GESTURE BASED SOCIAL MEDIA AND NEWS INTERFACE

By

Steven T. King Jr.

A DISSERTATION

Submitted to

The University of Liverpool

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

ABSTRACT

GESTURE BASED SOCIAL MEDIA AND NEWS INTERFACE

By

Steven T. King Jr.

The use of non-touch gesture-based user experiences is becoming more prevalent in gaming as well as scientific and medial fields. Microsoft's Kinect motion and depth sensor was the fastest selling consumer electronic device in history and there is more than 10 million worldwide. With such availability in consumer homes at a relatively in-expensive price it is conceivable that using gestures for consuming news and interacting with social networks could be widely adopted.

This research seeks to answer if gesture-based interfaces using motion tracking sensor technology can be used to consume news and engage with social and current events content? In order to answer this question, a software system and new user experiences were designed, developed and user tested on a typical at-home Kinect hardware installation that enables users to interact with a browser and thus news and social information using gestures.

Although making gesture interfaces viewable on a standard browser has proved difficult, this project designed, developed and implemented a solution that allowed for use of custom developed interfaces as well as common news and social websites. The project provides a brief analysis of each available gesture integration solution as well as detailed explanation into the implementation of the chosen solution. Details of the hardware implementation, comprehensive layouts of new user experiences as well as the technical analysis and implementation of DepthJS to enable gesture interfaces are discussed.

Based on the observations from the research, this work successfully answers the question of can gesture interfaces be used to consume news and social information and the work also provides actionable recommendations and best practices for developing a gesture interface.

These practical usability and design recommendations offer user experience designers and developers a faster timeline for deployment of a gesture-based interfaces.

ACKNOWLEDGEMENTS

The University of North Carolina at Chapel Hill School of Journalism and Mass Communication and Dean Susan King provided equipment, testing space and support to complete this project.

I would like to acknowledge the Massachusetts Institute of Technology Media Lab and the DepthJS development team for their original development and accepting my contributions to the source code.

My advisor [redacted for privacy] of University of Liverpool was instrumental in my development and completion of the project. Also, professor Laura Ruel of the University of North Carolina at Chapel Hill for her advisement and suggestions regarding user testing and my wife Amy King for her support and copy edits.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Chapter 1. INTRODUCTION

New technologies for interacting with computers are changing the way we consume social information and news media. While touch devices such as the iPad and Kindle have changed media habits and website interaction by using gesture controls and touch interfaces, gaming systems such as the Xbox 360 have integrated motion and depth detection.

At the same time, HDTV flat screens are becoming more prevalent throughout homes and business and better way to interact with those devices is needed. Motion and depth sensing technology is rapidly developing and with the mass release of the Xbox Kinect in MONTH of 2011 by Microsoft, this technology can be found in millions of homes and businesses.

While the hardware is developing and price falls there are efforts to bring this technology to the personal computer and not have it limited to tablets or priority gaming systems. Microsoft enabled the ability to develop motion-based interfaces by releasing the Kinect Application Programing Interface (Kinect API). Open source projects such as DepthJS worked to integrate the Kinect API and the universal web browsers but have seen little adoption.

All of these developing technologies, both hardware and software are now converging into gesture controlled, large screen experience. Users in public places no longer have to use a touch screen kiosk but cannot begin interacting with an application simply by stepping forward or swiping a hand. In homes, users a no longer limited to a remote control or a keyboard and mouse but can control the cursor with the wave and push of their hand.

This work will integrate the most common hardware configuration to test some current news and social websites as well as develop a custom-designed interface for gesture control and interaction with news and information.

1.1 Scope

This dissertation explores the necessary design and user experience concerns publishers and developers should consider when developing for the emerging technology. It implements the most common hardware configuration and integrates gesture enabling software design for the browser. The use of non-touch gesture interface design has numerous commercial, medical and scientific uses, known as the Kinect Effect, but this project focuses on use as an informational tool in the home. This project provides qualitative analysis of existing desktop and mobile interfaces as well as custom designed experiences. It also provides lists of design and development best practices for future development of gesture-based interfaces and applications.

The research and physical user testing is limited to the use of Xbox Kinect and DepthJS though other hardware and software solution are emerging in the market. The interfaces tested include a single website in the social (Facebook) and news (New York Times) category as well as two custom designed interfaces. It does not test the viability of adoption of this technology. It simply tests usability and feasibility of non-touch gesture interfaces to consume information. Usage of the word “interface” in this work refers to a browser-based user interface such as a webpage.

1.2 Problem Statement

Gesture based, non-touch user interfaces are becoming more common in console gaming and other specialized industries such as health care but has yet to become a common and useful way to navigate the Internet, specifically news and social media sites.

This project will seek to answer the question of, “Can gesture-based interfaces using motion tracking sensor technology be used to consume news and engage with social and current events content?” In order to answer this question the project will design, develop and test a software interface and a hardware installation that enables users to interact with news and social information using gestures without requiring touch or extra peripherals in order to understand the common usage.

1.3 Approach and Methodology

To discover if gesture-based interfaces using motion tracking sensor technology can easily be used to consume news and engage with current events content through a custom-designed user interface, an analysis of existing technologies was preformed, contribution to an open-source project was developed, multiple prototypes were built and tested, two finalized or public-ready custom-designed gesture interfaces we developed and existing websites and the new interfaces were tested.

Qualitative analysis research methodology of the interfaces as well as technical observations was adopted. This approach provided input from typical users as well as important technical analysis of the current implementations of the technology and the issues associated with current interfaces.

The main body of research consists of qualitative user tests of news, social and custom interfaces using gesture-based interactions.

This project required the implementation of consumer available hardware with the development of new software and user interface in order to compare the effectives and usefulness of this new technology with existing interfaces. The project took on the three major phases of exploration, development and testing.

The first task was to determine the most useful and most common hardware configuration for using gesture-based non-touch interfaces in a home environment. Screens, computers and sensors were compared to determine the most likely and most useful consumer implementation of gesture technology. The ideal implementation was tested documented and acquired.

Once a hardware solution was determined then the software options were explored. Various consumer products and open-source applications were compared and tested through informal developer experimentation. It became apparent the non of the products or applications available at the time of development would be sufficient to provide the ability to not only test existing web interfaces but to be able to innovate a new user experience.

The most viable and contributive solution to the industry and the project was to partner with developers in the Massachusetts Institute of Technology Media Lab and contribute to the open-source project DepthJS. This solution required some additional development and upgrades.

Development of several testing prototypes and finally the two web-based interfaces were created to demonstrate and test various commonly seen user interface elements. Once the interfaces were developed a five-person user test was conducted.

1.4 Document Structure

This document provides detailed analysis of background research and current literature in Chapter 2 – Background and Review of Literature. Chapter 3 – Analysis and Design, provides information regarding the development and testing of the application. The researching and development methods are detailed in Chapter 4 – Methods and Realization. Chapter 5 – Results and Evaluation, provides explanation of the results from the analysis and user testing. The final chapter, Chapter 6 – Conclusions, provides the findings and details the achievements of the objectives of the project, as well as suggestions for future research.

Chapter 2. Background and Review of Literature

This chapter details the previously published information regarding the use of non-touch gesture-based interfaces. There is a large body of work about the use of gestures on touch screen but a limited number of sources directly applied to non-touch systems which is detailed below.

Non-touch gesture based user interfaces are a fairly recent development in the personal computing world but more and more gesture uses are being discovered and studied. From singers being able to control the output of their voice with a hand motion (Scharin, 2013) to the use of gestures to “feel” a digital representation to advancements in disabled interaction gesture research in multiple fields provide valuable and comparable data and findings even if the technology or application of the technology is different. Specifically, writings regarding non-touch gesture interfaces, have been used in specific fields of study such as medical sciences, law enforcements, geography and the arts. One major field of research is the use of gesture interfaces for elderly users, which because of the long history of studies provides insights into this work.

For the purposes of this discussion I have categorized the current previously published work into four major categories starting with writings about Gesture Interfaces, which focus on the human factors and the use of motion to control a computer. The second major category provides a historical perspective that gives insights into the future by looking at the path of similar touch and gesture technologies. The last and most specific category is Kinect Work which looks at work specific to the Microsoft Kinect sensor and using it to control traditional computers and gaming systems.

2.1 Gesture Interface

In Long's Human Factors presentation on gesture interfaces in 1999 foretold many of the concerns and issues facing the industry today such as standards and inconsistencies but despite being dated offers insights into important human interaction interfaces.

Dan Saffer's book, *Designing Gestural Interfaces* (2009) is the most comprehensive and applicable resource on the subject. This complete technical volume explains the broad implications of using gestural interfaces and the physical and psychological considerations. This book while providing a valuable introduction is limited in the technical detail regarding these interfaces.

One of the first papers to accurately to describe the type of gesture interface control of a computer that current technology is now providing was written in 1997 by Rochelle O'hagan and Alexander Zelinsky in their article, Advanced Topics on Artificial Intelligence. The technology they describe is slightly more advanced than the Microsoft Kinect system but is quite similar to the technology that is soon to hit the market by Leap Motion, detailed in Chapter 3. One of the major topics they discuss is the ability for the detection system to passively acquire and track the user and importantly reacquire if tracking is lost. This is one of the major problems with the current implementation of the Kinect technology noticed developing this project.

As gesture technology becomes more widely adopted in both touch interfaces as well as non-touch users develop habits and tend to have a common way of using the device. One of those habits as presented by Annett and Bischoff (2013) is the use of the dominant hand. But the study found that users could perform the same learned gesture just as well with one hand as the other. This is important in full body non-touch gestures as use over extended amounts of time can result in arm fatigue as later explained this paper suggests that users should be able to switch from one hand to the other even if trained on a different hand.

Looking a gesture research historically provides insights not only to ideal uses but future uses. In the area of geriatric computing Bhuiyan, M and Picking, R have detailed the last 30 years of studies relevant to gesture computing and suggest "*Gesture Controlled User Interfaces (GCUI) now provide realistic and affordable opportunities, which may be appropriate for older and disabled people"* (2011). But there are also insights in this paper that are broader than just elderly users. For example, the collection of quantities data regarding test duration and use fatigue is important for designers to keep in mind when designing gesture interfaces for any audience. This paper also looked at various technology implementations with the goal of ease of use that provided insights into the implementation selected for this research.

A recent and useful writing on the topic details the use of gesture-based non-touch interfaces for gaming in public spaces that was demonstrated at the 2010 Shanghai World Expo. Roccetti, Marfia and Semerararo explain the use of existing tracking systems such as the Kinect and the Nintendo Wii Remote but also explains the use and implications of using a standard webcam. The interesting piece to this research is that using the webcam can provide for more detailed segmentation of the hand making for a more precise interaction than the Kinect or other available solutions. This research also provided example questions regarding the exit survey after users used a gesture system. The paper provides valuable insights into new hardware solutions but like much of the research does not test the various aspects of the user interface experience.

Similarly, researchers Juhlin and Onnevall (2013) are trying to study human gestures while watching televisions not as a user input as most of the writings presented her but in this case using ethnographic observational field work to gain a better understanding of social interaction and the content on the screen. The human observers met many of the same difficulties sensors have in detecting gesture such as the difficultly of detecting a gesture towards the screen verses a daily life motion or conversational gesture. This research provides insights on how best the types of gestures to ignore more than which ones to detect and could be valuable for further implementation of non-touch gesture systems in homes and public places where people will be interesting with each other in addition to the computer system.

Each of these academic sources provide some varied insights and establish a basis for further research but the most helpful insights come from periodicals with are making new and recent articles as the technology progresses. This is especially true concerning development and design of gesture-based interfaces where there no academic writing was found concerning the usability of gesture-based interfaces.

For example, Izkovitch (2012) wrote about the various considerations to take into account when designing for this technology for Mashable.com and App-side provided a list of top five considerations when designing gesture user experiences. Izkovitch provided many useful insights that were vital in the design of the gesture interfaces used in this project.

2.2 Similar Technologies

With such a new and emerging technology, another aspect to consider when is the history of similar devices that use gestures like the smart phones and tablets. Though little historical research is available on non-touch gesture systems much has been written about the path of the smart phone and tablets. Applicable to this technology is the evolution from Mobile-enabled sites, to mobile apps, to responsive design and debate among the different solutions. Pastore (2012) wrote an article on the role of web standards that provide summary of the events and the current debate concerning development for specific devices versus development using responsive design to make a single code base for all devices. “There is a sort of tension between the two approaches [adaptive vs. responsive. Most of the debate is focused on different technical features that evolve very quickly.” This is applicable to non-touch interfaces because there is a similar debate. Should gesture users use the same desktop experience with this new technology or should publishers and businesses produce a specific user experience for this type of interaction?

This historical perspective of mobile computing with regards to gesture computing also provides clue to the future of non-touch computing. For example, in an article about the future of mobile phones for *The Atlantic,* A. Madrigal predicts the use of non-touch gesture interfaces for uses currently services by mobile phones. And an article on GTV shows a patent application Microsoft filed that has direct implication to a gesture home environment.

One of the first technologies to use gestures was a pin-based tablet, which substitute a mouse. This technology has been in use by graphic designers and other creative professionals and certain software has developed specific gestures. According to Lietner, (2013) the pin-driven gestures are being used on large virtual white board displays to make for fast an efficient navigation. These pin displays Leitner research are large displays that cover a significant area compared to the arm reach of an individual thus gestures are an ideal use of navigation. The value of the contextual pin gestures provides a viable alternative on non-touch gesture systems for large screens or for complex navigational menus.

One perspective that has implications in the design and development of gesture applications is a paper about how understanding human movement and choreography. The paper focuses on the design of three-dimensional physical products but the research correlates directly to designing gesture interfaces because human movement is such a vital part of the interface. Hummels and Overbeeke (2007) write, "*Designers can benefit from knowledge and experience from the dance profession, now that the focus of design is shifting from the product itself towards interaction*."

This research has led to a new design method called Design Movement Approach that could be implemented by interface designers when building gesture user interfaces. The approach is to build the elements of the interface around the movement rather finding movements to interact with the elements.

2.3 Kinect Work

It was not until Microsoft released Kinect for Xbox360 that the sophisticated motion sensors reached many households. Kinect still holds the record for the fast selling electronic product ever. Microsoft took the technology farther by making providing an open API which has allowed developers to create new uses for the motion sensor technology and gesture based interfaces are being used for many unforeseen purposes such as security, health care and consumption of information (Schramm, 2010).

Periodical writings regarding non-touch gesture based interfaces provide details into the technical sensor, hardware, integration and usage of the technology.

Lai, in his presentation to IEEE Southwest Symposium (2012) provides necessary scientific documentation that the Microsoft Kinect product was able to detect eight common hand gestures with 99% accuracy in a typical home environment. Lai's research scientifically established the fact that the Kinect product is dependable allowing for further research on the next layer of integration, the application programing interface.

Doulos (2011) wrote a journal article about how to use Kinect in GIS along with Google Earth. The piece explained how to integrate the sensor with the existing Google Earth Application but did not critically analyze the usage and user experience. The must useful part of Doulos's article is the systematic explanation of integrating the hardware and the application.

The dissertation by Aarts (2012) not only demonstrates an application that integrates the motion sensor technology but also provides a detailed analysis of existing Kinect integration services and APIs. For example, Aarts provides valuable insights into the problems associated with the popular KinectJS which was technology considered for this project. Aarts' work is focused on his specific application and the issues faced in development of his specific usage.

An explanation of the workings of the Microsoft Kinect sensor array can be found in the article by Schramm (2012), which provides both an insightful view of PrimeSense, the company that designed and built the technology licensed by Microsoft, as well as a look at the technology inside the case. This article is one of few that profile Prime Sense as most focus on Microsoft.

With such as widely sold consumer device comes popularity and certain social considerations. The use of gesture gaming has been seen across the Internet and on mainstream television and Microsoft produced a video that presented many of the uses people found for the device called “Kinect Effect” (2012). This video provides insights into the different applications of this technology.

2.4 Related Work Conclusion

The current body of work regarding gesture interface design provides valuable insights into the proper design and implementation of a gesture user experience. Though a body of research exists on gestures in various fields, lacking from the available resources are applied uses for information presentation and social media interaction. Advancing the current body of research must be built on previous established technical implementation while pushing the interface into new uses and designs that can be tested. Building off the available work, both peer-reviewed and periodical was the basis of the system and testing and provided the major requirements for the new gesture interface designs.

Chapter 3. FRAMEWORK AND REALISATION

This chapter explains and details the design and architecture of the software that was built. It also provides explanation around the visual design of the user interface and explains the user test design.

3.1 Usage Scenario

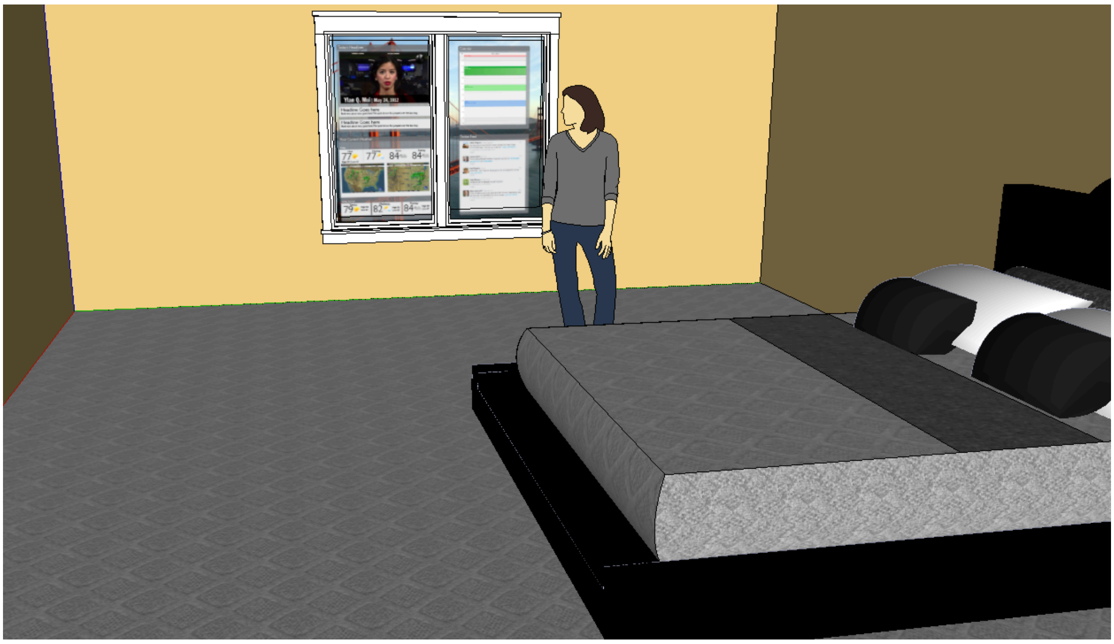


Figure 1: Potential Home Integration of Non-touch Gesture Interface System

In the near future, a 20-30 year old working professional wakes to a sunrise from a distant land on their screen and the system detects the person is awake as seen in the rendering in Figure 2. The top news headlines of the day begin playing while the person begins getting ready for work. Following the short newscast, a listing of the headlines, their personal calendar with their agenda for the day as well as the user's Facebook feed is displayed. The user is able to explore the content and catch up with both current events and social news from online friends.

With the goal of testing the use of non-touch gesture interfaces for use in the home a standard use case was design and each interface and testing environment targeted that use case. With the large-scale infiltration of large flat screen television in the home and he inexpensive availability of motion and depth detection through Microsoft Kinect, it is feasible that the previously presented hardware setup could be found in the living rooms and bedrooms of many consumer's homes. With the user scenario and the use case in mind, software was developed to be reflective of the most-likely adopter of this technology.

3.2 Use Case

As seen in figure 1, a users' gesture is detected and actions content is delivered to the screen via the web interface. The user can passively watch content such as a video or interact using gestures to read articles, read through the news news feed and interact with friends by sharing content. The gestures are the main interaction that allow the user to use the system. When the user finishes consuming content the system will wait for the next user.

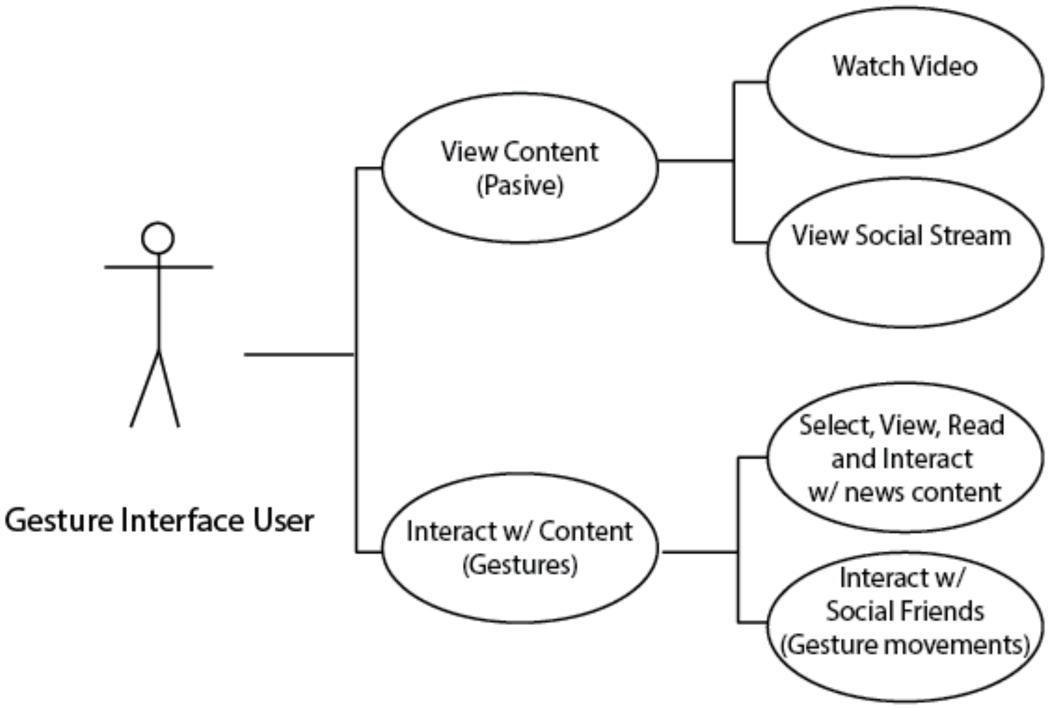


Figure 2: Technology flow diagram displays overview of integrated technologies

3.3 Architecture

Microsoft's Kinect Hardware and the Kinect API power this application. This consumer product provides the necessary motion and depth detection needed to detect user gestures. Following the application flow of figure 1, once the first gesture is detected, content is delivered which a user can either actively interact with using other gestures or they can choose to simple consume the content like watching a video on television and allow additional scripted content to cycle through, similar to an electronic sign in a public place but with news and personalized social content.

In order to make use of this information in the browser, the system integrates with the browser adding a layer of CSS and JavaScript that exists on top of the displayed content as well as ties into the browser's main navigational features such as forward and back. This allows for basic navigation of most HTML content making the amount of available gesture navigable content endless.

The technical architecture of the system requires multiple open-source components and several web-publishing languages. Figure 3 displays how the various technologies work together tor provide a non-touch gesture based navigation system.

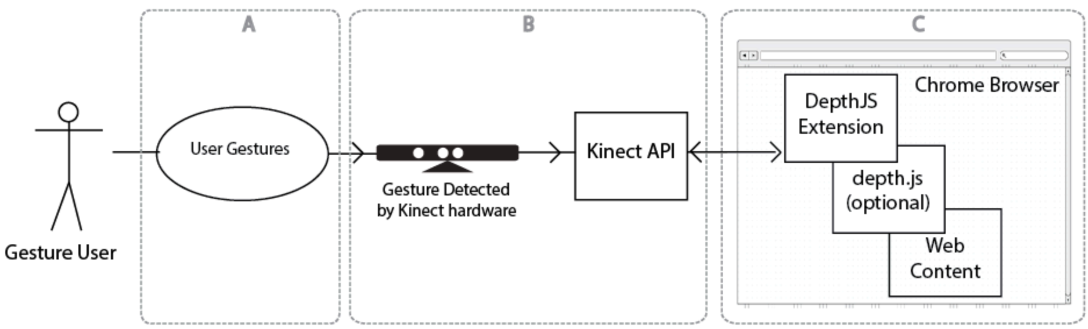


Figure 3: System Technical Architecture

Figure 3 Section A represents the action displayed by the use which is captured by the Kinect hardware seen in Section B along with the Kinect API. The Kinect API provides a programmable interaction based on specific gestures that are detected and dispatches and event along with numerous parameters such as position, speed, distance, etc. Into the browser extension.

The DepthJS extension applies some HTML, CSS and JavaScript that overrides the default functionality of the browser and the web content as well passes information back and forth between the Chrome Browser API and the Kinect API.which is seen in the bridge between Section B and C. Section C displays the layers integrated with the browser where the served web content is displayed natively through the browser. For custom interfaces developed specifically for gestures, a JavaScript file is appended to the content.

3.4 Interface Design

With this use case in mind, multiple prototypes were designed but two are feasible for development and testing nowadays. The wireframes were designed and then evolved into color compressive layouts that would drive the development. Design was completed in Adobe Illustrator and the full vector files along with documentation are available on the public Github project.

3.4.1 Traditional News Site

This interface presented a single dominant element or main story and four secondary stories along with navigation tools at both the top and the left side to access other sections. Figure 3 shows the original wireframe where the basic design elements were determined.

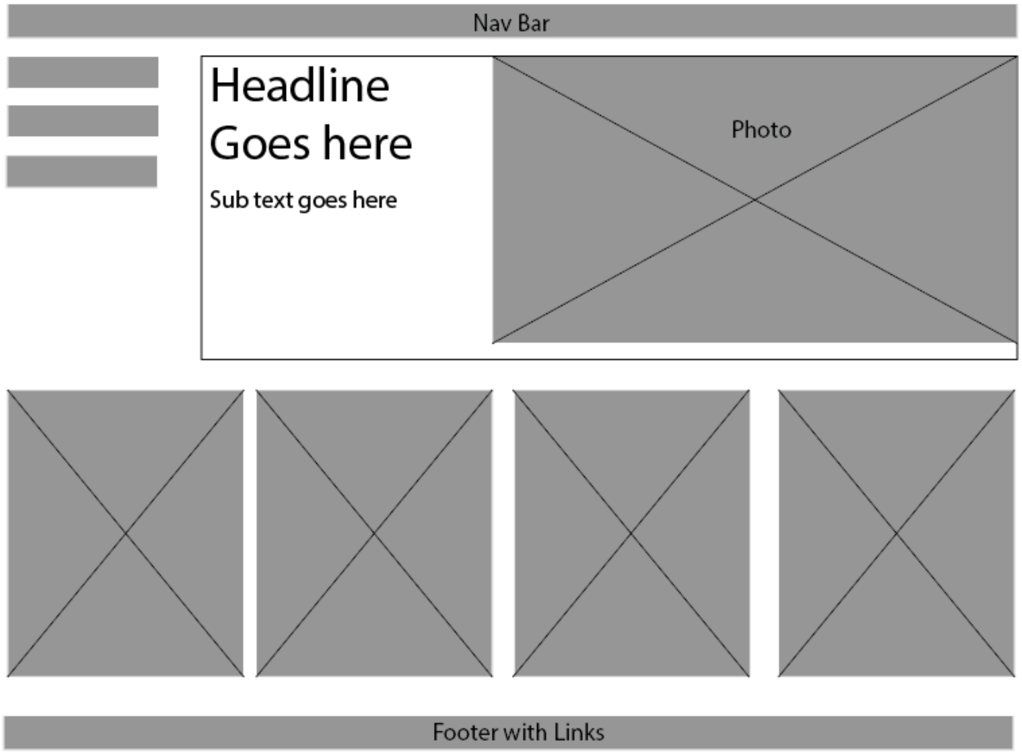


Figure 4: Wireframe of gesture-based news site.

This wireframe in Figure 4 and the color comprehensive layout seen in Figure 5 was the basis for the color compressive layout. It implements basic and traditional navigational items detailed above such as standard links and vertical tabs

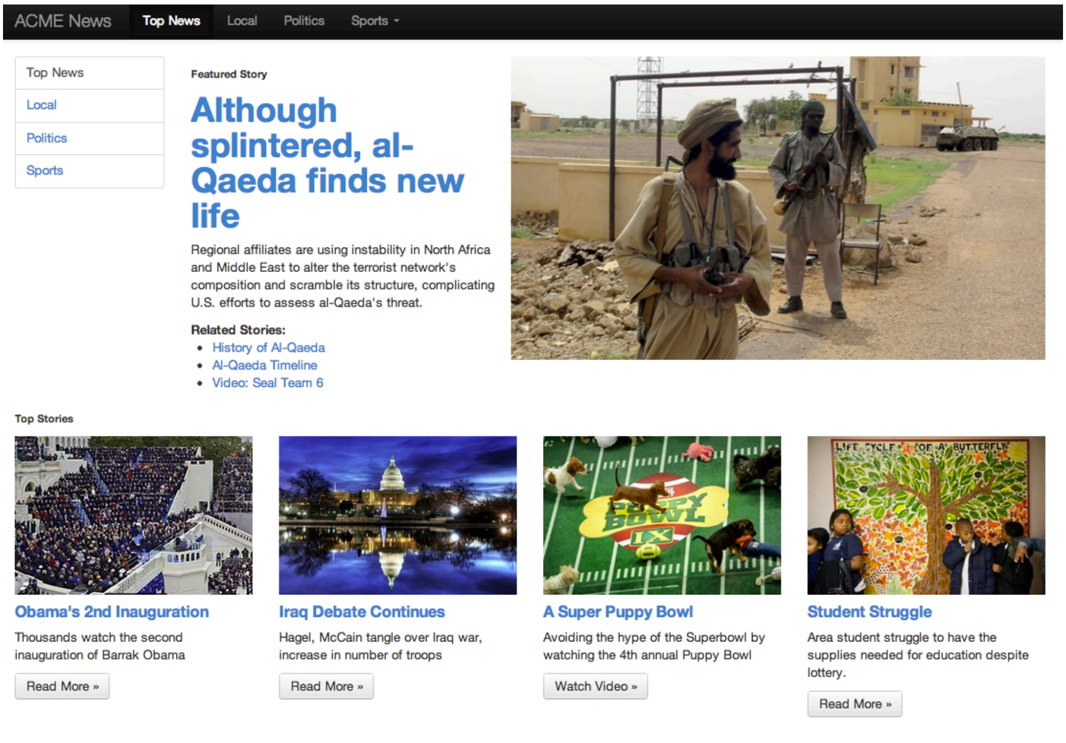


Figure 5: Color compressive layout of gesture -based news site

3.4.2 Large Image Social or News Site

This second interface concept came from various prototype designs where the need arrived to be able to control the interface without a cursor. The Design Movement Approach which first understands the movments and then designs around and with the human movment was implemented in this design. After watching various individuals use the tool and analysing the movements it became apparent that swipping was the easiest action for consuming content. The simple design, which can bee seen in the first wireframe shown in Figure 6, allows for more passive interaction rather than a constant involvement to keep the cursor active.

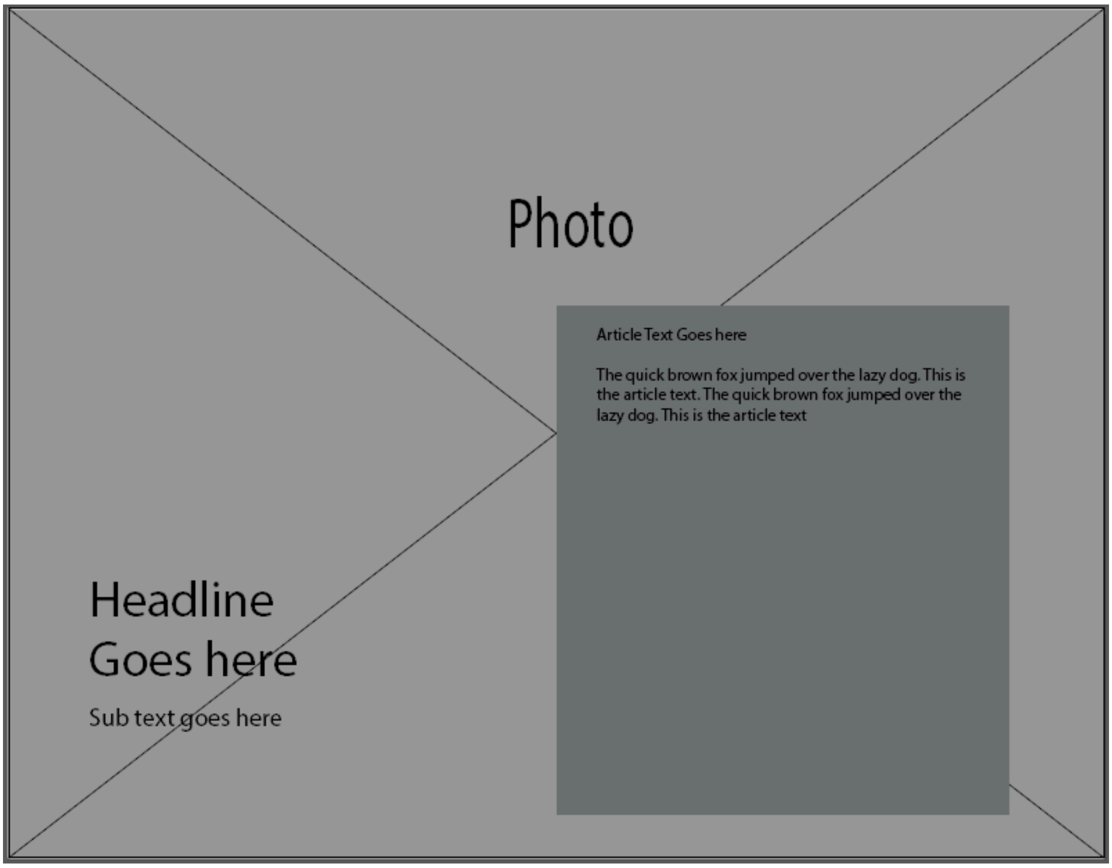


Figure 6: Wireframe of photo-driven gesture interface

Figure 7 shows the final design state. This design could be used for presenting social news or current events. This design seen in this figure also includes an instruction screen that disappears once the user's hand is detected which was added during development based on informal feedback from testers.

The interface in Figure 7 was modeled after tablet photo viewer experiences where a single image consumes most of the screen and additional images are available through swiping. Keeping with the Design Movement Approach the interface was built so that swiping from the users' right to left, slides in a new image and story. This is also consistent with tablet gesture interfaces such as the iPad. This was based on understanding the movement first and making the interface second. The same numbers of stories were presented with the most important being first and all the other stories followed but were not visible except individually after a swipe. Each story was represented with an image and a headline and subhead test to promote the story and the user could use the push gesture to access the text of the story.

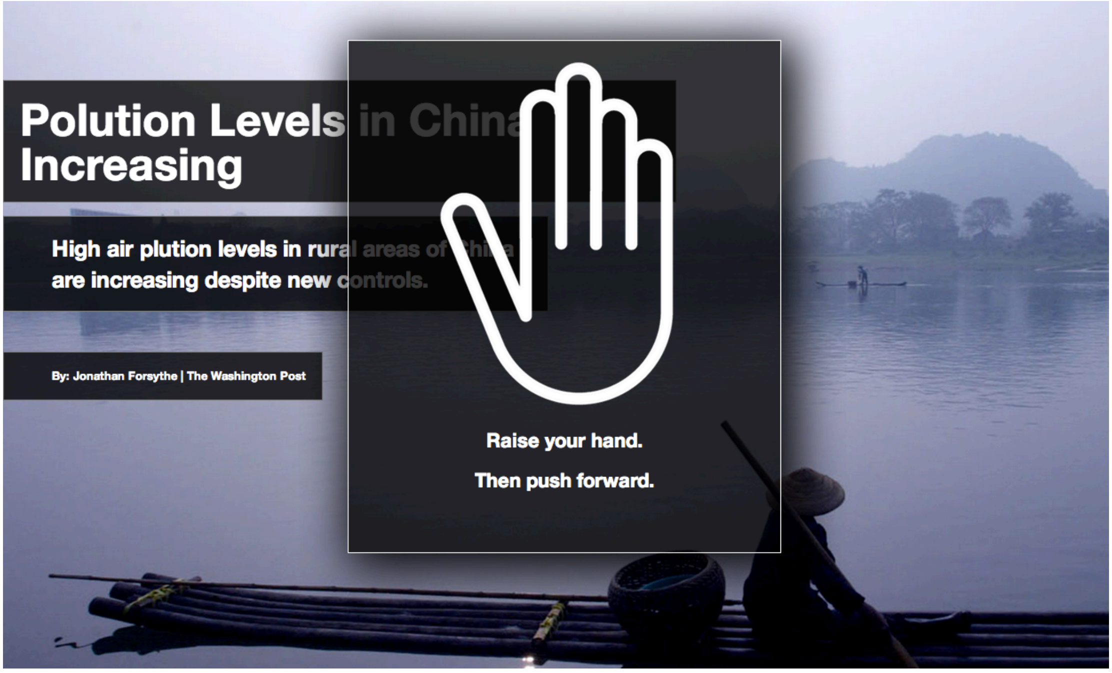


Figure 7: Color comprehensive layout of photo-driven gesture interface.

3.5 Interface Design and Development Methodology

The overarching design methodology employed in development of the interfaces was an adaption of Website Design Method, which is a user-centric method that has been employed widely in Kiosk development practices. Typically, WSDM analyzes the users and classifies users so that each group's individual needs can be addressed. Then interface designs are made and the application architecture is designed. The design is then developed and implemented. Finally the different classified user groups test the product and changes are made as needed. It should be noted that WSDM was originally implemented and often used in oriented programing and though HTML and JavaScript is not object oriented but the methodology still works.

In the case of this project only one user group was classified in order to have an accurate qualitative user test.

The targeted group for both developed interfaces was 20 to 30 year olds who were Internet users and comfortable using technology but had never used non-touch gesture interfaces before. With this in mind it made it easier to develop an interface for this group because many of the user interface elements were modeled after sites this group would frequent.

Development of both interfaces also used an iterative design process where the designs were first wireframed. Then color comprehensive layouts were design and the code was then built based on the layout designs. These designs changed over time with several iterations based on developer testing experience and informal user testing. For example, the direction of large photo interface the direction of the sliding was changed based on ad-hoc feedback from individuals who were in the target user group that tested the interface during development and felt the concept was backwards because it was the opposite of the iPad. Another example includes the addition of the simple instructions screen based on feedback during development.

3.6 Realisation Overview

The project encompassed, integrating existing hardware and developing interactive software that would enable creation of a user evaluation test. The project required examining existing web interfaces as well as custom user experiences. The project also required looking at the different technology, applications and testing situations in this computing area. The project was executed in three distinct phases in a particular sequential order that built on the previous phase. The phases were Hardware Selection and Integration, Software Development and user testing.

3.7 Hardware Integration

In order to effectively test gesture-based interfaces the hardware must be integrated into an environment that can be controlled and monitored. This requires the physical install of hardware and software configuration in the testing location as well as the development environment.

The central device in gesture interfaces is the sensor technology. There are several gesture-enabled motion tracking technologies available on the consumer market but by far the most popular is the Microsoft Kinect, which set a record for the “fastest selling consumer device” with over 10 million sold. Typically, the Microsoft Kinect is used with the Xbox360 gaming console but Microsoft released a publically available API that enabled desktop applications to interface with the sensor.

In addition to a large market penetration, Microsoft also publically released the Application Programing Interface (API) that enabled developers to create custom applications that use the Kinect sensor. The Kinect API is vital to innovation in this area and the large adoption and usage of this API made it an appealing option for the sensor technology.

As seen in Figure 8, The Kinect device consists of an infrared emitter and depth sensor, a color VGA video sensor and a four-microphone array that are all used for motion and depth tracking. The camera with a resolution of 620x480 pixels captures images and depth at 30 frames per second and used in conjunction with the infrared grid that is invisibly broadcasted while in use. This data is processed and made available through the Kinect API.

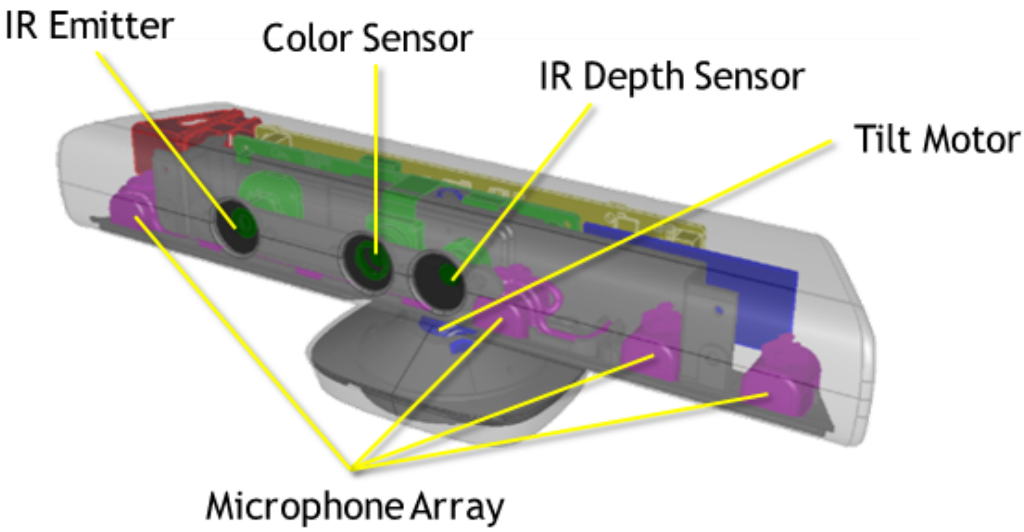


Figure 8: Microsoft Kinect's major detection components.

Many of the available consumer sensor technologies are similar to the Microsoft Kinect in resolution and tracking methods but it should be noted that a new and entirely different method of motion tracking has recently been be release. Leap Motion provides a more precise tracking matrix in a smaller space enabling the use of smaller movements and hand gestures rather than full body gestures. This product had not been release at testing and development but could provide a new way of interacting with websites. This technology was not chosen for testing due to the lack of consumer availability.

Selection of the ideal screen size required less detailed analysis. A recent consumer study from Morgan Stanly reported that the majority of consumer though about 45 inches was the ideal screen size for a “smart TV.” Though this project does not use or require a Smart TV, which is a television with an Internet connection and interactive applications, this became the basis for the size selection, as a smart TV is similar in usage from a consumer’s perspective. Also in consideration is the distance from the screen when interacting with gestures. The Microsoft Kinect requires the user to be at least five feet away and may applications suggest 6 to 10 feet and screen that allowed users to be able to read standard web text on a 1920x1080 resolution display from about six feet away had to be chosen. After considering each of the factors, a 44-inch 1920x1080 LCD flat screen monitor was used for both development and user testing.

It was important that the software be computer platform agnostic and both Apple Mac computers as well as Windows based PCs be able to be used for gesture-based interaction. A MacBook Pro was used for the user test do to personal availability but the applications will work on either major platform.

Hardware Implementation

- 1920x1080 HDTV
- Mac Computer
- Microsoft XBOX Kinect (Motion Tracking Sensor)
- Multi-camera video recording for monitoring

3.8 Software Development

Various options for integrating web sites with the Kinect hardware were considered. The Kinect API provides a lot of functionality but an intermediary application or utility is required to allow web browser to interface with the Kinect sensor data.

3.8.1 Framework and Library Considerations

The first consideration was a browser plugin ZigFu that has a JavaScript API. ZigFu is designed primarily for use with custom generated Flash or Unity 3D content and most examples are online games. The available API events were limited and did now have a way show a traditional cursor. This plugin would work for custom development but would not allow for the testing of existing websites like NYTimes.com or Facebook.

JavaScript library Kinesis.io was a consideration for development and there are several working examples of non-touch gesture-based interfaces developed for the Kinect but this library was not multi-platform enabled and only worked on Windows machines. This solution also has the drawback that it would not allow for testing of sites not specifically developed on this platform.

KinectJS is a popular JavaScript library that also provides access to the Kinect API. This library is designed and used primarily for gaming and not used for traditional browser navigation.

3.8.2 Depth JS Solution

Through testing of the previously listed solutions it became apparent that not only would new interfaces have to be built but in order to compare and test existing interfaces some middleware or utility application would be required to bridge the gap between the Kinect API and the browser.

Developers in the Media Lab of MIT created such a project in the form of a browser plugin. This plugin was built for Google Chrome, which provided the cross platform availability as well as the conceivability that this technology could soon be widely adopted.

DepthJS is both a browser plugin with a JavaScript API that allows developers to create custom events tied to specific gestures detected by the Kinect but it also provides default functionality for existing websites through the Chrome Browser API.

Though much development and effort was spent on DepthJS during a short development cycle, the codebase has not been updated since some major changes in Google Chrome that changed access to the browser API as well as how plugins interact with the browser. This became problematic and computers with the latest browser updates could not run the DepthJS plugin.

In order for the extension to work with the latest version of Chrome and be cross platform and enable testing of multiple sites several changes needed to be made to the existing code. The Manifest needed to be upgraded to the newest version and depreciated items removed as well as all inline links to scripts had to be replaced with absolute links to JavaScript files. The way that the extension or plugin sent and received messages from the browser API had to be changed to the new syntax. Also, new menu files had to be created. Everything was documented and the developers at MIT merged a pull request in October of 2012 and the plugin publically was distributed on Git Hub.

When the DepthJS plugin is enabled in Google Chrome, websites using HTTP protocol, not HTTPS, can be controlled through specific default gestures. Use of these default gestures allowed for the testing of existing news and social sites. This functionality also makes it feasible for a large user adoption of the technology because more than just custom sites can be navigated with gestures.

3.8.2.1 Gesture: Waving

Holding up one hand and pressing or pushing the hand forward enables a cursor. A blue circle appears in the center of the screen and the user can move the circle around the screen. To select a link, the user presses or pushes forward and any links that are within the radius of the cursor circle are presented in a simple menu and the user can select a specific link.

3.8.2.2 Gesture: Vertical Swiping

Users can scroll up and down the page using an open hand and moving it quickly up or down pushing towards the top or bottom of the browser window using a swipe from top to bottom or vice versa.

3.8.2.3 Gesture: Horizontal Swiping

When a user swipes from left to right or right to left the browser will emulate the pressing of the forward or back browser buttons, respectively, allowing users to navigate multiple webpages.

3.8.2.4 JavaScript API

In addition to the default gestures, DepthJS also provides a JavaScript API that enables developers to create interfaces that make the best use of gestures and overrides the default gestures.

3.8.2.5 Issues

DepthJS, though viable for this project has numerous issues that hinder it from being widely adopted for consumer product. The main issue being the browser extension is difficult to install and requires Google Chrome to extensions to be in Developer Mode. Another issue that plagued development with DepthJS is the seemingly inconsistent connectivity between the hardware and the browser. When these problems occurred the only solution was to quit the browser, unplug the device and try again.

It should be noted that the original developers who started the project provided feedback and were receptive to updates and additions that improved the code. Therefore, the potential is there for DepthJS to become an easier application to work with in the future.

3.8.3 Interface Development

Once the browser extension was complete developer of the user interface began using the DepthJS JavaScript API. The two interfaces were created that would allow for the testing of the more popular user interfaces elements and default gestures.

Figure 9 displays the concept of the technical integration with the Kinect API. Starting from the top, the Kinect API interfaces directly with the DepthJS browser extension in Google Chrome. The Chrome browser displays the HTML pages and the associated languages or CSS and JavaScript.

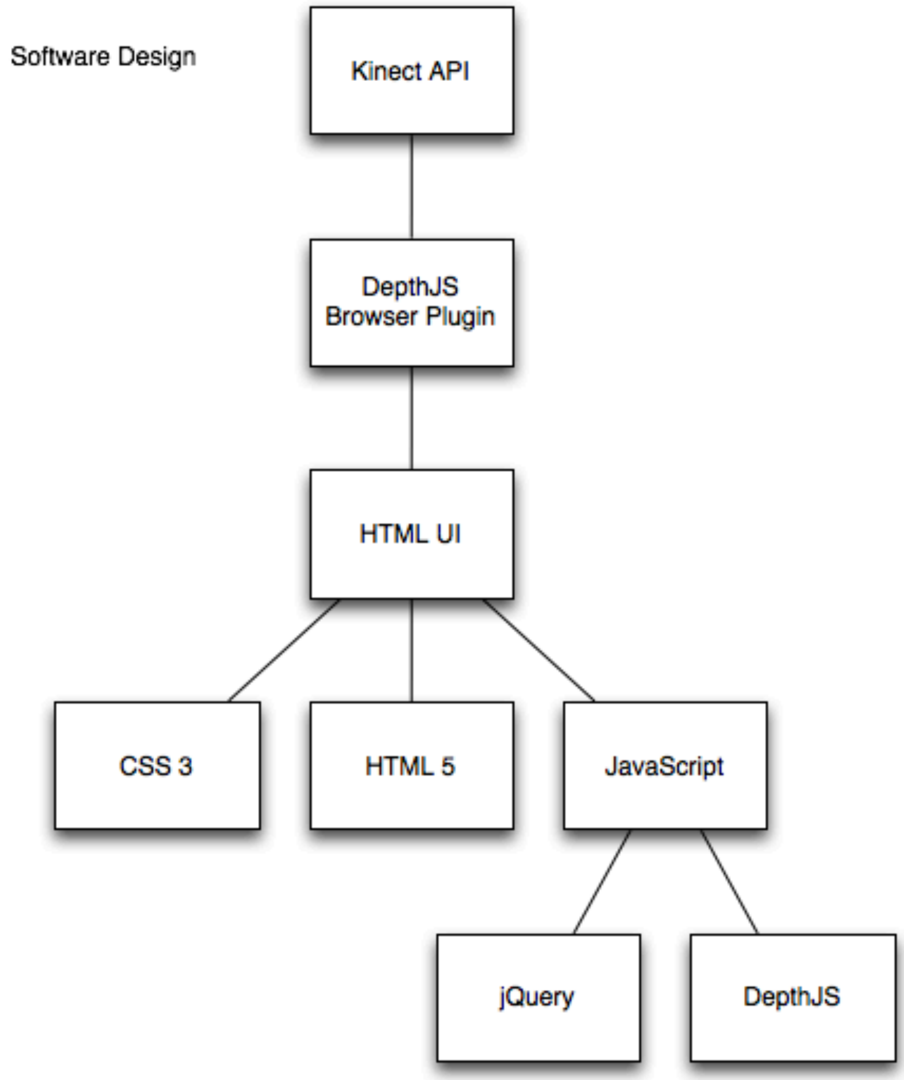


Figure 9: Diagram of software technology integration

Development of the two user interface designs was created using HTML5, CSS3, JavaScript, jQuery and the Twitter Bootstrap responsive framework. This industry-wide adopted framework includes easy implementation of user elements such as tabs, vertical menus, dropdown menus and buttons.

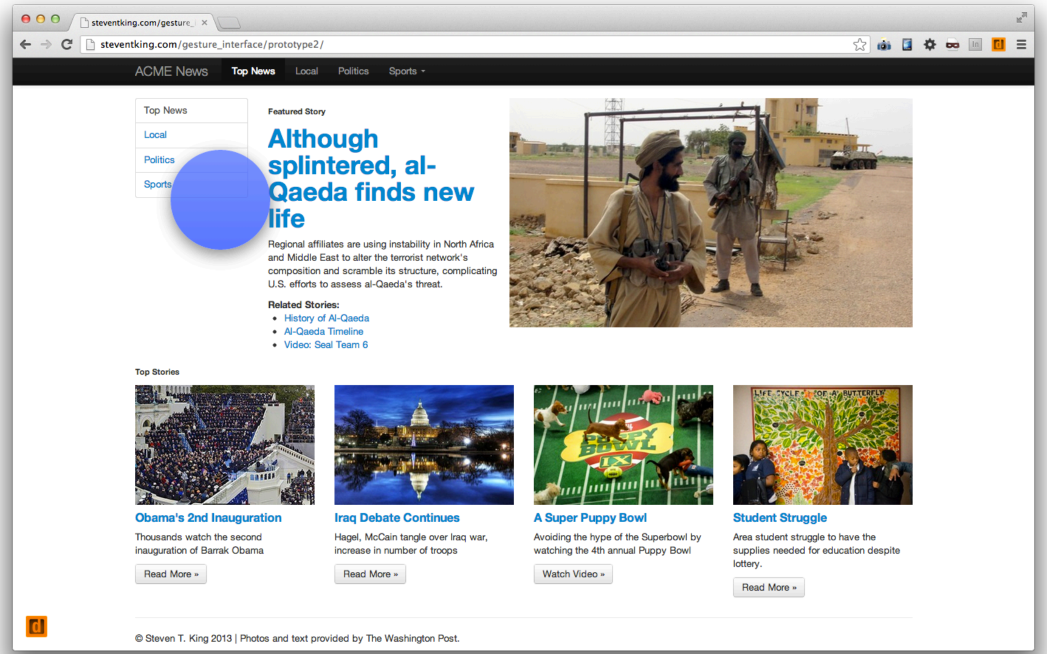


Figure 10: Screenshot of gesture interface using DepthJS cursor interaction.

Users have a blue circle, (shown in figure 10) which represents a cursor that is controlled through gestures. User can click on links, button, and menus and on the photos. When a user clicks, if there are more than one link under the hit space, the space of the blue circle, a menu of the selected links appears as an overlay submenu as seen in figure 11.

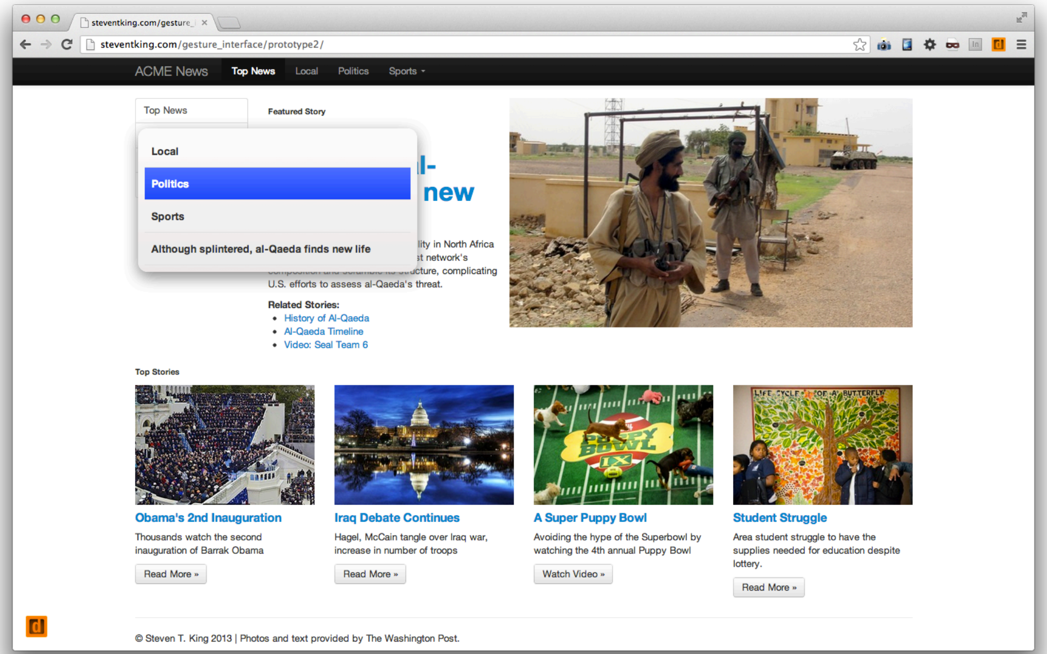


Figure 11: Screenshot of submenu of selected link on gesture interface using DepthJS

The second and large photo interface was developed with HTML5, CSS3, JavaScript, jQuery and the Twitter Bootstrap responsive framework as well along with implementation of a jQuery carousel plugin.

Figure 12 displays the slider interface where new stories/photos are displayed with swipes and text is displayed with a push gesture.

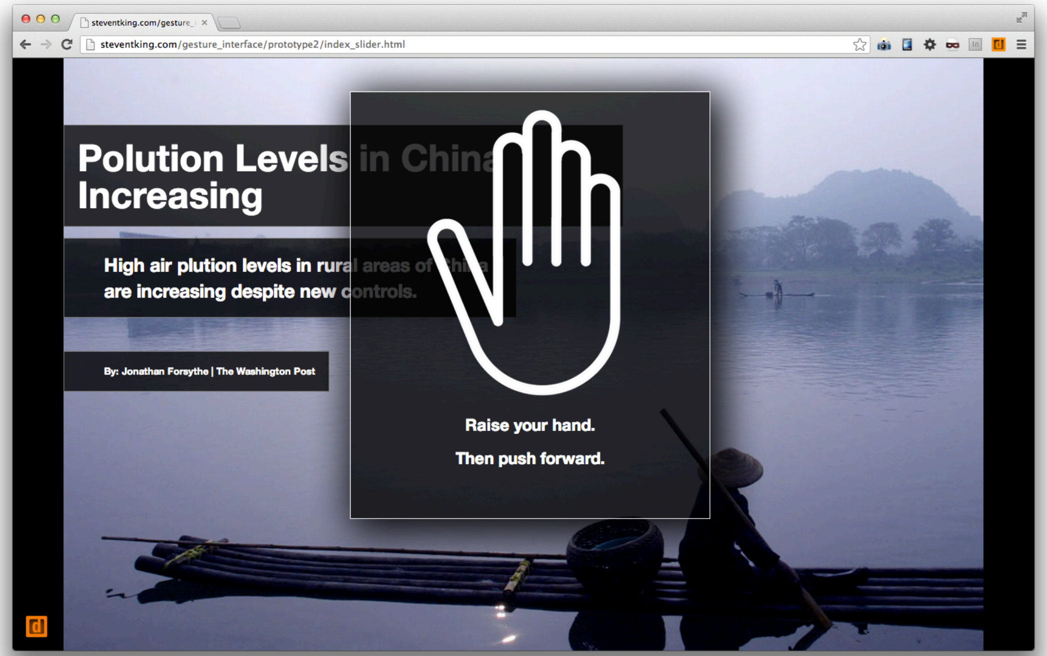


Figure 12: Screenshot of photo-driven gesture interface prior to user detection.

3.9 User Testing Solution

The final technological hurtle was to find a way to emulate a mobile device through a desktop browser so that the Facebook and NYTimes mobile sites could be tested using non-touch gesture interfaces. This was accomplished through the Chrome plugin UA Spoofer that changes the http request headers to resemble a mobile device.

Once integration of the hardware, development and testing of the browser extension, two interfaces and mobile emulation was complete the formal user test was preformed.

The task-driven user evaluation required users to complete a series of navigational tasks on six different interfaces of social and news websites. Each test was monitored by the researcher and recorded with multiple cameras so the actions could be reviewed after the test.

Five users or evaluators were recruited through a university-wide email list to find users who were familiar with the Internet.

Prior to the evaluation, users were instructed on how to use the system and be allowed time to interact a website not being tested. This allowed the user to become familiar with the new experience of non-touch gesture control as well as gave time for the “wow factor” to wear off.

The user evaluation script, Appendix A, details the steps of the user evaluation. The researcher administered the user test each time directly following the script and all users were tested within 24 hours to ensure consistency in the test.

During the tests, the order of the tested interfaces, or websites, was random and each task was timed. After completing the user test, the evaluator was asked a series of questions regarding the three different experiences through a web-based survey. This survey found in Appendix II, in addition to observations and timed tasks provided the data necessary to draw conclusion as to the effectiveness of non-touch gesture-based interfaces.

Chapter 4. Evaluation and Results

This chapter presents the testing methodology, implementation and results of the previously stated design. It details the user test subjects, testing environment and how the user test was conducted as well as the findings.

4.1 Testing Methods

A task-driven user evaluation was designed to test and compare existing news and social media sites as well as the interfaced developed for this project.

The user testing approach was modeled after Jacob Nelson's five-user test (2000). The user test was designed following a combination of the user test presented by Nelson where he states, "*After the first study with 5 users has found 85% of the usability problems*" and the usability testing method presented in Steve Krug's usability book, Rocket Surgery Made Easy (2010).

The task-driven user-testing model developed for this project provided a controlled environment that enabled qualitative comparison of existing and recently developed non-touch gesture interfaces. The design of the test is dependent on these controlled variables:

- Each user will receive the same instruction and become familiar with gesture interfaces prior to testing.
- Each user will be familiar with technology and demographically similar to the user group categorization.
- Each user will receive the same tasks and instructions.
- Each user will test the same interfaces that are presented to them in random order.
- Each user will be right hand dominant.

In order to meet the before mentioned user criteria, a group of five individuals were selected from respondents to email requests distributed on multiple email list services across the

University of North Carolina. The makeup of the group was selected based on the WSDM user group categorization from interested respondents to the email.

4.2 Testing Group Makeup

The five member testing group consisted of two Men and three Women in their twenties or thirties who were comfortable with technology and used the Internet daily. All were right handed and ranged in height from 5’ 2” to 6’ 2”.

Assessment of potential test user’s conformability with technology and usage of the Internet on a daily basis was determined based on respondent’s self selection and their answer to these two questions presented in reply to initial contact, “Would you consider yourself comfortable using new technologies?” and “Do you use the Internet daily?”

It is important to note that all users were right handed because though there should be no differences in the ways a left handed or right handed person interacts with the system, the sample size was not large enough to make comparisons on this factor so only right handed users were tested.

4.3 Testing Environment

The test was conducted in a user testing room provided by The University of North Carolina at Chapel Hill’s School of Journalism and Mass Communication in Carroll Hall. The room had white walls and controllable lighting.

For the user testing, the screen was mounted with the top edge 6 feet high on an 8’x10’ wall. The Kinect sensor array was placed on a small bookshelf three feet off the floor directly centered under the screen. The computer was placed on the shelf below the Kinect. One camera was placed facing the user just above the sensor. The second camera was placed behind the user's right shoulder so the user and the screen could be seen. This 3D rendering of the lab testing setup can be seen in figure 13.

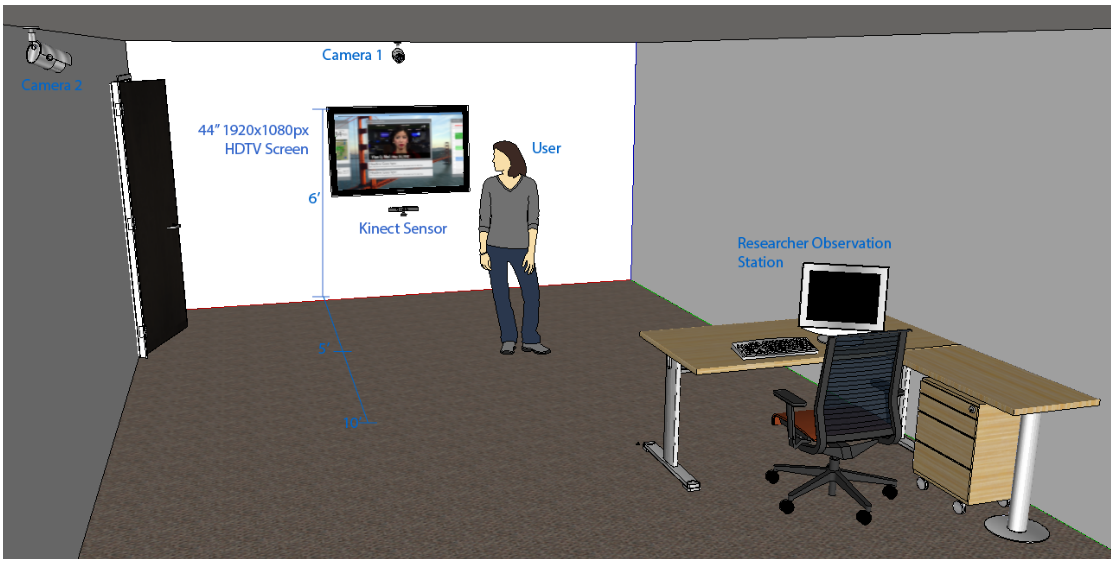


Figure 13: 3-D rendering of non-touch gesture testing environment.

For the purposes of later study the user test were recorded using two Cisco Flip Cams and this device was chosen due to available and ease of use but became problematic during testing as the cameras would at times turn off and stop recording with notification.

4.4 Testing Protocol

Test users were selected and appointments for testing made via email. All of the actions of the researcher followed the User Evaluation Script in Appendix A. Prior to the task portion of the test, users signed the proper paperwork and were verbally informed that video and audio was being recorded for research purposes. The users were briefed on how to use the gesture interface through a simple and consistent demonstration. Following the demonstration, each user was allowed time to interact with a website that was not being tested. This time of exploration and learning allowed the user to become familiar with the new experience as well as give time for the "wow factor" to wear off. Typically, the instruction time took about one minute and the user took between one and two minutes before they alerted the researcher they were ready to continue.

The test required users to complete a series of navigational tasks on six different interfaces of social and news websites. Each test was monitored by the researcher and recorded with multiple cameras.

Tasks were designed using the following criteria:

- Task requested should be a common action performed on that site using other interfaces.
- Tasks should utilize different interface elements, menus, links, buttons, etc.
- The collection of tasks should utilize different parts of the screen in order to test range of motion.

The researcher administered the user test each time directly following the User Evaluation Script to ensure consistency in the test for each user. Users were asked to complete three different tasks for each website and each task was timed and recorded for comparison. If a task could not be completed either after the user gave up or after three minutes of trying the researcher asked the user if they would like to continue trying or move on to the next task. Appendix B provides the researcher's worksheet used to record success, failure and time to complete each task.

The sites tested were, nytimes.com, mobile.nytimes.com, facebook.com, mobile.facebook.com, the gesture-enabled traditional news site and the large photo interface. The order of interfaces presented was random. The randomness of the order presented was important to eliminate the factor that users become more familiar with the gesture technology over time which could skew the results if the same interface was always tested first or last.

At the conclusion of all the tasks required, users were asked a series of questions regarding the six different experiences through a web-based survey. The survey provided the basis of the qualitative data to evaluate each interface. This survey can be found in Appendix C.

4.5 Data Tally Process

After all user tests were complete, the results of the survey were cataloged and tallied so comparison and observations could be made. The times to complete each task were tallied and averaged to provide can some broad observations and comparisons but the sample size is not large enough to make quantitative conclusions.

Because the proximity of links was a factor in the user test, the link density of each website during the testing time was calculated. Counting the number of links on the visible page and dividing by the area of visible webpage real estate provided the link density figure and each site was compared to the others.

4.6 Data Comparison

The data collected by the researcher through observation as well as through the exit survey was tallied and extracted.

The most promising data collected which suggest that this type of interface could become and adopted technology was the response to the question, “Could you see yourself using this type of gesture interface in the future?” 80% of the user test group answered yes. The person who answered no, they do not see themselves using this type of gesture interface also said, “I might use it for "enhanced-TV" and games” or … “if I had a nice big screen and I wanted to use it for reading, web browsing, etc. without leaving the couch”

Another interesting data point was that 100% of the users when asked, “If you were using this gesture interface to view websites, should publishers such as NYTimes and Facebook provide, a) a custom designed user experience for gesture users or b) is the main or regular desktop site acceptable?” selected a custom designed user experience. This would be similar to a gesture app or motion enabled website rather than just the basic website.

The test compared the different web browsing experiences, desktop, mobile and specifically designed gesture interfaces. As hypothesized, the websites that were specifically designed for gesture interface has no failed tasks and received a more favorable rating than the desktop or mobile sites. Figure 14 shows the number of failed tasks and the user selected easiest site are directly correlated as the "easiest" sites are those with no failed tasks and the "hardest to navigate" have more failed tasks.

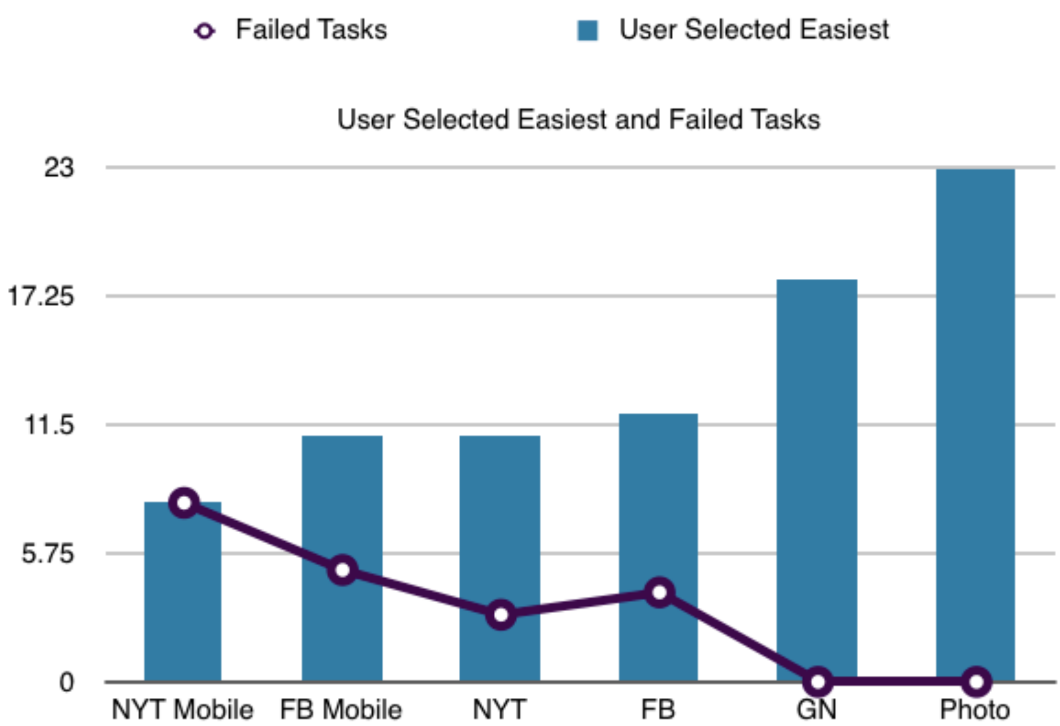


Figure 14: Results tally of user selected easiest interface correlated with number of failed tasks.

The time to complete each task shows a similar picture. The "easiest" sites, those on the right of the graph in Figure 14, had shorter average times to complete tasks with the Photo site averaging 10 seconds per task and both NYTimes sites were over 20 seconds.

Of the two custom interfaces the large photo interface scored slightly higher and only one person found it difficult to navigate.

In order to test the hypothesis that a simplified interface might provide a better experience both the mobile and desktop sites of Facebook and the NYTimes was tested. It was thought that the mobile sites might be easier to navigate than the desktop sites. Concerning Facebook, there was little difference in the number of failed tasks or the difficulty scored by the users. The New York Times mobile site actually scored more difficult to navigate with 100% of the users reporting it difficult or impossible to navigate.

On of the factors in ease of use is link density. The desktop sites of The New York Times and Facebook had, as expected, the large number of links per inch with a density of .38 and .39 respectably. Their the mobile counterparts was about 75% less with a density of .054 and .07. The traditional gesture news site had a density of .068, which is comparable to the mobile versions. Figure 15 displays the density of each site.

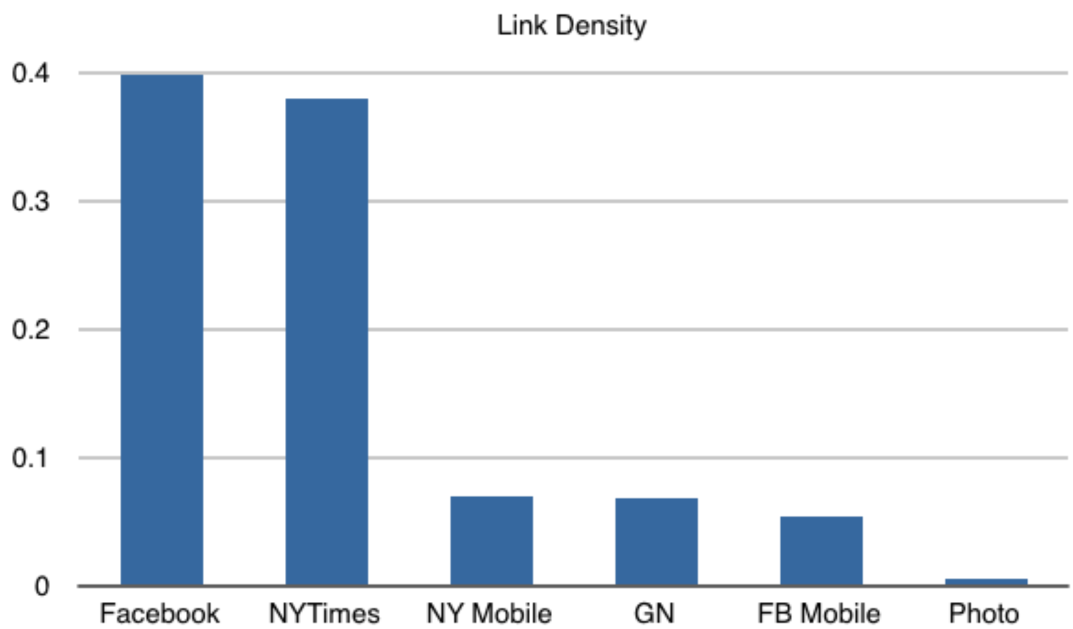


Figure 15: Results of link density measurement of interfaces

4.7 Antidotal Feedback

Overall, users seemed to like the technology concept but all noted ways to improve it as well as specific uses for the technology. For example, one user said, “I'd use it in public places, where touching stuff has germs, or would be otherwise icky. This would be nice while waiting for a subway, or train."

Two of the users noted uses as an addition or enhancements to the television rather than as an interface to a computer. "I might use it for 'enhanced-TV' and games," was one respondent's answer. The other user that noted TV wrote, "The reason I would use it would be to have a more immersed experience with watching videos and consuming news. One user wrote almost a summary of the use case presented above and stated, "I would consume news or social media with this type of gesture interface because it seems like it would be really easy and convenient from the comfort of my bed or home."

Users were also asked, "What was the most difficult part to navigate?" One user wrote, "Clicking on small buttons that have even more menus. The menus were easy, the small button were hard." Another noted the menus as difficult to select a specific item.

4.8 Researcher Observations

After receiving the demonstration of the technology all five users smiled and showed excitement trying the technology for the first time. They all seemed to grasp and understand the basic or default gestures and were able to interact with the page in a matter of seconds.

It became obvious that shorter users had trouble reaching the top of the screen. Despite being told, "you can move around as needed," three of the five users acted as if their feet were attached to the ground. The shorter users would lean, stretch and stand on their tip toes but would not move closer to the screen to reach the higher point. Also, the shorter users' tasks took longer to complete than taller users' tasks.

When user were using the submenu that is prompted when the hit space contains more than one link, many users accidently selected the wrong menu item. This was quickly rectified and completed on the next attempt. It seemed the farther out a person extended their hand to push, they had trouble selecting the next item. Users did not seem to realize they could pull their hand closer at anytime.

At the conclusion of the task portion of the user test, about 20 minutes of usage, three respondents made verbal comments about their arm getting tired. One switched from using the right arm to the left before the end of the tasks.

Chapter 5. DISCUSSION

Based on the user test and other research, this chapter provides a discussion proposing a recommended interface design as well as a list of best practices when designing non-touch gesture-based user interfaces. This goal of this chapter it to provide tangible, actionable examples and information to the journalism and technology industries that make developing for this platform easier, faster and a better user experience.

5.1 Ideal Design

Based on the research outlined in this document an ideal non-touch gesture based interface has been designed. This interface, presented in figure 16 presents social network feeds, the user's agenda for the day and local news and weather with simple swipe-driven controls.

In the Link Density summary (figure 15) the presence of additional content is communicated to the user as well as reinforcing the concept of swiping to access it by placing it slightly visible at the edges. There are only a few links on the page at a time and are accessed by swiping up and down to move a highlight and then a push selects the link. The video plays when the content is loaded to avoid the need for an extra push of the play button. Pushing forward or swiping at anytime pauses the video. The background is a large video image with subtle movement to make the most of the visual medium and the content can be swiped away to show just the scene and hear the wind sounds of the bay.

This interface can be seen in this 3D rendering in figure 2 of an ideal home installation where the user can access the relevant content while moving around the room.

Figure 16: Ideal Gesture Design

5.2 Best Practices for Designing Non-touch Interfaces

Below is a list of best practices for designing non-touch gesture based interfaces. These suggestions are based on historical research and user testing.

1. Design for widescreen 16:9 aspect ration with a resolution of 1920x1080 in mind.

2. Design expecting the user to be between 5-10 feet away so content and user interface elements should be larger than those presented on desktop screens.

3. Use Swipes and Push gestures whenever possible rather than traditional cursor control.

4. If using cursor control, separate links by at least 72 pixels on each side to enable easy selection. Use less links and more open space whenever possible.

5. Place important navigation on the side or bottom of the screen to avoid users not being able to reach the top of the screen, especially if using a public installation.

6. Avoid complicated drop down or submenus.

7. Provide access to important information such as weather, top headlines, social notifications on the top level or main page to avoid the need for selecting and navigating to another page.

8. Avoid rollovers to communicate important information, as that state is unavailable in gesture interfaces.

9. Use large, dominant elements with a single link rather than packaging lots of content together in a small space. Remember, this is a visual medium.

10. Create gesture interfaces with a specific purpose rather than just for having another platform. Develop the design and produce content for the specific use case a non-touch gesture interface can provide.

5.3 Code Base

All of the code for this project is provided with an MIT Open-Source license and made available for download or contribution through Github. This code can be a starting point for future development of non-touch gesture based interfaces as well as a location for community development of the user experience.

Chapter 6. CONCLUSIONS AND FUTURE DIRECTIONS

The main purpose of this project was to implement and a non-touch gesture-based user interface and test the ability to consume social and news media through that interface. The technical implementation of this project was successful as multiple websites and custom built user experiences were tested. The user-testing portion of the project provided insights into user behavior and the viability of wide adoption of this type of user experience.

This chapter details the important lessons learned and the future of the technology as well as future research on the subject.

6.1 Research Question Findings

This project sought to answer the question of, “Can gesture-based interfaces using motion tracking sensor technology be used to consume news and engage with social and current events content?”. Through user feedback and research ovservations along with development of software and implementation of hardward this project has successfully answered the question. In short, this research found gesture-based interfaces can be used for consuming and engageing social and new content but the ideal user experience requires specific user experiences to be developed for this emerging technology and audience.

The implementation and user researched provided adiquate data to suggest gesture-based interfaces using motion tracking sensor technology can be used to consume news and engage with social and current events content. The fact that a user test was able to be sucessfully completed suggests that non-touch gesture interfaces can be used to consume and interact with social and news content. The content was able to be consumed and users were able tointeract with that content without touching a screen or other input device. But based on the user test data and research overservations it is obvious that this new technology will require user experiences that are designed specifically for this technology and interaction.

It will not be a positive experience to the user if publishers simply depend on existing web designs to deliver information to this emerging audience. Non-touch gesture-based interfaces will need to follow the historical example of their gesture predecessors, the touch smart phones. As smart phones hit the market with the capabilities to view standard websites, publishers quickly realized that it was not an ideal user experience and custom applications and mobile websites were created. With the non-touch gesture interfaces in homes, the screen size is larger and the ability to control the browser is limited and those factors, along with others which are detailed in the best practices documented in previous chapter, and therefore requires a custom designed user experience. One example is that most websites have the main navigation at the top of the page and in the user test this was found to be problematic for some users. Therefore publishers should place the navigation on the side or bottom of the page.

6.2 Additional Findings

The design and development of the application and the interfaces as well as the testing of existing and custom interfaces provided valuable insights into the use of non-touch gesture based consumption of news and social information in addition to the main research question. Insights gained in the discovery of this concept will help shape future design, development and media presentation for non-touch gesture-based interfaces.

The standing experience gesture experience, might also required a different type of content based on the use case rather than simply presenting the same content seen in the desktop version as some publishers are doing with mobile sites, giving condensed content specific for the on-the-go audience. This required standing experience, which is a limitation to the Kinect gesture detection, is not ideal for long narrative content such as feature articles or documentary videos. This was evident when users noted their arms were tiered after 20 minutes of use. As technology develops and detection from a seated position is developed, for example the user is seated on a couch, it might provide a longer user attention span and thus providing longer, narrative content might be ideal.

It was hypothesized that simplified mobile sites would be a better experience using gestures but that was proven wrong through the user test. There was no advantage to using the mobile sites to consume social or news information and in the case of The New York Times mobile site it was more difficult. When developing non-touch gesture-based interfaces it is not just about simplicity, which the mobile sites provided, it is also about placement on the page and link density. As users noted it was difficult to select links when they were close together. This concurs with the conclusion stated above that custom interfaces for gesture experiences should be presented rather than just a simplified version.

Through the user testing observations as well as the feedback it was apparent that swipes, both left and right as well as a push action was the easiest way to interact with content. Users were able to select and interact with multiple items easily. Four of the five users tested noted swiping as the easiest way to navigate. Designers of these gesture experiences should try to make use of the default gestures like swiping whenever possible and use the cursor concept less often.

When a traditional cursor concept is used, designers should allow for more space between links and buttons. As seen in the user test, the pages with a link density of less than .1 were easier to navigate than pages with more links on a page. It should be noted that link density is not the only factor but more importantly is the distance or spacing between links both vertically and horizontally.

6.3 Research Limitations

This research provide a specific hardware implementation that though most likely to be used there are numerous other configurations and technologies available. The user-testing group was small and only one user test was performed on the gesture-enabled interfaces.

6.4 Future of Gesture Technology

The future of non-touch gesture interfaces will continue to grow over the next few years but it ultimately a transitional technology as it will be replaced in the long term future by optical retinal readers and gestures will not be necessary.

Presently, the technology is already advancing. Leap Motion is a consumer product that is coming onto the market in summer of 2013, which provides motion and depth detection that is 1000 times more precise than the tested and widely adopted Microsoft Kinect. This device focused on tracking small hand and finger movements rather than tracking full body movements. Because of the more precise detection which intern allows for smaller body movements this technology will make adoption easier in home situation. Also, Leap motion is working to make the motion sensing technology built-in to major laptops just as web video cameras are today. On another device front, Microsoft has announced a web browser for the Xbox platform that would allow for navigation using gestures. Once released this would provide access to over 100 million users with access to a gesture enabled web browser.

At the same time, there are examples of new motion or gesture sensing interface prototypes coming out from large companies such as Asus as well as individual innovators working on their garage. The version from Asus is similar to Microsoft Kinect and though untested, would in theory work with the same software presented in this project.

Long term, gesture interfaces could be surpassed with the growing development of eye-tracking software. This technology is currently used for security applications and for user behavior research studies but could soon be the fastest and easiest way to interact with a computer so no gestures or hand movement would be required.

6.5 Prospectus of Further Research

The implementation of the hardware, software and interfaces provided viable results in testing no-touch gesture interfaces but there is room for further development in hardware, software, better user interfaces and further testing.

The DepthJS plugin could also use further development. Currently, it is difficult to install and the documentation is limited. In order for this type of computer interaction to become widely adopted, the technology must become easier to install and use than the current situation.

It would also be interesting to expand the user testing to include testing the content presented to see if users prefer a specific medium or type of content for this type of interface. Working with content publishers to design and develop an application for non-touch gesture interfaces would also allow for the testing of various types of content in the system with a much larger testing group.

Leap Motion is a consumer product that is coming onto the market in summer of 2013, which provides motion and depth detection that is 1000 times more precise than the tested and widely adopted Microsoft Kinect. This device focused on tracking small hand and finger movements rather than tracking full body movements. The next logical step in non-touch gesture research is to test this new tracking technology.

One additional development in gesture tracking technology that would benefit application developers would be the ability to create and detect custom gestures. This would allow for scripted interactions such as playing two minutes of news video while the user brushes their teeth, which would truly integrate the information and technology into the daily routine.

6.6 Contribution

This project contributes to the computer science industry as well as the journalism and social media industries by providing multiple usable, public and open-source non-touch gesture interfaces, which can be used as a starting point for further development. Code has been contributed and merged with the open-source DepthJS project. All code is available to public through Github.

In addition to the code base, a series of best practices and a summary of insights and observations will help future development of gesture applications. As gesture sensor technology progresses, these insights should enable a more rapid release of new gesture interfaces by social and news publishers.

The application and findings has already been presented to researchers, journalists and developers at the Journalism Interactive Conference hosted by the University of Florida and to faculty researchers of the School of Journalism and Mass Communications and the School of Information Sciences at the University of North Carolina at Chapel Hill.

6.7 Conclusion

Gesture based, non-touch user interfaces are becoming more common in console gaming and other specialized industries such as health care but has yet to become a common and useful way to navigate the Internet, specifically news and social media sites. This project successfully answered the question of, “Can gesture-based interfaces using motion tracking sensor technology be used to consume news and engage with social and current events content?”

The use of non-touch gesture interfaces is a viable option for news and social news interaction but effort and technical knowledge required to implement the technology is a barrier to entry. The data showed that users did enjoyed the experience and with further advancements in the detection technology as well as custom designed and developed user experiences, gesture interfaces could become anther way users get information on a regular basis both in public and at home.

As noted in previous chapters the software is difficult for the average consumer to install and was inconsistent when interacting with the Kinect device. These barriers to entry for using gestures to interface with the browser are challenging but if the software moved from an open-source project that is rarely updated to a product with a focused developer team those hurtles could be resolved.

It is apparent that publishers pursuing the non-touch gesture user should create and deliver a custom user experience for this audience. Using the same interface that was designed for the desktop or a mobile device will not provide for a positive or even acceptable user experience. Responsive design methods are helpful for the varying screen sizes but there are many more design and interaction options that need to be considered for this specialized user experience.

As was seen in recent history with the evolution of mobile presentation on the web, as the audience grows so will the need for this type of interface and publishers must create custom gesture experiences if they want to be competitive in this type of experience. These custom interfaces, as seen in the user testing will provide specific and visual content to a new and growing audience in a way this is a positive and enjoyable experience for the user.

In conclusion non-touch gesture interfaces are a viable interface of news and social information. This new interface and implementation has the the potential to become a widely adopted way users consume news and social information but for this technology to be successful it requires further development in the software as well as custom designed user experiences from content providers.

REFRENCES CITED

Aarts, M. (2012), Using Kinect to interact with presentation software. Uppsala: Malardalens hogskola. Available at: http://www.idt.mdh.se/utbildning/exjobb/files/TR1288.pdf [Accessed 17 September 2012].

Anderson, F. and Biscof, W. (2013) Learning and performance with gesture guides. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, p.1109-1118.

Annett, M. and Biscof, W. (2013) Your left hand can do it too!: investigating intermanual, symmetric gesture transfer on touchscreens. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, p.1109-1118.

App-side (2012), Top Five Things to take into account when considering the gesture controlled user experience. Available at: http://www.app-side.com/2012/03/top-five-things-to-take-into-account-when-considering-the-gesture-controlled-user-experience/ [Accessed 17 September 2012].

Bhuiyan, M, & Picking, R 2011, 'A Gesture Controlled User Interface for Inclusive Design and Evaluative Study of Its Usability', Journal Of Software Engineering & Applications, 4, 9, pp. 513-521, Computers & Applied Sciences Complete, EBSCOhost, [Accessed 30 April 2013].

Boulos, M., Blanchard, B. & Walker, C. (2011), Web GIS in practice X: a Microsoft Kinect natural user interface for Google Earth navigation. International Journal of Health Geographics, 10(45).

Gray, T. (2001), The challenges of multi-touch gesture interfaces | Embedded. Available at: http://www.embedded.com/design/embedded/4231144/The-challenges-of-multi-touch-gesture-interfaces?pageNumber=0 [Accessed 17 September 2012].

Hummels, C, Overbeeke, K, & Klooster, S 2007, 'Move to get moved: a search for methods, tools and knowledge to design for expressive and rich movement-based interaction',Personal

& Ubiquitous Computing, 11, 8, pp. 677-690, Computers & Applied Sciences Complete, EBSCOhost, [Accessed 30 April 2013].

Izkovitch, A. (2012), Designing for Context on Multiple Devices. Available at: http://mashable.com/2012/07/07/designing-for-multiple-devices [Accessed September 17. 2012].

Juhlin, O. and Onnevall, E. (2013) On the relation of ordinary gestures to TV screens: general lessons for the design of collaborative interactive techniques. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, p.919-930.

Krug, S. (2010) *Rocket surgery made easy*. Berkeley, Calif.: New Riders.

Lai, K. (2012), A gesture-driven computer interface using Kinect. In Image Analysis and Interpretation (SSIAI),. 2012 IEEE Southwest Symposium on Image Analysis. Santa Fe, NM: IEEE, pp. 185 –188.

Leitner, J. and Perteneder, F, et al. (2013) Kolibri: tiny and fast gestures for large pen-based surfaces. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, p.919-930.

Long, A., Landay, J., & Rowe, L. (1999), Implications for a gesture design tool. In Proceedings of the SIGCHI conference on Human factors. CHI 99. Pittsburgh, Pennsylvania, USA: ACM.

Madrigal, A. (2012), iPhone 5? Yawn. What Will the “Phone” of 2022 Look Like? - Alexis C. Madrigal - The Atlantic. TheAtlantic.com. Available at: http://www.theatlantic.com/technology/archive/12/09/what-will-the-phone-of-2022-look-like/262300/ [Accessed 17 September 2012].

Marco, R, Gustavo, M, & Angelo, S. (2010) 'Playing into the wild: A gesture-based interface for gaming in public spaces', Journal Of Visual Communication And Image Representation, 23, pp. 426-440, ScienceDirect, EBSCOhost [Accessed 30 April 2013].

Neilson, J. (2000) *Why You Only Need to Test with 5 Users*. [online] Available at: http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ [Accessed: 7 May 2013].

O'hagan, R. (1997), Finger track — A robust and real-time gesture interface. Advanced Topics in Artificial Intelligence, 1342, 475-484.

Pastore, S. (2012). The Role of Open Web Standards for Website Development Adhering to the One Web Vision. International Journal of Engineering and Technology, [Online]. 2, 1823-1834. Available at:http://iet-journals.org/archive/2012/nov_vol_2_no_11/5551311348229481.pdf [Accessed 14 April 2013].

Rosenberg, A. (2012), Microsoft Patent Takes Aim At Holodeck Gaming - G4tv.com. Available at: http://www.g4tv.com/thefeed/blog/post/727820/microsoft-patent-takes-aim-at-holodeck-gaming [Accessed 17 September 2012].

Saffer, D. (2009), Designing Gestural Interfaces: Touchscreens and Interactive Devices. 2nd ed., Sebastopol: O'Relly Media.

Schramm, M. (2010). Kinect: The company behind the tech explains how it works | Joystiq. Available at: http://www.joystiq.com/2010/06/19/kinect-how-it-works-from-the-company-behind-the-tech/ [Accessed 17 September 2012].

The Kinect Effect (2013), TED | Ads Worth Spreading | The Kinect Effect. [ONLINE] Available at:http://www.ted.com/initiatives/aws/the_kinect_effect.html. [Accessed 15 April 2013].

Kessler, S. (2013), Microsoft Kinect Sales Top 10 Million, Set New Guinness World Record. [ONLINE] Available at: http://mashable.com/2011/03/09/kinect-10-million/. [Accessed 04 March 2013].

Unander-Scharin, C. and Höök, K., et al. (2013) The throat III: disforming operatic voices through a novel interactive instrument. *CHI EA '13 CHI '13 Extended Abstracts on Human Factors in Computing Systems*, p.3007-3010.

APPENDICES

Appendix A. User Evaluation Script

Adapted from Rocket Surgery Made Easy by Steve Krug

http://sensible.com/downloads-rsme.html

Hi, ___________. My name is ___________, and I'm going to be walking you through this session today.

Before we begin, I have some information for you, and I'm going to read it to make sure that I cover everything.

You probably already have a good idea of why we asked you here, but let me go over it again briefly. We're asking people to try using a new way of interacting with websites that we're working on so we can see whether it works as intended. The session should take about an hour.

The first thing I want to make clear right away is that we're testing the *technology*, not you. You can't do anything wrong here. In fact, this is probably the one place today where you don't have to worry about making mistakes.

As you go through the tasks, I will sometimes ask you to try to think out loud: to say what you're looking at, what you're trying to do, and what you're thinking. This will be a big help to us.

Also, please don't worry that you're going to hurt our feelings. We're doing this to improve the technology, so we need to hear your honest reactions.

If you have any questions as we go along, just ask them. I may not be able to answer them right away, since we're interested in how people do when they don't have someone sitting next to them to help. But if you still have any questions when we're done I'll try to answer them then. And if you need to take a break at any point, just let me know.

You may have noticed the microphone and the cameras. With your permission, we're going to record what happens on the screen, your actions and our conversation. The recording will be used to help us figure out how to improve the technology and demonstrate the different actions. And it helps me, because I don't have to take as many notes.

If you would, I'm going to ask you to sign a simple permission form for us. It just says that we have your permission to record you, and the recording could be published along with the written analysis.

Give them a recording permission form and a pen While they sign it, START the SCREEN RECORDER

First, I would like to show you how to use the gesture-based interface.

[Demonstrate: Wave, Click, Swipe Left, Swipe Right, Scroll Up and Scroll Down]

You can practice to get used to the motions.

[Give a few minutes to allow user to feel comfortable]

Now I'm going to ask you to try doing some specific tasks.

TASKS:

On New York Times

- Please find and view the main news story about [ENTER NEWS TOPIC OF TOP STORY FOR THE DAY]. Then return to Homepage.
- Go to the Sports Section and then return to Homepage.
- View a video at the bottom of the page.

Repeat Steps using mobile NYTimes

On Facebook,

- Please select any photo in the news feed to view. Return to the news feed.
- Like any item in your news feed.
- View your Facebook Messages (email-like page)

Repeat steps using Mobile Facebook

On Gesture News

- Please find and view the top news story about [ENTER NEWS TOPIC OF TOP STORY FOR THE DAY]. Then return to Homepage.
- Go to the Sports section. Return to Homepage.
- View the Top Headlines Video

On Photo Slider

- Please select view and read the first story.
- View other stories
- Go back to a previous story.

Thanks, that was very helpful. Now, I need you to please complete this survey. Thank you for your help and your time.

Appendix B. User Survey

The users tested were asked to complete a survey about the experience. The questions are listed below as well as an image of how the survey was presented in the web-based form.

1. For each website below, on a scale of 1 to 5 with five being the easiest and one being the worst, for each website easiest to complete the tasks
    a. NY Times
    b. NY Times Mobile
    c. Facebook
    d. Facebook Mobile
    e. GNews

-

2. For The New York Times, which was easier to complete the tasks?
    a. Desktop Site
    b. Mobile Site

-

3. For Facebook, which was easier to complete the tasks?
    c. Desktop Site
    d. Mobile Site

-

4. If you were using this gesture interface to view websites, should publishers such as NYTimes and Facebook provide a custom designed interface for gesture users or is the main or regular site acceptable?

-

5. What was the easiest part of navigating all of the sites?

6. What was the most difficult part of navigating all of the sites?

-

7. Can you see yourself using this type of gesture interface to consume news or social information?

8. Why or why not would you use this type of gesture interface to consume news or social information?

Appendix C. Evaluation Matrix

This chart below was used by the researcher to record the time to complete a task and if the task was completed.

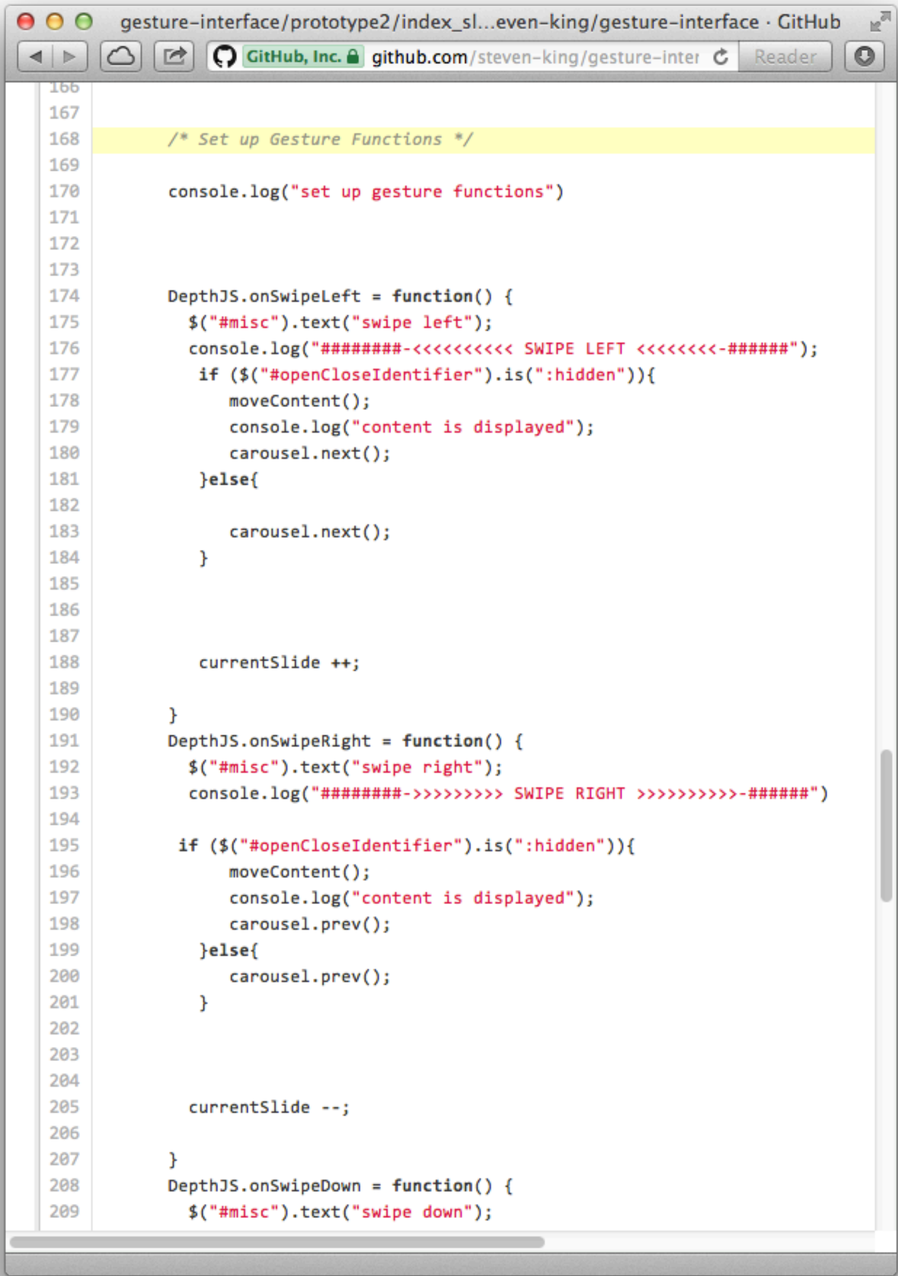
Task ID	Task	Result (Pass/Fail)	Time to Complete
NYMain1	• Find and view the top story (most important article) on the site.		
NYMain 2	Go to the Sports Section.		
NYMain 3	• Scroll down and view a video.		
NYMobile1	• Find and view the top story (most important article) on the site.		
NYMobile 2	Go to the Sports Section.		
NYMobile3	• Scroll down and view a video.		

FBMain1	Select any photo in your news feed to view		
FBMain2	Like any item in your news feed		
FBMain3	• View your Facebook Messages (email-like page)		
FBMobile1	Select any photo in your news feed to view		
FBMobile2	Like any item in your news feed		
FBMobile3	• View your Facebook Messages (email-like page)		
GN1	Find and view the top story (most important article) on the site.		
GN2	Go to the Sports section.		
GN3	• View the Top Headlines Video		
IM1	• View and Read First Story		
IM2	• View other stories		

IM3	• Look at a previous photo		

Appendix D. CODE SAMPLE DEPTHJS FUNCTIONS

The code sample presented below is an excerpt of the JavaScrip that implements the DepthJS for the swipe-driven interation of the large photo interface. Full code can be found at: https://github.com/steven-king/gesture-interface




```
      /* Set up Gesture Functions */

      console.log("set up gesture functions")



      DepthJS.onSwipeLeft = function() {
        $("#misc").text("swipe left");
        console.log("########-<<<<<<<<<< SWIPE LEFT <<<<<<<<-######");
         if ($("#openCloseIdentifier").is(":hidden")){
            moveContent();
            console.log("content is displayed");
            carousel.next();
         }else{

            carousel.next();
         }



         currentSlide ++;

      }
      DepthJS.onSwipeRight = function() {
        $("#misc").text("swipe right");
        console.log("########->>>>>>>>> SWIPE RIGHT >>>>>>>>>>-######")

       if ($("#openCloseIdentifier").is(":hidden")){
            moveContent();
            console.log("content is displayed");
            carousel.prev();
         }else{
            carousel.prev();
         }



        currentSlide --;

      }
      DepthJS.onSwipeDown = function() {
        $("#misc").text("swipe down");
```

gesture-interface/prototype2/index_sl...even-king/gesture-interface · GitHub

GitHub, Inc. github.com/steven-king/gesture-inter Reader

```
    }
    DepthJS.onSwipeDown = function() {
      $("#misc").text("swipe down");
    }
    DepthJS.onSwipeUp = function() {
      $("#misc").text("swipe up");
    }

    DepthJS.onPush = function() {
      console.log("########->>>>>>>>> PUSH >>>>>>>>>>-######");
      moveContent();



    }

    DepthJS.onRegister = function(){
      console.log("Registered");
      if (currentState == "white"){
           $("#directions").toggleClass('blue');
           currentState = "blue";
      }

      $("#directions").hide('slow', function(){

      });
    }

    DepthJS.onUnregister = function(){
      console.log("UN Registered");
      if (currentState == "blue"){
           $("#directions").toggleClass('blue');
           //$("#directions").toggleClass('white');
           currentState = "white";
      }
      if (firstTime == false){
      directionText= "<h3>Raise your hand.</h3>"
  }
  $("#directionText").html(directionText);



      //$("#directions").toggleClass('blue');
      $("#directions").show('slow');
```