gestures – such as pointing and selecting – can thus serve as a low-level interaction language. It is easy to imagine that directing a robot to stop or to move to some position can be done gesturally if the robot recognizes even a small library of common gestures, and that such commands – though simple – are useful in many interactions with an assistive robot.

APPENDIX B

VISUAL HAND DETECTION METHOD

The first critical component of gesture interaction system designed for operation outside of a laboratory setting is a method to robustly detect and track users' hands, which is no small problem. Because hands are highly articulated (with at least 26 degrees of freedom [49]) and thus highly varied in appearance, detecting them reliably in an image is very challenging. To support gesture interactions in the types of conditions that may be encountered by an assistive robot (e.g. poor lighting, occluded bodies, complex backgrounds, multiple foreground objects), a more robust appearance-based hand tracking method is needed.

Limitations of Heuristic Hand Detection Methods

Hand detection in the gesture recognition literature is usually done via heuristic approaches that rely on some preconditions or interaction restrictions to greatly simplify the hand tracking problem. Three common methods are used for visual hand detection: distance thresholds, skin-color segmentation, and motion detection. These methods can work well in a controlled setting, but they quickly fail in unplanned settings and also restrict a natural gesture interaction.

Skin color segmentation relies on the highly uniform hue (independent of illuminance) of skin colors among different people and ethnic groups [25, 67], but does not discriminate between hand and non-hand objects. Hands have to be distinguished from other skin-colored areas using clues such as motion [23] or geometric constraints [27], and hand detection is made harder or fails altogether in colored or very low lighting, if there are unexpected skin-colored objects in the scene, or if the user's hands are not skin-colored (e.g. if they are wearing gloves, or are covered in dust).

Locating hands by motion detection generally requires a stationary camera and a static background [49]. The user must typically wave their hand for such a method to work, and a dedicated tracking method such as mean shift is required since the user is typically not expected to keep their hand in constant motion [71]. Furthermore this method requires re-initialization whenever the tracking method loses the user's hands.

And depth thresholds – which apply a cropping plane at some distance in a depth image, and have been popularized by the availability of low-cost depth cameras such as the Microsoft Kinect – only work if the user's hand is the closest object in the scene. This method is an artifact of the best-case-scenario rationale that a user will always be facing the sensor and keep their hands in front of them [59]. And though this method produces temptingly good results under the assumed conditions [63], and IR depth cameras such as the Kinect have the advantage that they can work in very low light [65], the fact remains that depth thresholds fail for hand detection if the user's hands are not the closest object in the scene, which is often the case for uncontrolled interactions.

A Boosted Cascade Method for Hand Detection

In contrast to the commonly-used heuristic methods described above, the appearance-based hand detection method used in this gesture system applies no constraints on the user or the background, but instead has to contend with the high problem dimensionality introduced by the variability of hand poses. The method is based on boosted cascade object detection, developed by Viola and Jones [68]. The boosted cascade approach scans subwindows of an image at different scales and uses a cascade of boosted binary classifiers to discriminate subwindows containing the object from those that don't. The Viola-Jones method runs in real-time due to the cascaded nature of the classifiers (which reject most negative subwindows early), and the very efficiently computable Haar-like features used. Sample hand detection results (outdoors) are shown in Figure 37.

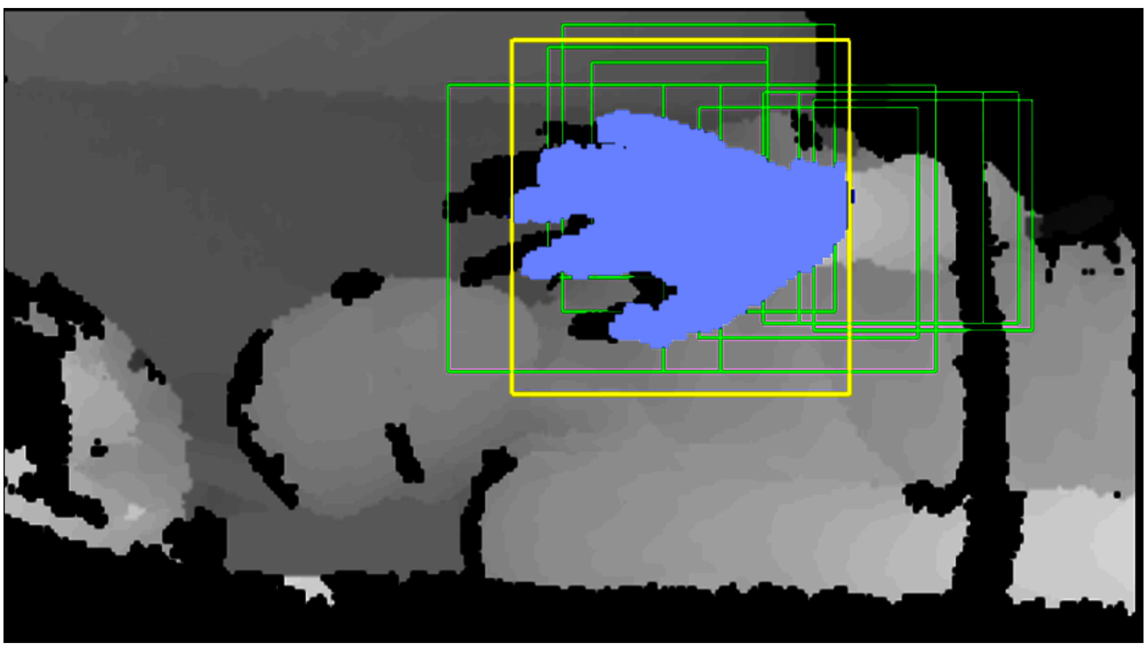


Figure 37. Sample results of the boosted cascade hand detector used in this work, from an outdoors test in a rubble pile at the TEEX Disaster City® training facility.

Using boosted cascade for hand detection is not a new idea, but it has thus far been done only with standard color cameras. Ong and Bowden [48] used a tree of detectors for different hand poses, with a general hand detector at the root. They produced the tree by using a K-mediod clustering algorithm to group a database of hand images into different poses. Kolsch and Turk [28] also used separate detectors for different hand poses, but they used a frequency spectrum analysis to select hand poses that are likely to be separable from a background.

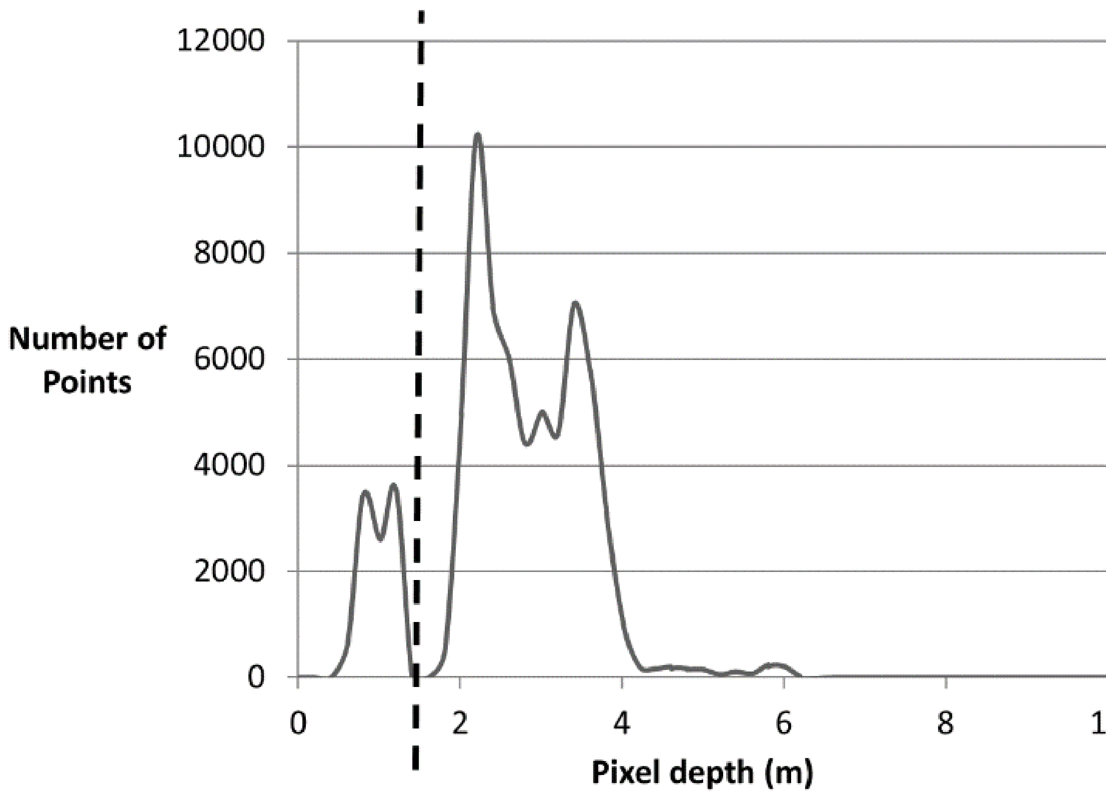


Figure 38. A smoothed depth histogram (20 bins) showing the discontinuity between foreground and background in a natural image.

The implementation created for this work uses a modified form of boosted cascades with a depth camera (originally an ASUS Xtion Pro Live, now an Intel/Creative Senz3D) instead of a standard color camera, enabling high detection performance and the ability to function in darkness. Depth cameras greatly simplify the background subtraction problem because they provide direct range information. Using a depth histogram, the foreground can be segmented by identifying the rear-most depth discontinuity in an image (Figure 38). And because the intrinsic camera parameters are known, scale information can be derived for objects in an image, which enables size-based filtering. I use such filtering to reduce false positive hand detections. A real-time (30 fps) version of my hand detector achieved a 94.7% detection rate (with 0.25 false detections per sample image) on an initial dataset collected from 5 volunteers in six indoors and outdoors conditions [57]. False positives are further reduced by detection averaging, and tracked hand paths are temporally smoothing to even out jitter.

However, it should be noted that the detection results on the initial dataset may not extend well to real-time performance, in which detection rates remain generally high, but false positives increase to as many as 10 (and occasionally more) per video frame. This is sufficiently good results for basic pointing interactions, but further refinement and testing of the detector performance is needed. The sensor data collected during the experiments will be used to for further testing and improvement of the detector as well as the development and testing of new hand tracking methods. As this requires manual labeling and checking of depth images to produce ground truth results, so was not be possible to perform this evaluation during the experiments. For this reason, evaluation of the modified boosted cascade method is beyond the scope of this work.

APPENDIX C

USER-STUDY RECRUITMENT EMAIL

Subject line: Talk with your hands! Be a participant in a hand gesture interaction study.

Howdy!

You're invited to be a participant in an exciting research study that evaluates new gesture recognition technology!

The study will take place in the Texas A&M Artificial Intelligence Robotics Lab. Participants will use hand gestures to interact with a computer and solve puzzles. Through measurements of gesture interaction performance, and responses from participants, we will be able to evaluate exciting features of a new gesture recognition system. The study will take about half an hour of your time, and is open to anyone over 18 years of age.

Sign up today! Email gesture.study@tamu.edu. DOOR PRIZES ARE AVAILABLE!

Research disclosures: Participation in this study is voluntary, and participants' responses will remain confidential, as will all study data. No video, audio nor other identifying information about participants will be collected. This research study is approved by the Institutional Review Board of Texas A&M (reference number IRB2015-0637D).

APPENDIX D

USER-STUDY RECRUITMENT FLYER

Talk with your hands!

Participants needed for an exciting study that evaluates new gesture recognition technology!

DOOR PRIZES ARE AVAILABLE!

The study will take place in the Texas A&M Artificial Intelligence Robotics Lab. Participants will use hand gestures to interact with a computer and solve puzzles. Through measurements of gesture interaction performance, and responses from participants, we will be able to evaluate exciting features of a new gesture recognition system. The study will take about 30 minutes of your time, and is open to anyone over 18 years of age.

Participation in this study is voluntary, and all study data and responses will remain confidential.

Sign up today! Email gesture.study@tamu.edu.

APPENDIX E

USER STUDY CONSENT FORM

Welcome, and thank you for your interest in this gesture interaction study being conducted by the Center for Robot-Assisted Search and Rescue at Texas A&M University. This form contains information about the research study and is provided to help you make an informed decision about whether you wish to participate. If you have any questions that are not answered here, please let the researcher know.

Introduction

This research study is about hand gesture interaction with a computer that uses a gesture recognition system. Gesture recognition is fast becoming a popular means of interacting in a natural way with computers and robots, much like speech recognition and touch input devices have in recent years. This study focuses on investigating the performance and quality of gesture-based interaction. The goal is to determine the effectiveness and ease of use of different components of a gesture recognition system.

If you choose to take part in this study, you will be asked to sign this consent form. If you choose not to participate, you may leave with no penalty to you.

What will I be asked to do?

If you choose to participate in this study, you will be asked to interact with a computer using hand gestures, and answer several written questions before and after your interaction. The questions are mostly multiple-choice responses, with a few free-response questions. The study will take approximately 20 – 30 minutes of your time, of which the actual gesture interaction time will take about 10 minutes.

For the gesture interaction, you will be asked to sit at a computer and complete two simple puzzle tasks by performing several gestures with your hand. The experience will be similar to playing a gesture-based video game such as those that use the Microsoft Kinect for Xbox.

Are there any risks involved in this study?

There is no greater risk involved in this study than in typical daily activities. If you can comfortably sit at a computer and gesture with your hands for up to 10 minutes, you should experience no unusual discomfort.

Are there benefits to me if I participate in this study?

There is no direct benefit to you from taking part in this study. However, your participation will contribute to a better understanding of gesture interaction principles and will aid in the development of future gesture interfaces for computers and robots. These future interfaces may allow assistive robots to better treat and support healthcare and elderly care patients, disaster victims, and members of the general population.

Do I have to participate? Can I change my mind?

Your participation is voluntary. You may choose not to participate, and you may withdraw at any time without any sort of penalty, and without having to provide any reason.

Will I be compensated?

You will not be paid for your participation in this study. However, you will be entered to win one of two $25 gift cards. Winners of these gift cards will be selected by drawing at the completion of the study. You do not need to complete the study to be eligible for this drawing. Additionally, participants with top performances (i.e. those who complete the gesture puzzle tasks in fast times and with few errors) will be eligible for $5 bonus gift cards.

Will I be recorded?

Neither video nor audio recordings will be collected during the study. Your hands will be tracked using computer algorithms and a gesture sensor. The sensor data will be recorded, which contains depth data only, no video nor audio.

Will my information be kept private?

Your participation in this study will be anonymous, and all records and data collected will be kept private. This means that the responses you provide and the data collected during the study will not be connected in any way to your name, likeness, or any other information that can be linked to you. All such data and responses will be stored securely and identified only by an arbitrary identification code. This consent form will be filed securely and separately. It is not linkable to your study data and responses.

Your research data will be kept confidential to the extent permitted or required by law. The records of this research project may be inspected by authorized research personnel, representatives of regulatory agencies such as the Office of Human Research Protections (OHRP) and entities such as the Texas A&M University Human Subjects Protection Program to ensure that information is collected properly.

Whom can I contact for questions about this research?

If you have questions regarding this study, you may contact Carlos Xavier Soto by phone (210-338-0610) or email (carlosxsoto@tamu.edu).

Whom can I contact about my rights as a research participant?

For questions about your rights as a research participant, to provide input regarding research, or if you have questions, complaints, or concerns about the research, you may contact the Texas A&M University Human Research Protection Program office by phone at 1-979-458-4067, toll free at 1-855-795-8636, or by email at irb@tamu.edu.

Hand Gesture Interaction Study | Consent Form

Statement of Consent

1. **I confirm that I have read and understand the information in the provided form, and that I have had opportunity to ask questions.**
2. **I understand that I can ask more questions at any time.**
3. **I agree and consent to participate in this research study, and understand that I may withdraw at any time without penalty.**

Signature: ____ **Date:** ______________________

Printed Name: ____

Signature of Person Obtaining Consent: ____________________________ **Date:** ___________

Printed Name of Person Obtaining Consent: ____

APPENDIX F

USER STUDY PRE-INTERACTION QUESTIONNAIRE

Age: __________

Gender: Male Female Other

Race/ethnicity:

White Black Asian Hispanic Native American

Pacific Islander Other: ___________________________________

Culture you most identify with (*note this may be different from your race or ethnicity*):

- ☐ English-speaking North American, European or Oceanian (Australia/New Zealand)
- ☐ Non-English speaking Western European
- ☐ Eastern European
- ☐ Southeast Asian
- ☐ Indian
- ☐ Middle Eastern
- ☐ African
- ☐ Latin American
- ☐ Other: ____

Highest education level:

- ☐ Some High School
- ☐ Completed High School
- ☐ Some College
- ☐ Completed College
- ☐ Graduate School

Major Field of Study: ___________________________________

Current or most recent occupation: ____

What is your level of computer experience?

(Beginner) (Expert)

1 2 3 4 5 6 7 8 9 10

Have you ever used or interacted with a robot or drone? Yes No

If yes, how often?

Once A few times Regularly Very Often

When was the most recent time?

Past Day Past week Past Month Past Year

Longer Ago

What types of robots?

☐ Consumer utility robots (like a Roomba or pool cleaner)

☐ Consumer toy robots (like Sony Aibo)

☐ Consumer UAVs or drones (like Parrot drones or DJI Phantom)

☐ Industrial robots (like those used for vehicle manufacturing)

☐ Guide or telepresence robots (like museum guides, or office teleconferencing robots)

☐ Self-made robots

☐ Other: ____

Do you play video games? Yes No

If yes, how often do you play?

Once A few times Regularly Very Often

What types of games do you play?

☐ Console games (Xbox, PlayStation, etc.)

☐ Handheld games (Gameboy, PSP, etc.)

☐ PC games

☐ Games for mobile devices (smartphones, tablets)

☐ Other: ____

Have you ever owned a pet? Yes No

If yes, what kind?

☐ Cat

☐ Dog

☐ Other: ____

Have you ever used a gesture interface to interact with a computer, game or robot (*for example, have you played video games that allow you to play by gesturing with your hands or body*)**?**

Yes No

If yes, how often?

Once A few times Regularly Very Often

When was the most recent time?

Past Day Past week Past Month Past Year

Longer Ago

What types of gesture interfaces have you used?

☐ Microsoft Kinect (for Xbox 360 or Xbox One)

☐ Nintendo Wii remote

☐ Sony PlayStation Move remote or PlayStation Camera

☐ Leap Motion Controller

☐ Custom or self-made gesture interfaces

☐ Other: ____

Below are a number of words that describe different feelings and emotions. Read each item and indicate to what extent you've felt that way **TODAY**. Use the following scale to record your answers.

1	2	3	4	5
very slightly or not at all	a little	moderately	quite a bit	extremely

_____ interested

_____ distressed

_____ excited

_____ upset

_____ strong

_____ guilty

_____ scared

_____ hostile

_____ enthusiastic

_____ proud

_____ irritable

_____ alert

_____ ashamed

_____ inspired

_____ nervous

_____ determined

_____ attentive

_____ jittery

_____ active

_____ afraid

On the following pages, there are phrases describing people's behaviors. Please use the rating scale below to describe how accurately each statement describes you. Note the following instructions:

- Describe yourself as you generally are now, not as you wish to be in the future.
- Describe yourself as you honestly see yourself, in relation to other people you know of the same gender as you, and roughly your same age.
- So that you can describe yourself in an honest manner, your responses will be kept in absolute confidence, and will not be connected to any personally identifiable information about you.

Please read each statement carefully, and then circle the appropriate number on the following scale.

1	2	3	4	5
Very Inaccurate	Moderately Inaccurate	Neither Inaccurate nor Accurate	Moderately Accurate	Very Accurate

I am relaxed most of the time.

1 2 3 4 5

I often forget to put things back in their proper place.

1 2 3 4 5

I feel little concern for others.

1 2 3 4 5

I have a rich vocabulary.

1 2 3 4 5

I am the life of the party.

1 2 3 4 5

I get stressed out easily.

1 2 3 4 5

I have a vivid imagination.

1 2 3 4 5

I am not interested in other people's problems.

1 2 3 4 5

I make a mess of things.

1 2 3 4 5

I feel comfortable around people.

1 2 3 4 5

I worry about things.

1 2 3 4 5

I have excellent ideas.

1 2 3 4 5

I am easily disturbed.

1 2 3 4 5

I insult people.

1 2 3 4 5

I start conversations.

1 2 3 4 5

I leave my belongings around.

1 2 3 4 5

I am quick to understand things.

1 2 3 4 5

I seldom feel blue.

1 2 3 4 5

I am not really interested in others.

1 2 3 4 5

I talk to a lot of different people at parties.

1 2 3 4 5

I am exacting in my work.

	1	2	3	4	5

I use difficult words.

	1	2	3	4	5

I shirk my duties.

	1	2	3	4	5

I get upset easily.

	1	2	3	4	5

I don't mind being the center of attention.

	1	2	3	4	5

I make people feel at ease.

	1	2	3	4	5

I spend time reflecting on things.

	1	2	3	4	5

I follow a schedule.

	1	2	3	4	5

I change my mood a lot.

	1	2	3	4	5

I don't talk a lot.

	1	2	3	4	5

I feel others' emotions.

	1	2	3	4	5

I am full of ideas.

	1	2	3	4	5

I like order.

	1	2	3	4	5

I have frequent mood swings.

1 2 3 4 5

I keep in the background.

1 2 3 4 5

I take time out for others.

1 2 3 4 5

I have difficulty understanding abstract ideas.

1 2 3 4 5

I get chores done right away.

1 2 3 4 5

I get irritated easily.

1 2 3 4 5

I have little to say.

1 2 3 4 5

I have a soft heart.

1 2 3 4 5

I am not interested in abstract ideas.

1 2 3 4 5

I pay attention to details.

1 2 3 4 5

I sympathize with others' feelings.

1 2 3 4 5

I don't like to draw attention to myself.

1 2 3 4 5

I often feel blue.

1 2 3 4 5

I do not have a good imagination.

1	2	3	4	5

I am always prepared.

1	2	3	4	5

I am interested in people.

1	2	3	4	5

I am quiet around strangers.

1	2	3	4	5

APPENDIX G

USER STUDY POST-INTERACTION QUESTIONNAIRE

Below are a number of words that describe different feelings and emotions. Read each item and indicate to what extent you felt that way **DURING THE INTERACTION**. Use the following scale to record your answers.

1	2	3	4	5
very slightly or not at all	a little	moderately	quite a bit	extremely

_____ interested	_____ irritable
_____ distressed	_____ alert
_____ excited	_____ ashamed
_____ upset	_____ inspired
_____ strong	_____ nervous
_____ guilty	_____ determined
_____ scared	_____ attentive
_____ hostile	_____ jittery
_____ enthusiastic	_____ active
_____ proud	_____ afraid

Below are a number of words that describe different feelings and emotions. Read each item and indicate to what extent you feel that way **AT THIS MOMENT**. Use the following scale to record your answers.

1	2	3	4	5
very slightly or not at all	a little	moderately	quite a bit	extremely

_____ interested	_____ irritable
_____ distressed	_____ alert
_____ excited	_____ ashamed
_____ upset	_____ inspired
_____ strong	_____ nervous
_____ guilty	_____ determined
_____ scared	_____ attentive
_____ hostile	_____ jittery
_____ enthusiastic	_____ active
_____ proud	_____ afraid

How physically comfortable did you feel gesturing with your hands during the interaction?

1	2	3	4	5
very uncomfortable				very comfortable

How physically comfortable do you think you would you feel using hand gestures to interact for a longer period of time (1 hour)?

1	2	3	4	5
very uncomfortable				very comfortable

Do you feel that you would be able to use hand gestures to interact successfully with a *computer interface*?

1	2	3	4	5
definitely no				definitely yes

Do you feel that you would be able to use hand gestures to interact successfully with a *robot*?

1	2	3	4	5
definitely no				definitely yes

How much did you enjoy using hand gestures in this interaction?

1	2	3	4	5
not at all				very much

How often do you feel that *the gesture recognition system* made mistakes during this interaction?

1	2	3	4	5
never				constantly

How often do you feel *you* made mistakes during this interaction?

1	2	3	4	5
never				constantly

If you had to interact with a computer or robot but could not speak to it or touch it, do you think you could successfully interact with it by using hand gestures?

1	2	3	4	5
definitely no				definitely yes

How easy or difficult was it for you to interact with this system using hand gestures?

1	2	3	4	5
very difficult				very easy

How easy or difficult was the sorting task to complete using hand gestures?

1	2	3	4	5
very difficult				very easy

How easy or difficult was the maze navigation task to complete using hand gestures?

1	2	3	4	5
very difficult				very easy

Do you have any comments about using this hand gesture system or the interaction tasks?

Do you have any other comments about the experiment?

Did you notice the visual feedback provided by the system while you performed hand gestures?

Yes No

If you answered no to the previous question, you may skip the rest of the questions.

How easy or difficult was the visual feedback to understand?

1	2	3	4	5
very difficult				very easy

Was the feedback confusing or distracting during either interaction task?

1	2	3	4	5
very confusing or distracting				not confusing nor distracting at all

Did the feedback help you understand what the gesture system was "seeing" and "thinking"?

1	2	3	4	5
did not help me understand at all				helped me understand very much

Did the feedback help you to notice or identify errors during the interaction?

1	2	3	4	5
did not help me notice any errors				helped me notice every error

Was the feedback helpful overall?

1	2	3	4	5
not helpful at all				very helpful

Was the feedback helpful when completing the interaction tasks?

1	2	3	4	5
not helpful at all				very helpful

Would this type of feedback help you to use a new, unfamiliar gesture system?

1	2	3	4	5
not helpful at all				very helpful

Would you change anything about the feedback the system provided?

Do you have any other comments about feedback for gesture systems?

APPENDIX H

EVALUATION OF MOOD AND PERSONALITY COVARIATES

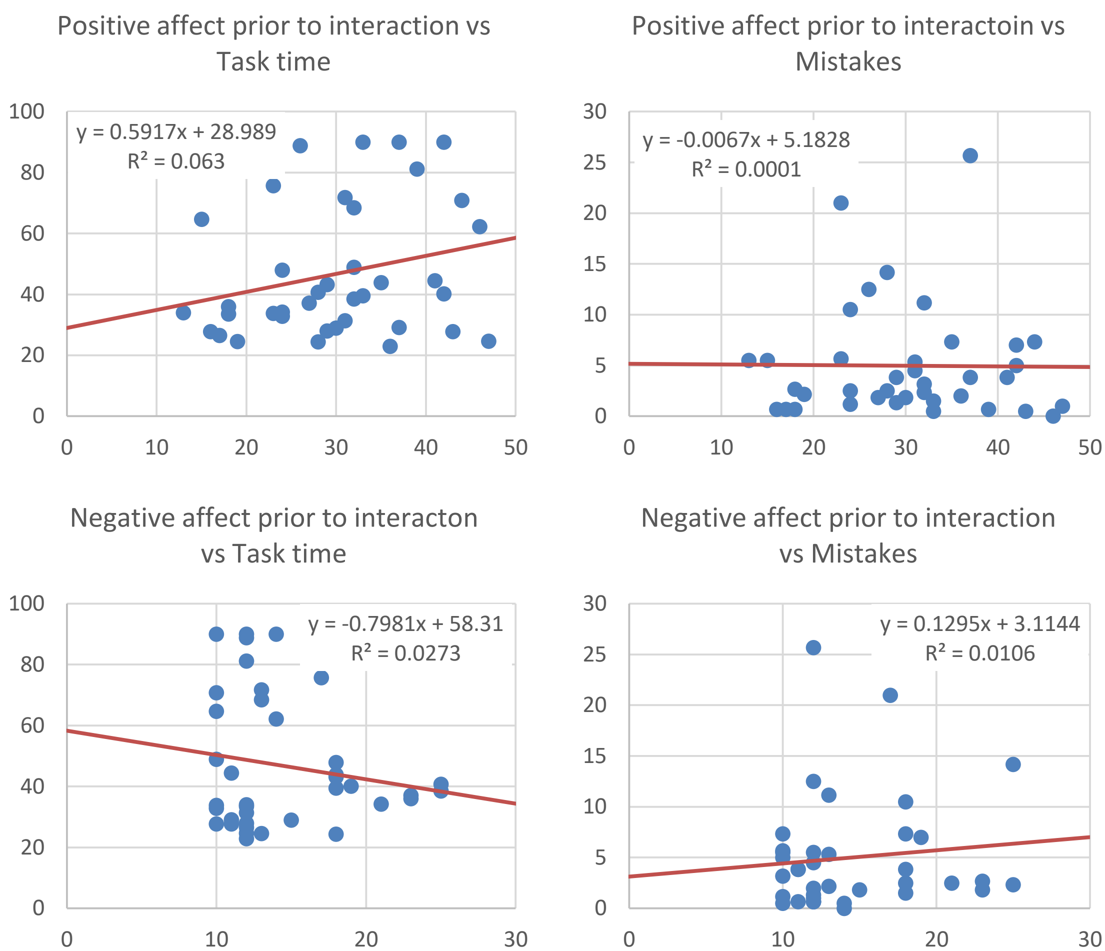


Figure 39. Linear regressions between mood covariates and the primary performance measures: task time and number of mistakes. Goodness-of-fit metric R^2 is provided for all refressions. All averages are per task, evaluated on all study conditions and for all tasks completed.

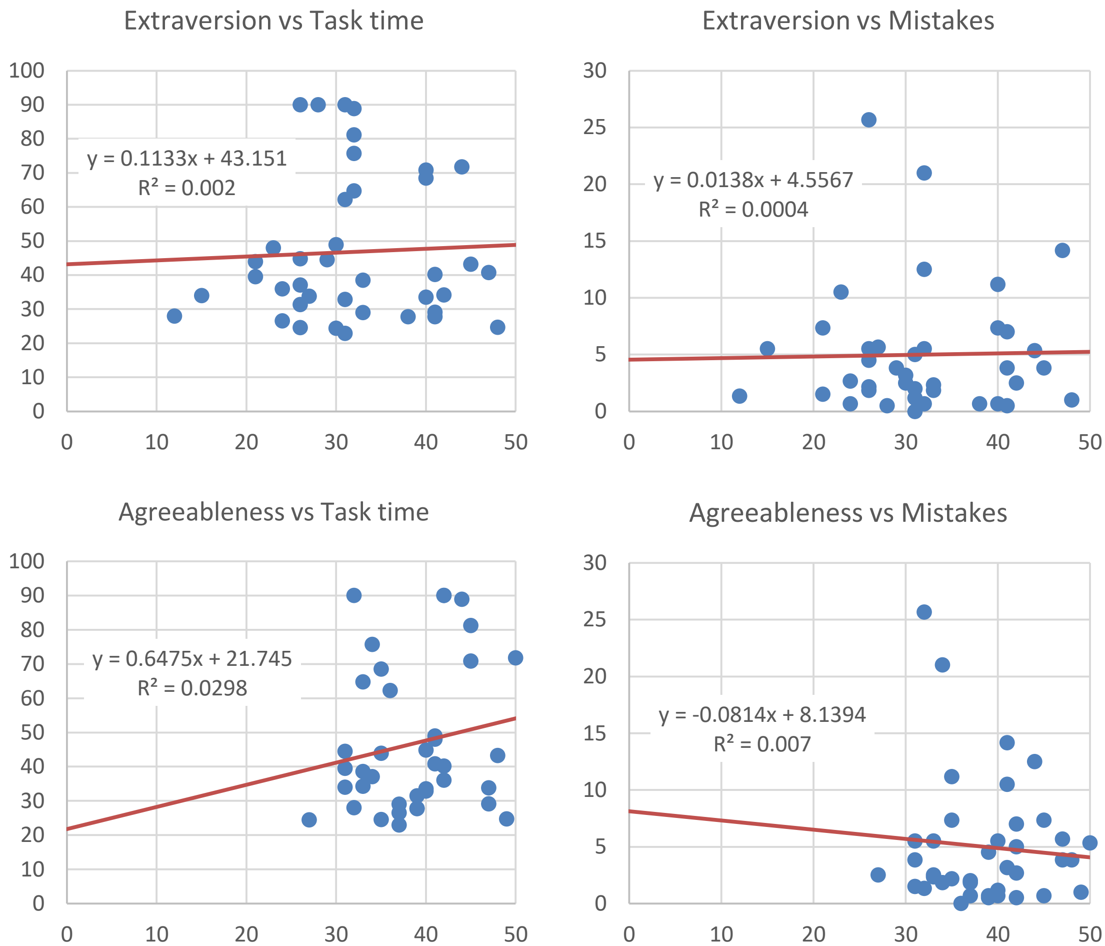


Figure 40. Linear regressions between personality covariates and the primary performance measures: task time and number of mistakes. Goodness-of-fit metric R^2 is provided for all refressions. All averages are per task, evaluated on all study conditions and for all tasks completed.

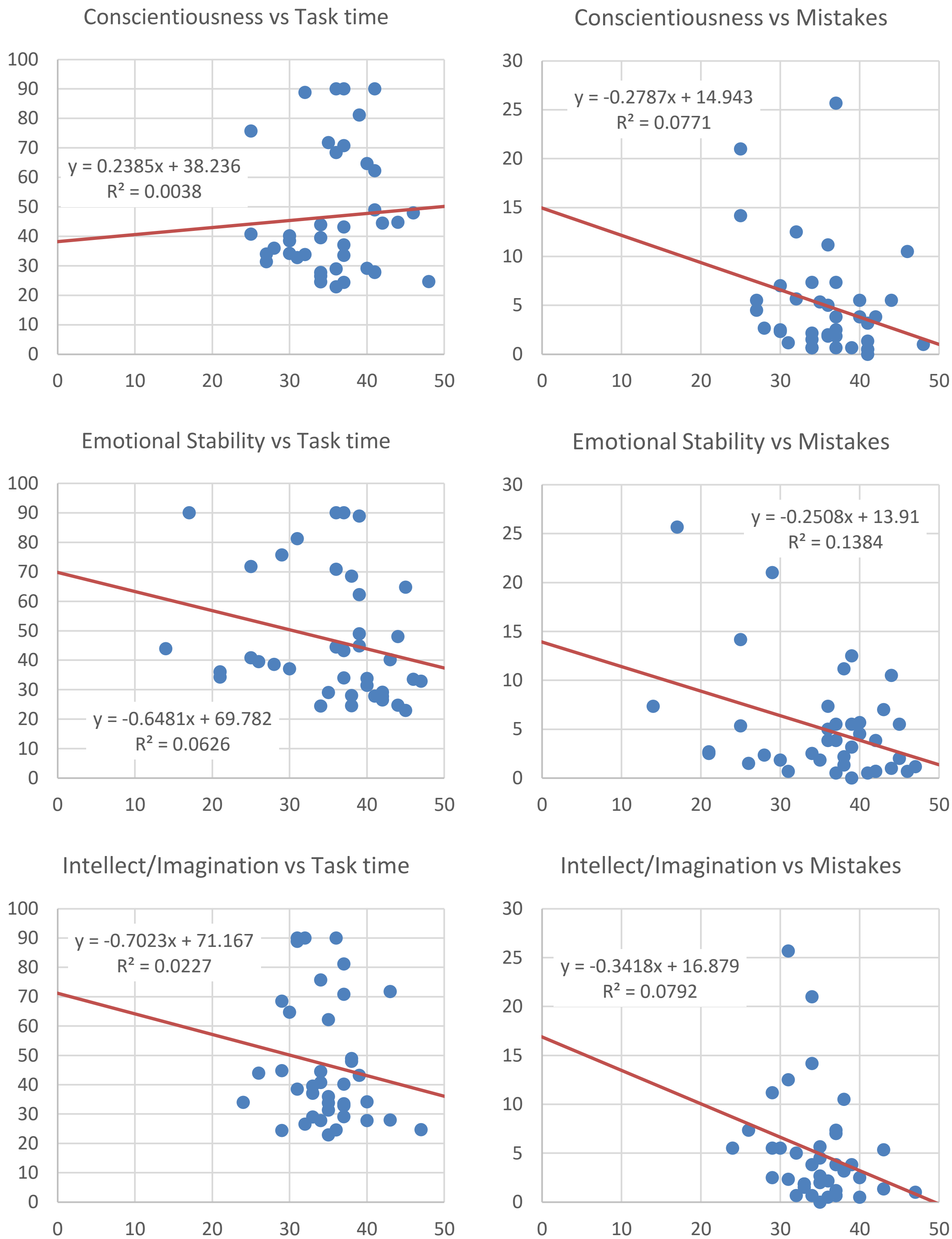


Figure 40, continued.