Gesture Elicitation for Image Editing

Andreea Bancila

This thesis was prepared under the guidance of Professor Christopher Andrews

Presented to the faculty of Mount Holyoke College in partial fulfillment

of the requirements for the degree of Bachelor of Arts with Honors

Department of Computer Science South Hadley, Massachusetts

May 6, 2013

ABSTRACT

Natural User Interfaces (NUIs) represent one of the most important steps that computer scientists have taken towards shaping a more intuitive, more detached kind of interaction with the technological sphere. NUIs, which refer to mouse-less interfaces, are advancing swiftly and seeking to replace the graphical user interfaces (GUIs) based on windows, icons, menus and a pointer (WIMP). The term natural is related to the way people interact with the physical world, as it is those experiences that the NUIs are trying to mimic.

In this work we focused on gesture based interaction which poses a greater interest than for example touch applications since there is a lot of uncharted territory. Gestures fit into the context of daily life fairly well, but they are not discoverable in the way that menu options are which could lead to user confusion or lack of awareness of the features of the tool.

One way to address this problem is to attempt to discover gestures that are actually intuitive by examining which gestures users initially try when asked to perform certain tasks. This research project focused on image editing as a task, in an attempt to accommodate the ever expanding community of people generating pictures. Instead of providing the users with a package of previously created editing gestures, we asked people to provide us with a collection of gestures that they could spontaneously and instinctively think of in the context of image editing.

Our main goal was to discover whether there were any common gestures that people shared for specific image editing tasks. In many cases users created similar motions, which validated the approach and provided a starting point for building a gesture-based image editor prototype (which could be expanded to many other gesture-based tools).

ACKNOWLEDGEMENT

I would like to thank my thesis advisor, Professor Christopher Andrews, who guided me through the whole process of conducting my research and writing my paper. During the past couple of years you have become not only my mentor but also my friend and I learned so much from you.

❖

I also want to thank Professor Audrey St. John, who through her spirited and wonderful teaching style made me want to pursue Computer Science.

❖

I want to thank Professor Tatiana Ginsberg, who during my sophomore year of college introduced me to digital art, something I became captivated with ever since.

❖

I want to thank all the professors in the Computer Science department for their ceaseless support and kind words of advice and encouragement.

❖

I also want to thank my family for their continuous care and guidance.

❖

Lastly, I thank all my friends, especially Ines and Nathan, for being my greatest supporters and always reminding me to have fun throughout this process.

TABLE OF CONTENTS

TABLE OF FIGURES

Chapter 1

INTRODUCTION

In 1963 Douglas Engelbart, a pioneer of graphical user interfaces, was designing the first computer mouse at the Stanford Research Institute. Fifty years later, Engelbart's square, wooden mouse is mentioned on the front page of reddit.com, a prevalent social news and entertainment website. Millions of people read the article and see pictures of the great invention which becomes once again infectiously popular, yet for very different reasons. For some the images attest to computing progress, while for others they depict an "antique" curiosity, an impractical object which was a major stepping stone decades ago, but would never render itself useful to our growing hunger of technological power.

Over the years the context of computing has changed exponentially. From Von Neumann's 1945 EDVAC, one of the earliest electronic computers designed to execute basic arithmetic operations, to IBM's 2009 Roadrunner, a supercomputer capable of performing 1000 trillion operations per second, technology has fundamentally changed its core. However, it is not just its mechanics that has developed, but also the way we interact with technology. The reason people were puzzled and intrigued after seeing Engelbart's computer mouse in modern day context is because of the unfamiliarity factor. The unnatural, square shape surpasses our first intuition of how we might interact with it. The contemporary mouse fits in the palm of our hand seamlessly, basically dictating the way it should be held. It requires little to no instructions on how it should

be operated. Its use seems natural, instinctive, so much so that you could even forget you are working with it. This exact concept – that computers should blend into the background, become an invisible means to an end, something almost if not completely hidden from people's awareness – was first described in 1988 by Mark Weiser as the dream of ubiquitous computing.

The dream of ubiquitous computing is shaping into reality. The direction technology is headed is one that seeks to effortlessly link machinery and society. People want to be able to perform tasks fast, reliably, everywhere and at any time.

1.1 Natural User Interfaces

A feature commonly associated with ubiquitous computing is the use of Natural User Interfaces (NUIs). They represent one of the most significant steps that computer scientists have taken towards shaping a more intuitive, more detached kind of interaction with the technological sphere. These mouse-less interfaces are advancing swiftly and replacing the graphical user interfaces (GUIs) based on windows, icons, menus and a pointer (WIMP). The term natural is related to the way people interact with the physical world, as it is those experiences that the NUIs are trying to mimic. The typical approaches that are labeled "natural" include interactions such as pen, touch, gesture, speech and sketch [1].

With the proliferation of devices such as the iPhone, iPad and Kindle touch has become very popular. These gadgets provide multi-touch displays and distinctive software that allow users to manipulate objects on the screen directly through touch. While implementing touch as the sole means of interaction has worked very well for some applications, combining NUIs could lead to

additional functionality. Hinckley et al. proposes a combination of both pen and touch arguing that “one device cannot suit all needs and that some are inherently more natural than other for certain tasks” (e.g. using a pen for drawing or tackling delicate, small tasks rather than fingers) [2].

Speech recognition technology is also developing rapidly leveraging people’s ability to simply and efficiently communicate with the device through their voice. Apple’s Siri [3] which uses Nuance Dragon [4] is a device which has raised awareness of this kind of technology. The user can ask Siri an endless list of questions and expect sometimes precise, sometimes humorous answers.

Gestural interaction is gaining momentum as well. Gesture recognition is used to interpret human gestures through mathematical algorithms, thus allowing a human computer interaction based primarily if not uniquely on bodily motion. The Nintendo Wii followed by the Microsoft Kinect has started bringing gestural interfaces out of the realm of science fiction and into the home. With the release of the Kinect a multitude of games, educational projects, applications and music were developed using our natural, familiar interactions in physical space. Multiple sensors are used to capture almost every exchange between user and device in order to create a highly responsive, powerful system (Figure 1). Another example is the GE GEstures project at Frankfurt airport, where travelers can become informed about wind turbines by moving hands in circular patterns in front of a large screen displaying the turbines’ noses [5]. Such projects offer a glimpse into how enjoyable, captivating and rather intuitive if designed well these gestural applications can be.

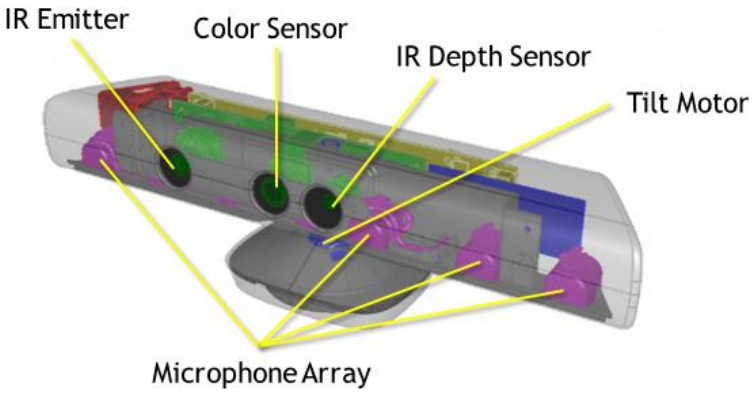


Figure 1: The Kinect device and its sensors

1.2 Project idea

Having this new realm of human-computer interaction then raises the question of what kind of things we could achieve by applying it. The airport informational project, the Wii and Kinect applications are just a few of the areas in which NUIs prove to be suitable. However, gestural interfaces can sometimes prove to be hard to navigate. Intrinsically, gestural interfaces are not discoverable in the way that menu options are, which could lead to user confusion or lack of awareness of the features of the tool.

One way to address this problem is to attempt to discover gestures that are actually intuitive by examining which gestures users initially try when faced with performing certain tasks. Instead of providing users with a package of previously created gestures, one could instead elicit them from

users, asking them to provide gestures and seeing what they spontaneously and instinctively think of. Such gestures that naturally occur to the users could be more intuitive to perform and easier to remember. The main question we wanted to be able to answer was whether there would be common, fluent gestures within the array of motions created by our users.

The goal of our research is to allow the users to feel more connected with the task rather than the technology used to achieve that task and we think this could be possible for two main reasons. Firstly, gesturing (as opposed to using a mouse or a touch screen) is something that everyone does and secondly if we are able to identify a collection of common gestures through gesture elicitation, then these would be an assortment of popular, frequently used motions that users could more easily think of.

For our research we decided to focus on image editing and started shaping an idea of working with natural user interfaces as a way to explore image editing within the context of wide displays. This would allow people to perform quick editing tasks while not sitting in front of a computer and also, if desired, share their process and photographs with everyone around. However, is this approach of editing pictures through gestures fast and reliable? Aren't mobile applications such as Instagram or PhotoFunia the fastest approach since the pictures are already residing on the device? With the third generation of Apple TVs supporting iCloud (a cloud storage and cloud computing service launched by Apple Inc. on October 2011), the images on the mobile phone are part of the Apple TV as well. This means that every time a picture is taken with the phone device, it becomes part of a Photo Stream that is also viewable when using the television set. Therefore, the task of transferring data from one device to another becomes trivial.

Viewing and editing images on something other than the tool that was used to capture them is consequently just as swift and consistent.

1.3 Thesis structure

The following chapters not only describe the approach we pursued in order to test the viability of gesture elicitation but also provide the results accompanied by systematic discussion. Chapter 2 encompasses some related work within the field of human computer interaction and natural user interfaces. Chapter 3 provides an in depth description of the project and of the set-up of the testing environment. It also describes how the main study was conducted. Chapters 4 and 5 analyze the results, interpret their significance and discuss how they could be implemented. Lastly, Chapter 6 represents the concluding chapter of this thesis and provides thoughts on future directions this project could take.

Chapter 2

LITERATURE REVIEW

As natural user interfaces are gaining momentum, an increasing number of studies are starting to place this direct, instinctive type of interaction at the core of their research. From research on what this natural interaction really means and how it could be implemented, to examples of studies and applications that focus on gestural interfaces, this chapter provides a glimpse of the work that has been conducted in the field of natural human computer interaction.

Pike et al. define the science of interaction as the study of methods by which people generate knowledge through the manipulation of an interface [6]. However, the authors argue that while some of the interaction occurs within the context of a software tool, most of it arises internally, in one's mind. Therefore, they are of the opinion that a more natural and intuitive human-technology interaction is needed, and that in order to achieve that more interaction research is required.

The authors suggest that even though gestural interfaces are a sub-group of natural interfaces, it does not necessarily mean that "natural" gestures are readily provided. This is something that we explored in our research as well. Since gestures are not as discoverable and self-explanatory as for example icons, or the keys on a keyboard, there has to be another way for users to know

which gestures they are expected to use when interacting with a gestural application (hence us exploring gestural elicitation).

One step towards attaining it would be to understand the connection between interaction and inquiry in order for coherent, reliable analysis capabilities to be available for the users whenever they are thinking about a problem space. The four authors agree on the fact that interaction should not be an afterthought, but rather the primary thing that is considered when developing an analysis system. Interaction should not be just a set of controls that allow the user to modify a display or a program.

As a closing thought, Pike, Stasko, Chand and O'Connell acknowledge the fact that visual representations can be informative without interactions (e.g. static information graphics). However, they mention that interaction is considered to be responsible for engaging and keeping the user in a ceaseless 'cognitive flow'. Consequently it is important to continue work in this research area. Ubiquitous interaction, capturing user intentionality, interaction evaluation, principles of design and perception, studying all these core elements will help shape the way people will engage with information spaces in the future.

In *Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions* [7], the authors discuss the fact that interaction technology has been gaining a lot of momentum. They highlight the proliferation of touch enabled phones and multi touch slates which signifies that people are becoming more and more interested in mouse-less interfaces. Despite the recognition of the significance of interaction visualization, comparatively

little has been done within the interaction design community and graphical user interfaces (WIMP GUIs) are still the most common models to design InfoVis interfaces. Some of the drawbacks of using WIMP GUIs include options that need to be found in multi-step hierarchical menus or amongst a multitude of buttons, checkboxes and other widgets; the mapping of multi-dimensional data tasks to 2D widgets is not particularly natural; not the most appropriate interface for multiple display or analysis environment.

One very important fact outlined in this paper is that regardless of how simple and easy to use an interface is, it will still enforce a layer of cognitive processing for people. Researchers have been working on minimizing the cognitive distance between intent and the execution of the intent and on the effort required when interacting with an interface. The goal is to have people focus on the task they are performing and not the interface for specifying the task. This idea is something we are trying to capture with our own gestural image editing research, by looking at gestures that are most frequently used and implementing those with the image editor.

Lee, Isenberg, Riche and Carpendale propose a few topics that could prove to be interesting research themes: further exploration of how new means of input can enhance the data analysis experience by leveraging people's kinesthetic memory of data; or investigate how people behave when using new technologies and how they control theoretically increased freedom of expression to pursue their interaction objectives

In a research article by Francese, Passero and Tortora the emphasis is placed on 3D gestural user interaction: the Nintendo Wii and the Microsoft Kinect devices. [8] The authors recognize the

potential of these devices which are capable of enriching, through low cost motion capture, the user interaction with desktop computers by constructing new forms of natural interface. One of the first listed traits of natural interfaces is one pertaining to gestural interfaces: "Easy to learn: Simple and intuitive …" Looking at this research paper was meaningful to our study, since we were interested in seeing how users responded to interacting with these "simple, intuitive" 3D environments.

The three scientists point to the recent market trend that has changed in order to accommodate the ever evolving customer preferences. The emphasis has been placed on realistic human-computer interfaces rather than on computing performance or on graphical capabilities. The Nintendo Wii console perfectly demonstrates this preference switch, proposing a gaming platform not particularly thrilling in terms of performance, yet offering the users an enhanced game experience with active gestures and very effective playing metaphors. Breaking several records as the best sold console, the Wii device definitely portrays the effect of the associated innovative gestural interfaces on user satisfaction. Following Nintendo, Sony and Microsoft, the other two competitors in the game console segment, presented their motion detection game controllers: the PlayStation Move (2008) and the Kinect (2010). Both the Wiimote created by Nintendo and the PlayStation Move produced by Sony, are very similar in their design, requiring the user to operate a controller in order for their gestures to be captured. The Kinect however is the first consumer full body motion capture device relying on an Infra-Red emitter and two video cameras.

Francese, Passero and Tortora conducted a short study via standard questionnaires in order to better understand consumers' preferences. The results pointed to a satisfaction enhancement related to the Kinect device, which suggests that the more natural and encompassing the body in action an interface proves to be, the more the user is satisfied and involved in the 3D experience. The conclusion and general advice is to avoid when possible the traditional window, icon, mouse interaction and experiment with new gestures and original forms of physical commands.

Another article that focuses on gestures as a way of interacting with the technological sphere is *Charade: Remote Control of Objects Using Free-Hand Gestures* [9]. In this paper, Baudel and Lafon look at the advantages of using free-hand gestures and at ways in which these gestures could be implemented. The authors begin by listing three reasons free-hand gestures are significantly valuable. First, they provide natural interaction – "gestures are a natural form of communication and are easy to learn". Second, they offer concise and powerful interaction – "a single gesture can be used to specify both a command and its parameters. The position and movements of the hand and fingers provide the potential for higher power of expression". Lastly, free-hand gestures represent a direct kind of interaction – "The hand becomes the input device, eliminating the need for intermediate transducers. The user can interact with the surrounding machinery by simple designation and appropriate gestures".

Baudel and Lafon point out that in spite of all the listed benefits, research laboratories have not been able to create a product encompassing all those features. The main reasons are that gestural communication can result in fatigue, as well as the fact that existing hand gesture input devices require wearing a glove and being connected to a computer, thus decreasing autonomy.

However, tiredness from moving the wrist, fingers, hand and arm can be combated by creating gestural commands that are concise and quick to issue, hence minimizing effort. This is something we definitely discovered during our own gestural image editor study: when asked to perform image editing gestures, most of our participants created very minimal, simple gestures.

An additional interesting idea proposed by the two authors is something they referred to as the "Immersion Syndrome". Generally, systems capture all the motions performed by a user. Therefore, every gesture can be interpreted by the system, whether or not it was intended. As a result when using the gesture recognition device, the user cannot communicate simultaneously with other systems or people. It is consequently important for a gestural device to provide well-defined means to detect the intention of the gestures, a concept we as well struggled with when designing the basis of our gestural image editor.

Moreover, Baudel and Lafon talk about the segmentation of hand gestures, since usually most gestures are continuous by nature. A device that understands and translates gestures, needs to be able to segment the continuous stream of captured motion into distinct" lexical entities", a process that is more or less artificial and always requires approximation. As a result, most systems identify steady positions also known as postures (Microsoft's Kinect is able to recognize postures, something that proved to be very useful during our study) instead of dynamic gestures.

Further investigating how free-hand gestures could be achieved, the two authors and developers created their own prototype entitled Charade. Charade employs hand gestures in order to control

computer aided presentations. A DataGlove that could measure the bendings of each finger and the position and orientation of the hand in 3D space was used.

The two authors conclude their paper by stating that since people are already very skilled in using gestures to communicate, it is important to seize this advantage and use their natural skill with gestures in order to control interfaces and programs.

A Study of Hand Shape Use in Tabeltop Gesture Interaction [10] also looks at ways gestures could be employed in order to accomplish various tasks. The main difference from the studies we previously looked at is that this research specifically focuses on hand gestures (something that we could not use with the Kinect tracking technology which cannot recognize finger movement). However, the way the study was conducted was very relevant to our gesture elicitation study.

The foundations of this article are suggestions and ways in which gestures for tabletop interaction could be designed. The three authors examine both touch screen and computer vision when thinking of ways gestures could be implemented. However, as mentioned before, the most inspiring part of the paper is the way their study was carried in order to determine which types of hand shapes users preferred for different types of tasks executed on a tabeltop.

Epps, Lichman and Wu define the main distinction between computer vision and touch screens. Computer vision is not as direct and robust as a touch screen, but it can track and recognize hand gestures even when the hand is not in contact with a surface. Furthermore computer vision is able to recognize certain gestures and sequences of hand movements (e.g. ‘grab and release’) that would be very difficult to identify with a touch surface.

The three scientists conducted a study trying to ascertain user preferred gestures for a variety of tabletop tasks. Before even starting testing, they anticipated that participants would use and reuse simple hand shapes. Consequently, Epps, Lichman and Wu were interested in observing to which extent this occurred and in correlation to what types of tasks. The study was set up in the following way: "For each task, a static image (a screen shot) occupying the full screen was displayed, and subjects were requested to perform an action on one or more objects displayed in the image. The experimental design has the limitation of not providing feedback to the user, however the objective if the study was to observe user behavior without imposing an arbitrarily defined gesture set upon them. Without knowing a priori what hand shapes and movements a subject will use, it is very difficult to provide adequate feedback". Feedback is something we gave much thought to when designing our study for the gestural image editor since we wanted something that would reassure and help ease the flow of our participants' gestures. In the next chapters we will discuss ways in which we were able to accomplish this, even though we did not know a priori what kinds of gestures our users would perform.

Epps, Lichman and Wu presented 36 tasks (selection of an icon or text, opening of an application, moving of an icon or a slider, scrolling, drawing, cut/copy, rotation of a geometric shape, zooming of a photo and instantiation of a floating menu). In 20 of them, participants were instructed to use one hand only, while in the remaining 16 tasks subjects were instructed to use both hands and sometimes perform an action with one hand and another with the other hand. The study was recorded using a video camera installed above the tabletop and was later analyzed

manually. The three scientists organized each response into a specific category of hand shape/ gesture.

One of the most intriguing results was that some tasks showed little variation in terms of the gestures that were generated for each of them, while others presented a wide range of unique shapes and patterns. Selection, opening, text selection, drawing and slider operations all shared very similar gestures (the index finger was predominantly used). Cutting, copying, rotation, zooming and instantiation of a floating menu exhibited a lot of differences. Epps, Lichman and Wu also looked at how consistent participants were with their own choices for a single type of task. They observed that some participants also tended to modify their strategies as the tasks progressed which revealed the fact that this was the first time they were attempting this kind of interaction.

When thinking of how they could implement the results of the study, the three authors agreed on employing multiple hand shapes in order to allow a range of different types of input to be used. This would mean that the user would have to learn a multitude of gestural shapes. In the end the study suggested the use of both a touch screen and computer vision gesture tracking and recognition devices, especially for applications that require a wide range of commands.

In a very recent article published on January 2013 [11], two scientists looked at how gesture recognition could be introduced in the operating room, where doctors would use gestures to review medical images and records during the surgery. Since stepping away from a patient and using the keyboard and mouse to browse for the desired information might increase the risk of

complications and infections, introducing NUIs in this environment could be a great benefit. The two scientists used the Kinect motion tracking technology and found that the system had 93% accuracy in decoding gestures into specific commands.

This example proves how efficient and helpful gestural interfaces are and also the wide spectrum of possibilities one could use them for. While the example above showcased an interdisciplinary usage of gestural interfaces, the gestures selected for the surgeon reviewing medical images scenario were not chosen through the same process of elicitation described in our study. The gestural tasks were selected by a team of surgeons, but the gestures themselves seem to have been pre-determined by the two scientists. This is not necessarily a drawback, but it does not offer any insight into how similar gestures could be created for diverse scenarios and environments. With our study, we wanted to identify a process through which someone could successfully detect common, intuitive gestures for myriad different circumstances.

Chapter 3

PROJECT DESCRIPTION

The goal of this project is to conduct a study centered on gesture elicitation in an attempt to discover commonalities, similar gestures. Frequently, natural user interfaces are not as intuitive as they were intended to be, since they are based on the design crafted by one person, the programmer, rather than that of the general public. We wanted to conduct a study that could lead to discovering gestural patterns commonly shared amongst groups of people.

Therefore we started designing a testing environment in which participants would be asked to provide their own gestures when faced with the task of editing images. Image editing may not seem like the most obvious application for a gesture based interaction. However, one of the first appeals of choosing it as a foundation is its popularity. Image editing has become very widespread due to the highly increasing number of people who own a variety of smart phones with built in cameras and image editing applications. We wanted to present users with a familiar setting, something they would feel comfortable and confident in using during their creative process.

Firstly, editing describes something we all do all the time. As readers, writers, viewers, consumers of culture we all apply editing in our everyday life. In “Strategies in the Production

and Reception of the Visual" [12], Catherine Soussloff talks about the notion of editing as democratizing the function of aesthetics. What this means is that things are made accessible through editing, we change and cut and paste different fragments until the piece as a whole matches our understanding of how it should be seen or read or heard.

Secondly, images are starting to permeate every sector of our lives. We are constantly taking pictures of everything surrounding us: nature, people, food, fashion, music, politics, chores, projects, holidays etc. Every day and every minute people document and share their most valuable and interesting experiences through pictures. The main reason is the proliferation of cameras – even more precisely, the enormous number of mobile phones with built in cameras. With approximately five billion phone connections worldwide [13] it is no coincidence that people are taking more pictures now than ever.

When merging the two and looking at image editing, we observe this large community of people generating pictures who want to adjust their work according to different visions. While Photoshop, Illustrator or Gimp provide a wide array of editing tools, users are constantly seeking for something that does not require as much time and commitment, especially if this time is idly spent in front of a computer. The growing number of phone based image editors (Instagram, Color Touch or the Facebook camera application) show not only that people want to edit their photographs, but that they want to accomplish it away from the context of the "computer". This brings us back to the ubiquitous computing dream, which we are trying to get closer to through our research on gesture elicitation.

Eliciting gestures for an image editor combines the worlds of gestural interfaces and image manipulation, which at first glance could seem like a very uncommon fusion. Why would image editing be a suitable task for prompting the creation of gestures? Being familiar with the idea of image manipulation, when presented with image editing tasks, participants will have an idea of what they are supposed to do and achieve. Many people already have an understanding of basic image editing tasks. Tasks that are not as popular could be easily explained through the use of illustrations. This is because the tasks are already visual and therefore there is a physical aspect to the transformations that other actual commonplace tasks lack from (e.g. running a spell checker).

Moreover, from a social perspective, photographs are meant to be shared. During various social gatherings displaying images from past events on wide screens such as television sets is preferable to passing around images on mobile phones. The emerging technology is one that allows displaying photographs from the phone on the television, making it easier to see them and to share them. Moreover, instead of relying on a remote control to change the slideshow settings, one could think of replacing the traditional WIMP interface with an interface based on gestures. Imagine being able to not only control the slideshow process from any corner of a room using gestures, but also edit the images as you see them. A gesture based image editor could therefore have many uses, from expanding the possibilities of editing photographs, to offering users the chance to comfortably sit on a couch and edit images at the same time.

3.1 More on Reasons for Selecting Image Editing

If one of the reasons for choosing an image editor as the base for gestures elicitation is its familiarity aspect, a secondary one is the fact that it was easy to show the result of an operation without biasing the process to get there. Some of the studies covered during the literature review chapter ([6], [9], [10]) do a lot of priming with their participants, either showing them videos of how other users performed, or images of what they are expected to accomplish etc. Priming is similar in a way to subliminal advertising, offering various cues on how a task should be performed without hindering or interfering with the actual process.

When thinking of how we were going to make the image editing tasks as clear and concise as possible without offering participants any ideas on how gestures could be performed, we realized that we needed to edit the photographs and then show the users the task and the edited results. Thus, they would be able to see what they are trying to accomplish through their gestures, without us actually demonstrating it for them.

Using image editing for eliciting gestures eased our interactions with the participants during the study. Being able to have such a well-defined and well-known foundation for the project allowed us to focus more on the gestures that we were receiving rather than on us trying to explain the users how to generate these gestures.

3.2 The Image Editing Tasks

Taking into consideration the popularity of image editing and thinking about which editing tasks users could most benefit from in order to quickly yet creatively edit their photographs, we selected ten editing tasks. The ten editing tasks that users had to act out through gestures were chosen in such a way that they would be simplistic yet sufficient for a well-rounded image editor. The tasks included: rotating an image to the right, zooming in and out, cropping two images (cropping a landscape and a portrait), navigating through two differently shaped menus (one was a list, the other one a grid), increasing and decreasing the effects of a filter that was already applied and making an image half of its original size. In this section, we will examine each of these tasks in turn and describe why they were selected.

3.2.1 Task 1: Rotating an image

Rotating an image to the right or left, while not as common as other image editing tasks, has still been widespread in the world of NUI editors. Looking at touch applications, which are currently some of the most popular natural interfaces, one can see how rotating and various other editing and navigation tasks have been implemented. The main reason for drawing comparisons between the world of touch and gestural interfaces and somewhat bridging the two is the fact that people are already fairly accustomed with using the natural interfaces of touch related applications. One question we wanted to explore was whether users tried to apply their experience with touch interfaces to the world of gestures.

A memorable year for the world of touch applications was 2007 when Steve Jobs was reforming the world of smart phones with the introduction of the iPhone. During his official presentation of the Apple product he stated “we also have the coolest photo management app ever, certainly on a mobile device, but I think maybe ever […]”. [14] The photo management application on the 2007 iPhone was both a viewer and an image editor and some of the features that it included were its very responsive touch interface and the re-introduction of the “pinch” and “swipe” as a way to edit pictures. While other projects (the 1991 Digital Desk [15], or the 2001 Diamond Touch [16]) had previously employed similar touch interactions, the “pinch” and “swipe” became widely known and an embodiment of the smart phone after Steve Jobs’ product launch.

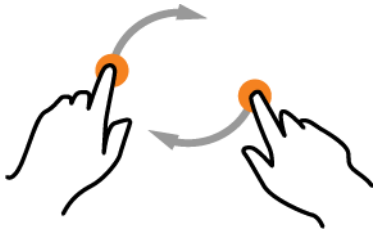


Figure 2: Rotating an image on a touch application [17]

The circular swipes as depicted in Figure 2 represents a motion path that touch editors have been consistently using since 2007 in order to describe the rotation of an image. Therefore when introducing the rotating of a picture as a task for our gestural image editor, we wanted to include a feature that users had be accustomed to seeing in a NUI image editor. In addition, we were also

interested in looking at how participants would transition from the well-known swipe touch motion to a greater, possibly more expanded and expressive gesture.

Even though we were anticipating that some of the participants in our study would be familiar with the concept of rotating an image, we wanted to be sure that the task was clear to all users. Therefore, as mentioned in part 3.1 of this chapter, before participants were prompted to generate gestures, they were shown how the editing task would affect a given image. Figure 3 represents the picture that participants saw right before they were prompted to create a gesture for the rotation to the right of an image. We selected a fairly simple stock photograph, depicting a field topped by a gray skyline, since we did not want the portrayed landscape to derail our users' focus from the actual task. Next to it we added an image that showcased the rotating process. This is when the visual aspect of the image editing tasks extremely helped us in conveying the effects of the task using no words, no captions, but just a picture.

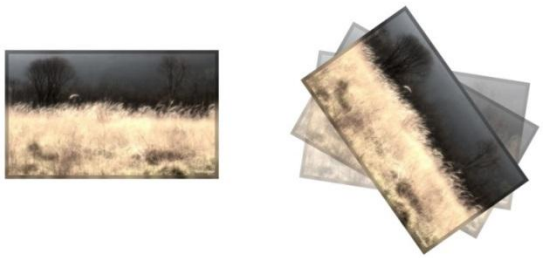


Figure 3: How to rotate an image [18]

3.2.2 Tasks 2 and 3: Zooming in and out

Zooming in and out of a picture are perhaps two of the most popular image navigation tasks, offering the possibility of seeing even the most minute details in a photograph. They are a navigation tool provided in all image editors and in almost all cases they have a representative shortcut for ease of access. With NUI editors, the "pinch" has become a trivial touch motion that describes the process of zooming in and out of a picture.

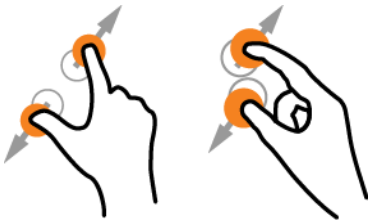


Figure 4: Zooming in and out of an image on a touch application [19]

When including zooming in and out as tasks for the gestural image editor, we were again taking advantage of the familiarity aspect. We anticipated the majority of participants being very comfortable with the idea zooming in and out of a photograph, but we were once again curious to discover if the gestures would be similar in any way to the famous "pinch" touch motion (Figure 4). However for this task, as for all other tasks, we introduced pre-slides which depicted the way the editing effects would alter a given image. In this way we were confident that all users had a good understanding of the set-up as well as of the tasks they were supposed to accomplish. Once

again, we used the same landscape image to exemplify zooming in and zooming out of an image (Figure 5 and Figure 6).

Figure 5: How to zoom in [18]

Figure 6: How to zoom out [18]

3.2.3 Tasks 4 and 5: Cropping a landscape and a portrait

Cropping represents another very common feature, present in almost all image editors. Within NUI editors, such as Lightbox, Photoshop Express, Instagram or FX Photo Studio, cropping has been predominantly denoted by a selection box delimiting the area of the photograph that is to be preserved, and a "pinch" touch motion used to enlarge or shrink the size of the selection box (Figure 7).

Figure 7: Using the "pinch" to outline the selection box [20]

This is an example of a more complex image editing task that in the existing touch based editors requires two different concepts, the selection box and the “pinch” motion. When asked to create a gesture for cropping, our participants were nevertheless faced with a challenging task, since they had to think of ways to incorporate the idea of a selection box or create a whole new concept for cropping.

We asked users to perform two different cropping actions: cropping of a landscape and of a portrait. The cropping of a portrait task was a more conventional type of cropping which involved selecting a small region out of the photograph. The portrait that was to be cropped (Figure 9) depicted a young woman surrounded by wild nature and the task was to crop the woman’s silhouette out. This was a more precise, refined type of crop.

The landscape cropping task asked to crop an image portraying a golden field in its lower half and a sky in its upper half (Figure 8). The task was to crop the field out of the picture, therefore

asking the user for a more elongated, larger crop, something that could prove to be very different for the users since the task was not really bounding box based.

We included both cropping of a landscape and of a portrait as tasks since we were trying to determine if cropping finer details using gestures would prove to be any different from cropping greater areas of the photograph.

Figure 8: How to crop a landscape [18]

Figure 9: How to crop a portrait [21]

3.2.4 Tasks 6 and 7: Navigating through a list and grid menu

Navigating through a menu was also included into our set of tasks. While not necessarily pertaining to image editing, menu navigation is an essential part when thinking about switching tools and orienting around an image editor. In the existing touch editors, navigation is usually achieved through scrolling, pressing and swiping.

There are two basic types of menus, one dimensional and two dimensional ones, and we included both in the study and asked participants to think of gestures for a menu list as well as a grid menu. These couple of tasks were probably the ones that most benefited from having descriptive images. Since we were asking users to execute very specific tasks (going down one item in the menu list and going down two items and then one to the right in the grid menu), we needed images to better explain these tasks. We thought that these tasks would be better understood when visualized and also easier to remember when actually executing the gestures for the tasks.

For navigating one item down through a menu list, we selected a different image from the previously used landscape and portrait ones, mostly to keep the participants visually interested. On the left side of the picture we added a list of various items mimicking the usual image editing menus. For the before-picture we selected a starting point item and for the after-picture we simulated the act of going down one item and circled the destination item (Figure 10).

Figure 10: How to navigate through a menu list [22]

For navigating through a grid menu we yet again selected a new image to demonstrate what the task was going to accomplish. We created a small scenario where we told the users that we had a menu of various filters and the effects they would have on a given image. In order to get to a desired filter (the one circled on the right side of Figure 11) participants were asked to create gestures for going two items down and one to the right from the original item (selected on the left side of Figure 11).

The reason for selecting tasks for both a list and a grid menu was that we wanted to determine if users were consistent in their menu navigation gestures, regardless of the format of that particular menu.

Figure 11: How to navigate through a grid menu [23]

3.2.5 Tasks 8, 9 and 10: Increasing and decreasing the intensity of a filter and reducing the size of an image

The three last remaining tasks, changing (increasing and decreasing) the intensity of an applied filter and making a photograph half of its initial size, were not as crucial to provide as, for example, cropping, but we wanted to enhance the potential of the image editor.

When portraying how increasing and decreasing the intensity of an applied filter affected a photograph, we selected changing the brightness of that image, since it is a fairly common used filter, suitable for exemplifying the task. The image was a simple, blue landscape, which effectively portrayed the brightness fluctuations (Figure 12 and Figure 13).

Figure 12: How to increase brightness

Figure 13: How to decrease brightness

Making an image half its initial size was more or less self-explanatory, but we wanted to be consistent throughout our tasks and study and provide a clear image of how the task would affect the somewhat gloomy landscape depicted in Figure 14. The reason why this example showcasing the effect of making a photograph half its original size is using the same image as the rotating and zooming tasks is that these were the first five tasks presented to the participants. Afterwards, we considered changing the pictures in order to keep the users visually stimulated.

Figure 14: How to make an image half its original size [18]

3.3 Creating the Testing Environment

After choosing the ten tasks for our study, we started working on setting up the testing environment. Using the Kinect Toolbox we wrote a program in C# Visual Studio that provided visual stimuli (different images) to the users and recorded their gestures using the Kinect.

For each image editing task that we had, the program would first display an image and its edited version according to the required task. Next, the unedited image was shown again, this time full screen, and each participant was prompted to generate gestures to accomplish a certain task. At the bottom left corner of the unedited image we added a small skeleton that tracked all the user movements (Figure 15). We wanted to add the skeleton feature because one of our main concerns was user reaction. How would participants respond when perceiving the fact that their gestures were not affecting the images in the desired way?

Because the application only records gestures and it does not have any knowledge of how these gestures relate to image manipulation and editing, it was important to provide users with a form of feedback. In order to minimize the startling effect of having nothing happening with the image, when the user started gesturing, we added the skeleton tracking the user's movements on top of the image being edited. This reassured the participant that the application was working and showed the way in which the program was interpreting the gestures through the skeleton tracking tool.

Figure 15: Skeleton tracking while generating gestures

The users were presented with this view on a large projection screen in order to mimic the environment in which such a tool might be used. The operation was controlled by a second view that was running on an attached laptop. This second view was a control dialog that allowed us to

set the participants' name, advance the image editing slides, have the Kinect start and stop recording as desired and select the gesture type and number.

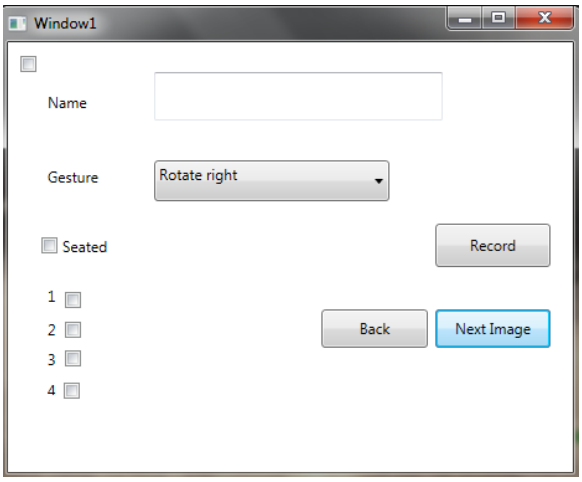


Figure 16: Programmer's control panel

3.4 Pilot Study

Once we had the testing environment, we started with one pilot study and three participants, the number of them being determined by the changes that we were constantly implementing and by the fact that we wanted to have our main study run as smoothly as possible. The goal with the pilot study was to discover how many gestures per individual task users could generate before fatigue or lack of inspiration settled in. We were also interested in analyzing the user-technology interaction and determining if any changes needed to be implemented in order to ease the

process. In addition, we wanted to find the most suitable way to display the gesture elicitation slides. Also, part of the aim of the pilot study was to validate the approach for our main study, since the alternative to using the skeleton tracking tool as a feedback device was the Wizard of Oz approach.

The "Wizard of Oz experiment" [24] gets users to interact with a computer program and leads them to think that it is autonomous, when in fact someone else is operating it, potentially incognito. However, we were trying to avoid the extra effort required to set up this particular testing environment and, indeed, the pilot study showed that implementing the Wizard of Oz experiment was not necessary.

After conducting the three runs of the pilot study, we were able to see and analyze our participants' response to interacting with the initially proposed testing environment. The skeleton tracking tool proved to be sufficient as the users were confident in carrying out their gestures. After being initially instructed and told that their gestures would not affect the images according to the image editing tasks, participants used the skeleton tracking tool as a reference on how their gestures would be interpreted and as a result they were not hindered by the application's lack of response.

We also discovered that three to four different gestures per task was a realistic number and if we were to ask for more gestural input, users in our pilot study tended to lose focus and lack in creativity. However, the central change that was implemented during the three runs of the pilot study was having participants concretely mark out loud every time a gesture started or ended.

Users were asked whether they were ready to begin gesturing so that the Kinect would not record any other noise but their actual intended gestures for the given image editing task. Similarly, whenever participants ended their motions, they had to signal it out loud using a relevant phrase. Before adopting this tactic, the person conducting the study had to estimate the start and end of a gesture, which usually resulted in unwanted data padding the beginning and end of the Kinect videos.

We were then able to proceed with our main study right after implementing some minor changes to the way the pilot study was designed. With the added seated control, the verbal priming of our users prior to the start of the study and the large display we felt confident about delving into the core study of this project.

3.5 Main Study

The main study was conducted over a period of two months and it encompassed the ten editing tasks previously discussed. We had fifteen people participating in our study, with each user session lasting between 30 to 40 minutes. The sessions included verbal instructions of the study, recordings of each of the gestures users would create for the editing tasks, as well as replays of some of the recordings at the very end. At the end of each session, we wanted to replay the recordings of the more ambiguous or complicated gestures back to the participants, so that they would have a second chance to explain what their gestures and their thought process were. The way we decided which Kinect recordings we would play back so that our participants could explain them, was the following: while we were recording their gestures we would mark on a

piece of a paper whether each gesture was self-explanatory and clear; if not then at the end of that participant's session we would ask him/her to clarify their motion paths and the reason for selecting that particular gesture.

The pool of fifteen participants was varied both in terms of gender and interests. Six out of fifteen users were male. Regarding majors or work affiliations we had users trained in engineering, business, music, American studies, architecture, economics, computer science, medical service, biology, art, psychology, education and mathematics. Most of our users were students; however three participants were college graduates who had been working in their field of study for more than a couple of years.

3.5.1 Priming and Testing environment

The final priming and set up of the testing environment which were applied for every participant followed the subsequent pattern:

Each participant was informed about the purpose of the study. We explained that the goal was to examine and record gestures that they would have to create when thinking about image editing tasks such as cropping, formatting, zooming or applying filters. Since different tasks can be interpreted in various ways, it was important to let the users know about our interest in observing trends and patterns and how they would interpret each editing action through gestures. We also emphasized the fact that that there was no right or wrong answer with regards to the way gestures were correlated with an image editing task. Moreover, participants were not only

encouraged to ask for clarifications during any time of the study, but also to verbally describe every motion and thought during their gestural process.

During the study an image editing task was selected from the programmer's control panel and the users were invited to apply it to the image displayed on their screen. They were presented with the initial image, then with the transformed image, and then they were shown the image again and asked for a gesture to accomplish the transformation. Each task was repeated four times by each participant, twice standing and twice seated. The one exception was participant one who was the first and only participant to generate three gestures per editing task and all while standing. After that, we decided that we wanted users to input gestures both while standing and being seated.

The reasoning behind this added feature was that we wanted to provide our users with the possibility of working with the final product (the image editor) not only while standing but also for example while sitting on a comfortable couch in front of their TV set. Since we were encouraging participants to create intuitive gestures, we also started to think of ways the image editor could be used as naturally and easily as possible. Having the product respond to seated and standing gestures allowed for a more relaxed, broader usage. In order to apply this feature, we added a new seated option to the control menu of our testing environment. When selected, the Kinect would only record the motions of the user's upper body and ignore the rest.

When thinking about the order of the seated and standing gestures, we decided that we would ask some users to first generate all the gestures for the ten tasks while standing and then repeat the

entire process while they were seated. We wanted to be certain that having participants create gestures first while standing did not influence their process, so we had the rest of the users switch the order of the standing versus seated gestures. The selection of which participants generated the standing gestures first was purely arbitrary.

Participants were also made aware of the existence of the skeleton tool tracking their movements and were encouraged to be confident and to not limit their gestures in any way.

Chapter 4

ANALYSIS OF MAIN STUDY

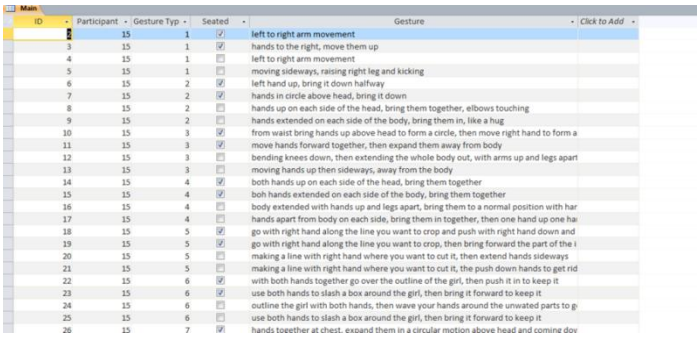
After conducting the main study, we had a plethora of gestural data both in terms of video and Kinect recordings. The primary task that surfaced after finishing the main study was finding the most fitting way to use all the informational data in order to determine the most common gestures for each given image editing task.

Having approximately 600 Kinect recordings, one for every gesture that was created during the study, and fifteen video recordings, each one representing a participant session, meaningfully sorting and visualizing the data in order to be able to start identifying cohesions became essential.

4.1 Organizing Data – First Approach

Databases are frequently used when dealing with large amounts of data. Our initial attempt at processing the collected information was to create a Book of Gestures database, a place where we could list each gestural entry (Figure 17). However, with approximately six hundred gestures generated after our main study, it meant that creating every entry in the database would require an extended amount of time. In order to input each gesture, rewatching the corresponding Kinect recording and video session was inevitable. Adding gestures to the database became a tedious

process that required launching a GestureViewer application, manually loading every Kinect recording into the application, viewing the gestural process and confirming it by also watching its matching video camera recording (which unlike the Kinect recording also included sound and therefore the participant's explanation of the gestures they were creating).



ID	Participant	Gesture Typ	Seated	Gesture	Click to Add
2	15	1	☑	left to right arm movement	
3	15	1	☑	hands to the right, move them up	
4	15	1	☐	left to right arm movement	
5	15	1	☐	moving sideways, raising right leg and kicking	
6	15	2	☑	left hand up, bring it down halfway	
7	15	2	☑	hands in circle above head, bring it down	
8	15	2	☐	hands up on each side of the head, bring them together, elbows touching	
9	15	2	☐	hands extended on each side of the body, bring them in, like a hug	
10	15	3	☑	from waist bring hands up above head to form a circle, then move right hand to form a	
11	15	3	☑	move hands forward together, then expand them away from body	
12	15	3	☐	bending knees down, then extending the whole body out, with arms up and legs apart	
13	15	3	☐	moving hands up then sideways, away from the body	
14	15	4	☑	both hands up on each side of the head, bring them together	
15	15	4	☑	boh hands extended on each side of the body, bring them together	
16	15	4	☐	body extended with hands up and legs apart, bring them to a normal position with har	
17	15	4	☐	hands apart from body on each side, bring them in together, then one hand up one ha	
18	15	5	☑	go with right hand along the line you want to crop and push with right hand down and	
19	15	5	☑	go with right hand along the line you want to crop, then bring forward the part of the i	
20	15	5	☐	making a line with right hand where you want to cut it, then extend hands sideways	
21	15	5	☐	making a line with right hand where you want to cut it, the push down hands to get rid	
22	15	6	☑	with both hands together go over the outline of the girl, then push it in to keep it	
23	15	6	☑	use both hands to slash a box around the girl, then bring it forward to keep it	
24	15	6	☐	outline the girl with both hands, then wave your hands around the unwated parts to g	
25	15	6	☐	use both hands to slash a box around the girl, then bring it forward to keep it	
26	15	7	☑	hands together at chest. expand them in a circular motion above head and coming dov	

Figure 17: Book of gestures database

Moreover, using a database also required developing a language of gestures in order to capture each gesture in a concise way that was at the same time comparable with all the other gestures' descriptions. This process proved to be very problematic, since every time a new gesture was added to the database it meant that we had to check with all exiting entries to see if analogous ones had already been entered. This eventually also led to a lack of accuracy since each gesture description was based on our summary. As we were adding gestures to the database, we constantly had to look if a similar gesture had already been entered and if that was the case, then their descriptions needed to be identical. The constant examination added to the time required to

view all of the recordings and we decided that if we wanted a more precise and swift process, the analysis approach had to be altered.

4.2 Visualizing Data – Second Approach

Instead of relying on words to describe each gesture, we decided to create something with a visual impact. The aim was to be able to collectively look at all the gestures and promptly identify commonalities since visualization would allow us to look at a large number of images embodying gestures simultaneously. This would not only signify that we could more easily identify patterns, but also that we could increase the level of accuracy when looking for similar gestures.

Decoding each Kinect recording of a gesture into an image proved to be a challenging process due to the inherent struggles when trying to capture a 3D temporal action into a single static representation. The resulting image had to encapsulate not only the gesture as a whole, but also the passage of time, and how the gesture progressed and changed through time. We wrote a specialized program in C# that sequentially loaded each Kinect recording and as a recording was loaded an image was generated. The program would construct an image from a Kinect recording by looking at its every frame and drawing onto a digital canvas the skeleton joints from that frame. Each joint was represented as a solid, round, filled in dark blue brush stroke. The skeleton joint for the head was the only exception, as it was drawn as a dark blue ellipse with a hole in its center (“donut shaped”). The program would draw the joints from the head down to the feet for

each frame, overlaying them on top of each other. After the last frame the canvas was converted into a PNG (Portable Network Graphics) image and saved onto an external drive.

Figure 18: First visualization attempt

Figure 18 portrays the image that was created from a Kinect recording of a gesture for the task of navigating downwards through a menu list. The reason why the head is not an empty center ellipse is that the overlaying of the head joints from each frame filled the hollow center. By looking at the image one can infer that only the participant's left hand moved since all the other joints have been perfectly drawn on top of each other therefore keeping their original position. The one variable missing from this picture is how the left hand moved through time. Did it first go up at an angle, then for a short period of time straight down, then down again hitting the leg as reflected on the left side of Figure 19? Or did the left hand first go up at an angle, then straight up again and finally downwards as portrayed on the right side of Figure 19?

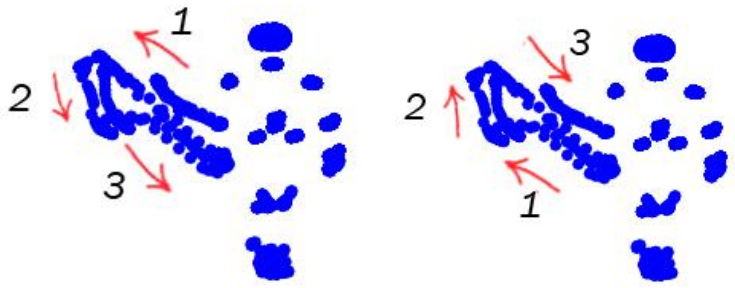


Figure 19: How did the gesture progress?

It is almost impossible to answer the questions above without estimating and speculating. One could only guess the direction in which the gesture progressed. Therefore, in order to be accurate, determining how the gestural pattern changed through time required introducing a different coloring technique. Color gradients were the ultimate choice since they reflect transition perfectly. We decided that for the first frame of a Kinect recording, the program would draw the joints using a dark blue hue and as more frames were drawn the color would gradually become lighter. Reading how the gesture changed and transformed in time thus became clearer, since the lighter the joints became, the later in time they were executed.

Another change that was implemented was that the joints were not drawn fully opaque as depicted in Figure 18 and Figure 19; we added a slight transparency to the color of each joint so as to be able to better perceive what was drawn a few frames back. Full opacity has an alpha value of 255, but after multiple overlaying trials, we selected an alpha value of 180 for our colors. We selected an in-between value since full or too increased transparency made the

contour of individual joints very difficult to perceive. Conversely, full opacity reduced the amount of information that could be read from the images.

Moreover some of the joints were completely removed: the left and right wrists, the central shoulder and spine joint. The reasoning behind this decision was that the wrists, central shoulder and spine joints were adding extra noise to the image without being relevant. For example, if the right hand and shoulder moved, we knew that the right wrist also moved and therefore drawing the wrist joint offered no additional information. Similarly the central shoulder joint was just below the neck and in between the right and left shoulder joints. Its movement was completely dependent on the other joints' displacement and as a result we chose to ignore it. So was the case with the spine, which was placed in the middle of the hip joints and could not move independently.

Figure 20: Second visualization attempt

Figure 20 displays the same gesture as Figure 18 and Figure 19: navigating downwards through a menu list. However, the implemented changes allow for a clear interpretation of how the gesture took place. The confusion portrayed in Figure 19 is no longer a concern, since the color graduation, from dark to light and the slight transparencies, allow us to see that the left hand first moved up and then down as depicted by the red arrow in Figure 21.

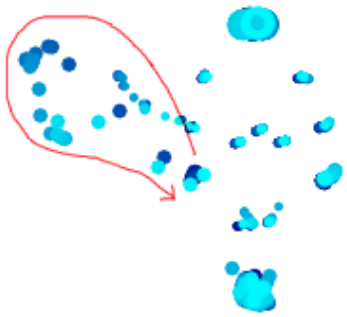


Figure 21: Second visualization attempt; gesture using one hand

Another visible change is that not all joints are of equal size. Since shoulder, hip, knee and elbow joints tended to be not as highly used or used in correlation with the hand and foot joints, they were drawn smaller in size so as to reduce the overall noise yet again. This visualization approach, which used a reduced number and different size skeleton joints, as well as the dark blue to cyan color gradient and the translucent brush strokes, produced satisfying, easier to comprehend images. However, if for example both hands were used to generate a gesture using

one gradient color became problematic. Imagine a crisscrossing movement of the hands. It would be almost impossible to identify which joint belonged to which hand.

Figure 22: Second visualization attempt; gesture using both hands

Figure 22 represents such a case, where both hands moved. It is difficult to tell whether each hand moved on its respective side of the head, or whether the two hands intersected paths. In order to be able to make that distinction we introduced two more gradient colors. As the gesture progressed, the left side joints of the skeleton would transition from dark blue to cyan, the right side from purple to pink and the center skeleton (the head, shoulders and hips) from black to gray.

Figure 23: Third visualization attempt

Figure 23 displays the one color gradient image next to the new three colors image. What was not evident before was that the right hand did move to the left side of the head. Looking at the image on the right of Figure 23 one can see that while the left hand joints had an arched up and down movement constrained within the left side of the body, the right hand joints moved up towards the left side of the head, then horizontally across and towards the right, then down back to its initially resting position (as represented in Figure 24).

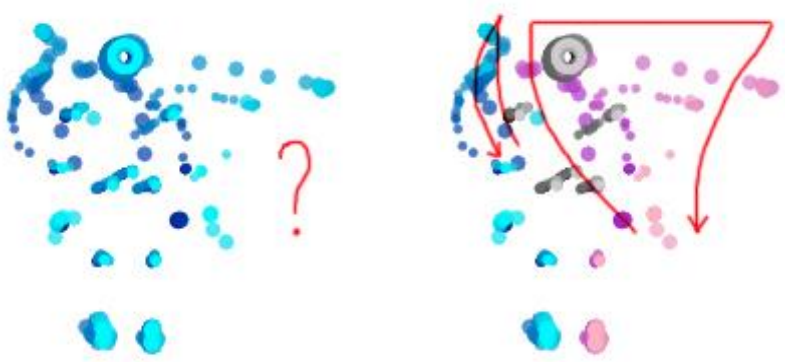


Figure 24: Progression of gesture

The three transitioning colors, the change in opacity, a reduced number of joints as well as different sized skeleton joints based on their importance, were the final features of the program that generated all six hundred PNG images of the Kinect recordings encompassing all our participants' gestures.

4.2.1 Identifying Commonalities

Organizing the six hundred visual representations of gestures was crucial in the process of identifying their similar features. All gestures pertaining to a specific image editing task were manually grouped in a Photoshop file. For example, for the rotating an image to the right task, the analogous Photoshop file contained all the corresponding seated and standing rotating gestures coming from the fifteen participants. Since we wanted to be able to distinguish which gesture belonged to which participant, the gestures in the Photoshop files were placed in rows, each row representing a participant (as depicted in Figure 25).

Figure 25: Partial Photoshop file of all the rotating gestures generated by the 15 participants

The Photoshop files were fairly large, and fitting them on a computer screen without being extremely zoomed out was impossible. We wanted to analyze them in as large of a format as possible, since seeing every detail was of great importance. Therefore, we printed all these files. The prints were each approximately three feet in height and allowed us to look at all gestures for a particular task collectively. Analyzing each print and judiciously marking similar gestures using color coding resulted in us reaching one of our key goals: finding similar gestures for each editing task (Figure 26). Interpreting the visual representations of every image was however not a trivial task. In order to be sure that we correctly identified all gesture similarities we not only looked at the prints but also at the corresponding Kinect recordings of those motions that proved to have a lot in common with others. The images that looked intricate or hard to decipher were also re-checked with their equivalent Kinect recordings.

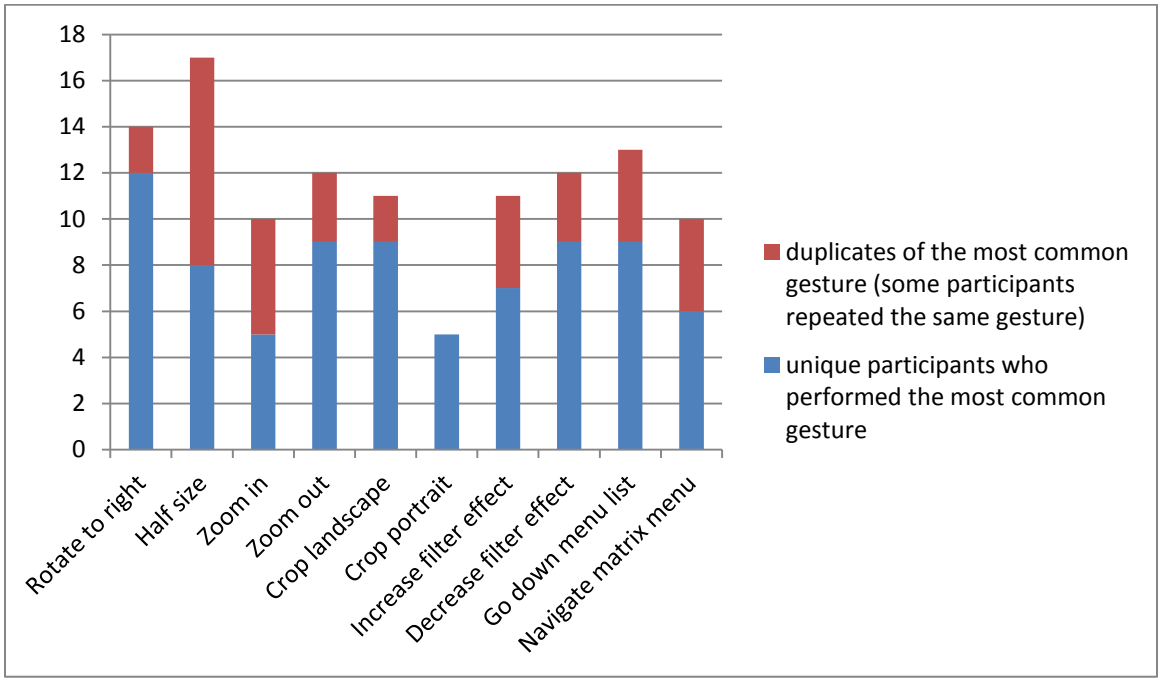


Figure 26: The number of users that generated the most common gesture for a particular image editing task

4.3 Results – Common Gestures

The graph portrayed in Figure 26 showcases the numbers for only the most common gesture for each of the ten image editing tasks. However, for every task that we had, we were able to not only identify one common gesture, but also other gestures that while not as widely shared were still generated by two or more participants. In the following subsections we will be looking at both the top most used gesture and the less common used ones.

4.3.1. Rotate Image to the Right Task

For the rotate an image to the right task, Figure 27 portrays the most common gesture generated by twelve different participants in fourteen distinct instances (some participants created the same gesture multiple times during the four gestures per task sessions). This gesture can be described as a circular pattern: the right hand starts at head level while the left hand starts at the left hip level. As the right hand begins arching down, the left hand rises following a curving path as well.

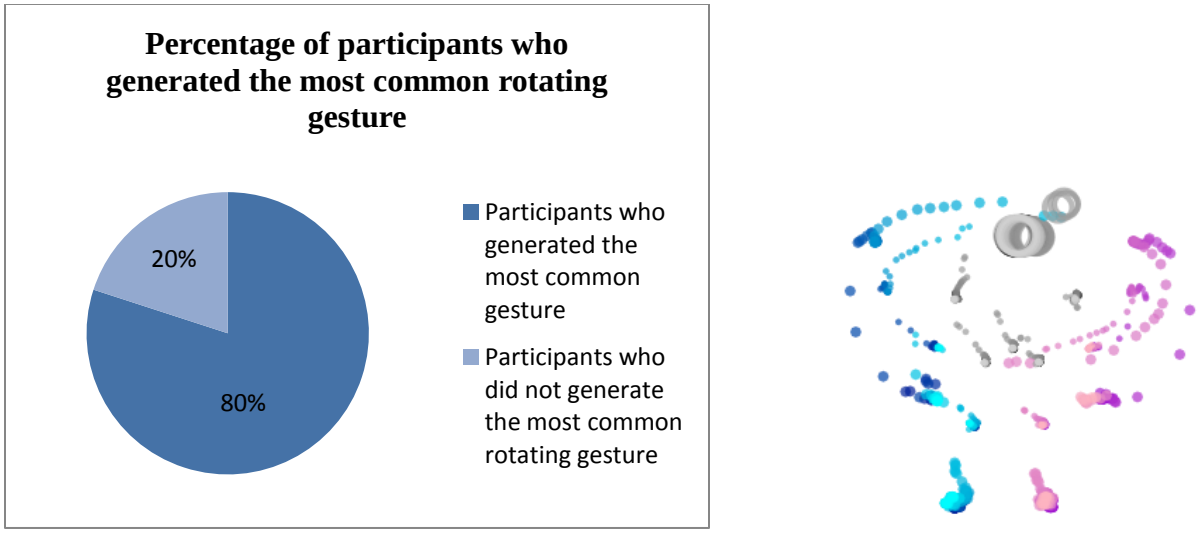


Figure 27: The most common gesture for rotating an image to the right

This clockwise movement of the hands was definitely the most popular rotating gesture. However, there were two other common gestures which only had six similar instances each (Figure 28). They represented a variation of raising either one or two hands.

Figure 28: Two other rotating gestures that were shared by participants

4.3.2 Making an Image Half the Size Task

Making an image half its original size was a task for which a lot of participants agreed on one specific gesture (represented in Figure 29). This gesture was created by eight participants in seventeen instances (again some participants repeated their gestures). The motion starts by lifting both hands and having them reach head level. However, one of the hands is at a slightly higher altitude than the other. Progressively, both hands close in towards the head. One important mention is that while Figure 29 shows the gesture as being executed starting with the right hand a little higher than the left hand, we also considered the opposite (left hand higher, right hand lower) as being the same gesture. When lacking inspiration, participants would often mirror their gestures in order to create more and reach the four gestures per task quota. Therefore, we counted all such pairs of gestures as being the same.

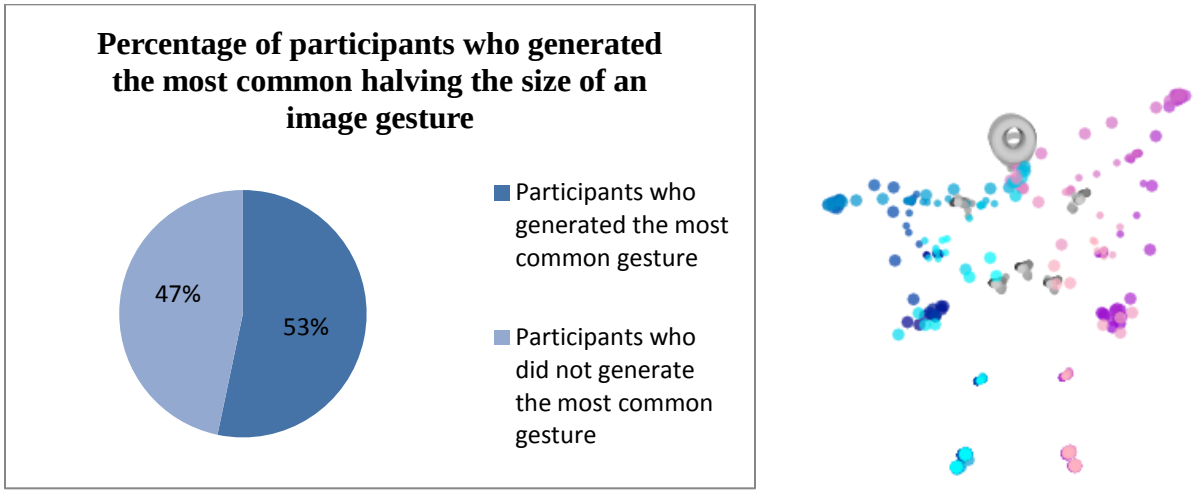


Figure 29: The most common gesture for making an image half the size

Another gesture shared by our users, yet not in such significant numbers (6 participants and 14 instances as opposed to 8 participants and 17 instances for the most common gesture) was one involving the extension of both arms to the left and right side respectively (Figure 30).

Figure 30: Another shared making an image half the size gesture

4.3.3 Zoom in Task

The Zoom in task also had a two-handed gesture for its most common motion (Figure 31). Five participants in ten instances created this same gesture. The gestural motion starts with both hands up and close to each side of the head, as depicted by the darker brush strokes. In order to zoom into a picture, participants then extended both hands outwards (brush strokes become lighter towards the edge of the image). If we did not implement the color gradation, it would have been impossible to determine whether the gesture progressed inwards or outwards.

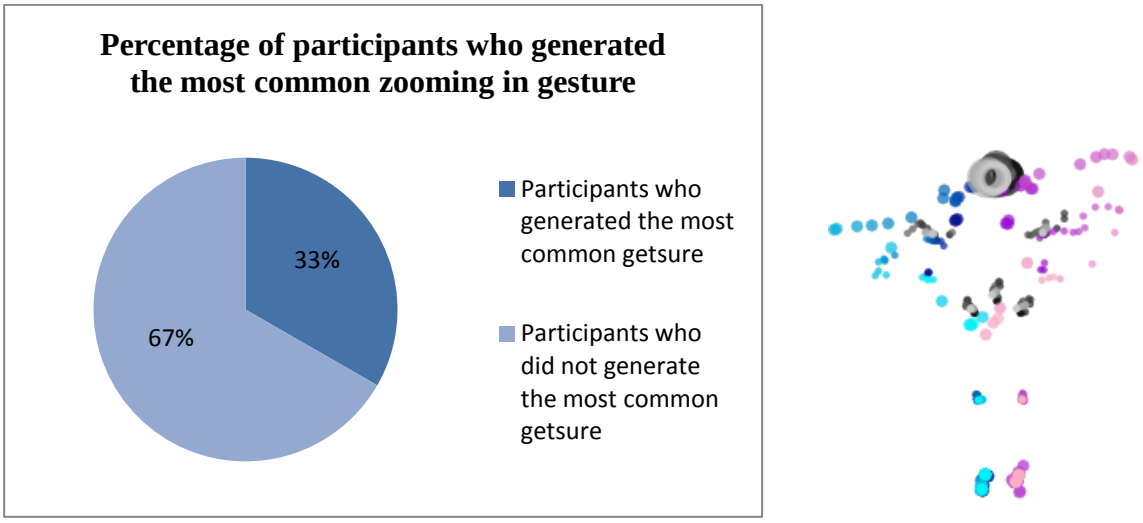


Figure 31: The most common gesture for zooming in

There was another gesture that was almost as popular as the one described above (generated in 9 instances as opposed to 10 instances for the most common one). This gesture involved raising

both left and right arms as seen in Figure 32. There is some negligible shuffle within the body, which naturally occurred as the participant was raising both arms.

Figure 32: Another shared gesture for zooming in

4.3.4 Zoom out Task

The Zoom out task had a gesture (Figure 33) that was the exact opposite of the Zoom In and identical to the Making and Image Half the Size tasks. Nine different participants created this gesture in twelve distinct instances. Both hands rise at an angle until they reach the level of the head and then gradually start coming together. While it was expected that the Zoom out gesture would be the reverse of the Zoom in motion, since the two tasks are antonymic, it was interesting to discover that halving the size of an image and zooming out were extremely similar. The way this gesture would be implemented with the image editor and differentiated from the task which produced the same most common motion, is a process that will be described in the following chapter.

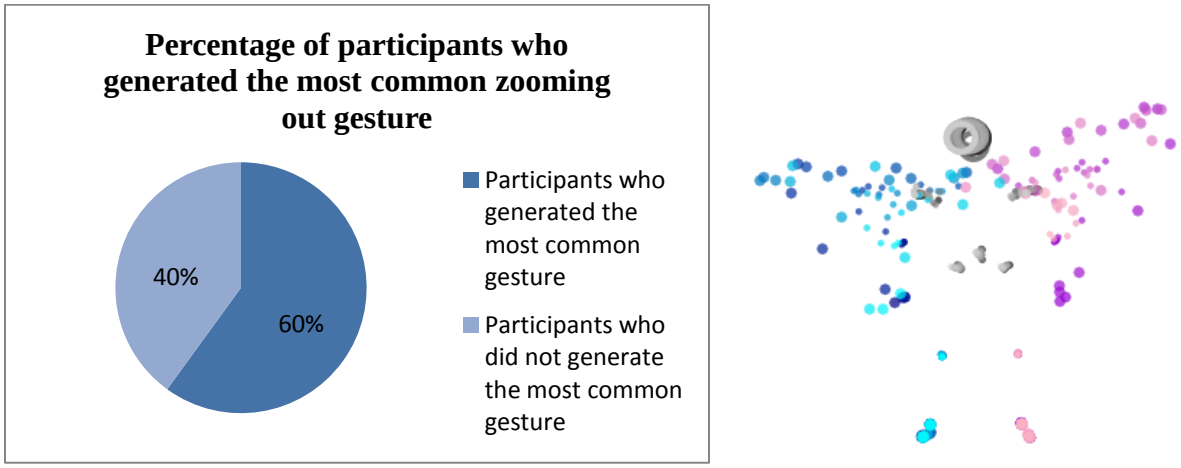


Figure 33: The most common gesture for zooming out

For zooming out we could not identify any other shared gestures amongst our participants.

4.3.5 Crop Landscape Task

For Cropping a landscape most participants generated the gesture portrayed in Figure 34. This task asked participants to crop a golden field which was topped by a sky in such a way that the resulting picture would contain just the field. Nine out of fifteen participants in eleven instances created the same motion which can be described as a horizontal slashing. The right hand starts by going up (reaching the level where the image needs to be cropped) and then repeatedly moves from left to right along an imaginary cropping line. This cropping task was very particular and closely related to the chosen landscape image, so it was important to follow and determine whether similar, cohesive gestures were created for the portrait cropping task.

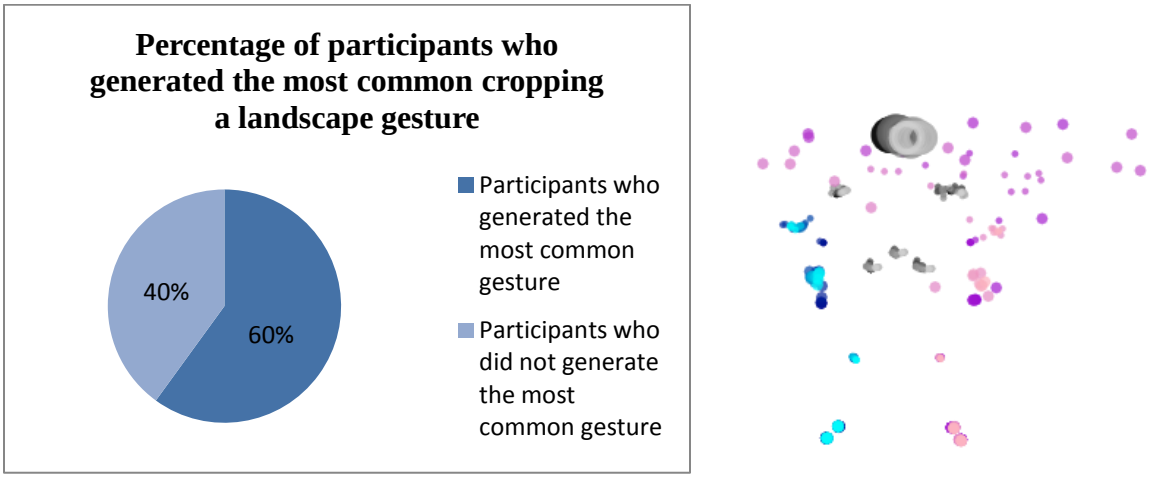


Figure 34: The most common gesture for cropping a landscape

We identified two other shared gestures, one generated in 5 instances and another in 4 instances. The gesture created in 5 instances involved moving both hands towards the right side of the body, while the other third most common gesture proposed raising both hands (Figure 35).

Figure 35: Two other cropping the landscape gestures that were shared by participants

4.3.6 Crop Portrait Task

Cropping a portrait proved to be a more challenging task that had less common gestures than most of the other editing tasks. Five participants in five instances generated the gesture depicted in Figure 36. Users were asked to crop the face of a young woman. The most similar motion involved slashing patterns, identical to the ones used for cropping a landscape. However, for this task, participants used both vertical and horizontal line hand movements, ultimately framing the woman's face in the picture.

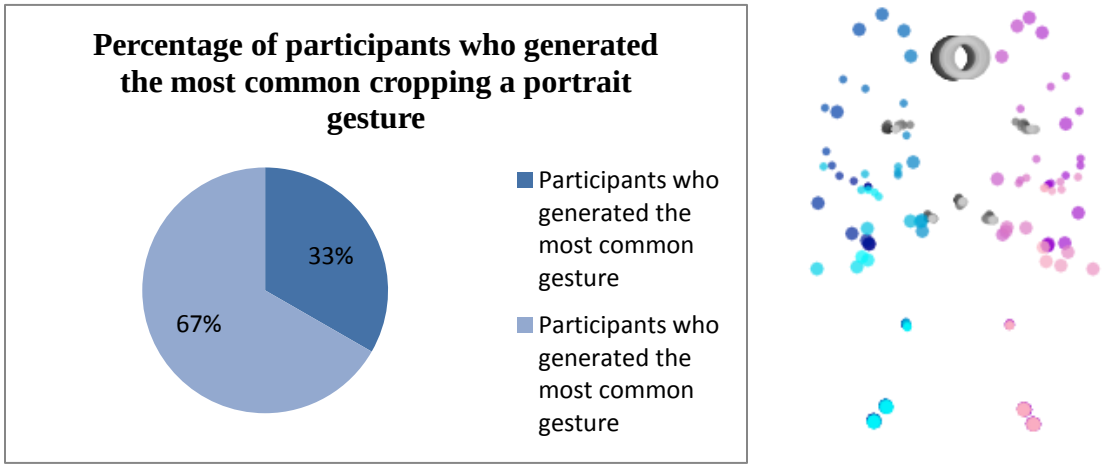


Figure 36: The most common getsure for cropping a portrait

For cropping a portrait, another shared gesture was one created by 3 participants as opposed to 5 for the most common one. This gesture is represented by the extension of the right and left arms and it also involves a slight shift of the body (Figure 37).

Figure 37: Another shared gesture for cropping a portrait

4.3.7 Increase Filter Effect Task

Increasing the effect of a filter which was already applied on the image being edited was a task that resulted in a lot of common gestures. Seven participants in eleven instances created the same gesture (Figure 38). This one-handed gesture starts by having the hand in the resting position by the hip, and the gradually bringing it up. Once again participants mirrored this gesture and alternatingly used both the right and the left hands to describe the same task. These gestures were counted as being similar since apart from being executed using opposite hands, they followed the same upwards path.

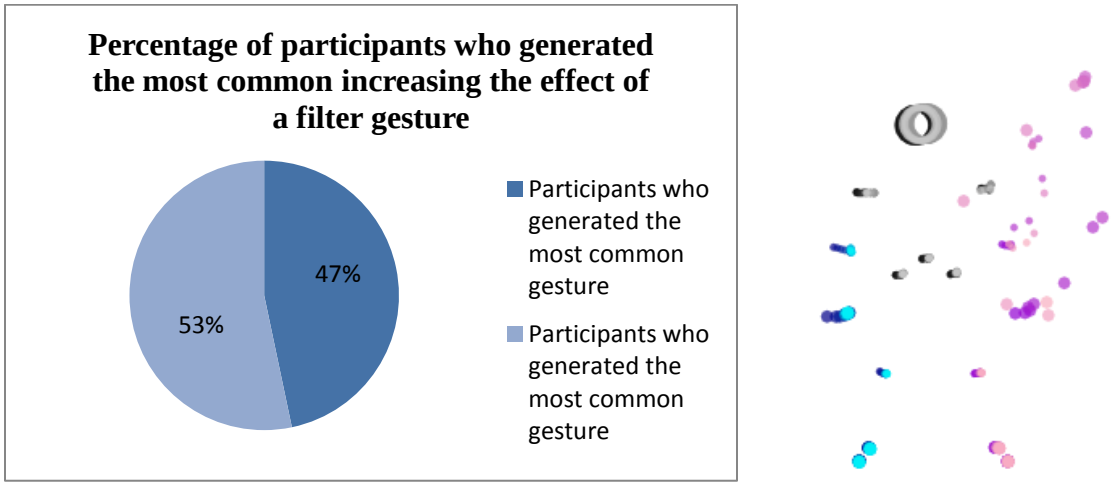


Figure 38: The most common gesture for increasing the effect of a filter

We also identified two other shared gestures for this particular task (Figure 39). They were generated in 3 and 4 instances as opposed to 11 instances for the most common gesture. The first gesture represented extending the right arm, while the second one involved raising both hands.

Figure 39: Two other fairly common increasing the effect of a filter gestures

4.3.8 Decrease the Effect of a Filter Task

For Decreasing the Effect of Filter task, nine out of fifteen users in twelve instances created the gesture showcased in Figure 40. Similar to the Zoom in and Zoom out tasks, Decreasing the Effect of a Filter is the counterpart of the task which increased the filter effect. The dark purple brush strokes representing the right hand joints go upwards reaching a starting position and then progressively lighten up suggesting a downwards motion. This gesture was yet again created using both the right and left hands.

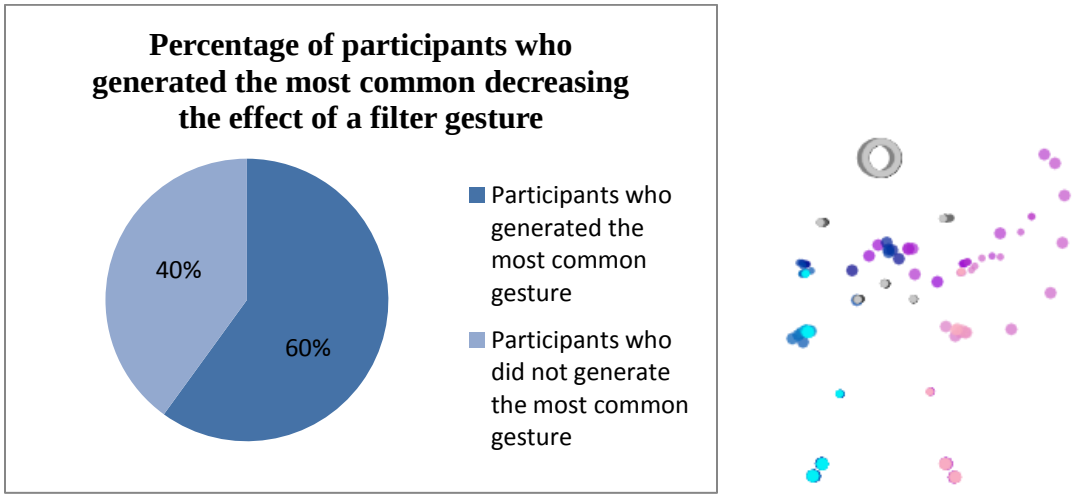


Figure 40: The most common gesture for decreasing the effect of a filter

Another shared gesture for this task was one created by 5 participants as opposed to 9 for the most common gesture (Figure 41). The gesture is represented by a right to left hand swipe.

Figure 41: Another shared gesture for decreasing the effect of a filter task

4.3.9 Go down through a Menu List Task

Navigating downwards through a one dimensional menu was a task that generated the same gesture among nine participants in thirteen different instances. The most common gesture (Figure 42) is similar to the one depicted in Figure 40 (decreasing the effect of a filter task) and has one of the hands moving from atop and following a downwards straight path. Figure 42 reflects the previously mentioned mirrored gesture case, which in this instance uses the left hand.

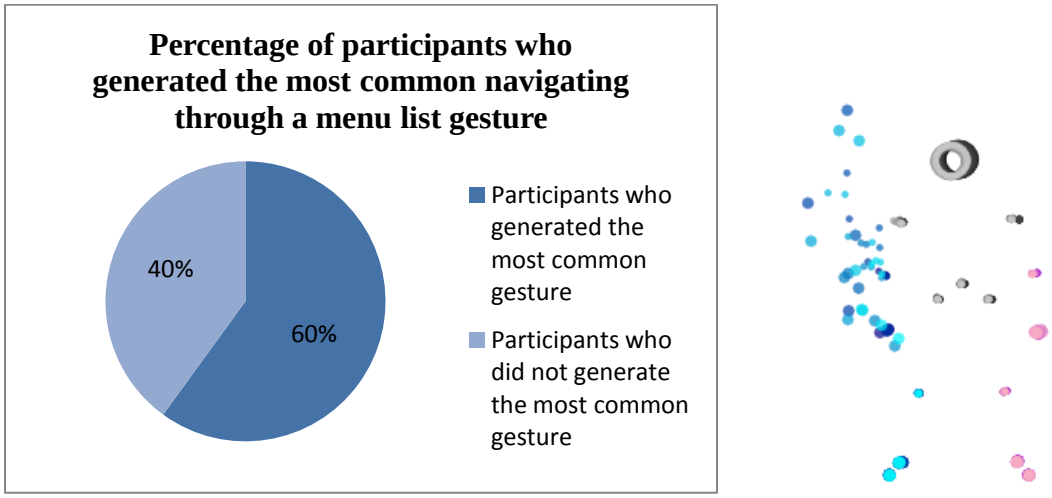


Figure 42: The most common gesture for navigating through a menu list

An additional shared gesture is the one depicted in Figure 43, a gesture that was generated in 7 instances (there were 12 instances of the most common gesture). The motion described below is one that involves lowering the whole body (the participants seem to have created an analogy between going down an item in a menu list to moving their bodies downwards).

Figure 43: Another common gesture for navigating through a menu list

4.3.10 Navigate through a 2-dimensional Menu Task

For the last image editing task, navigating through a matrix shaped menu, users had to follow a specific given route. Six different participants replicated this gesture in 10 instances. They were instructed to move downwards and then towards the right side of the menu. The most common gesture (Figure 44) starts at the level of the head, moves down, then to the right ending at the resting position with the hand by the right hip. The gesture clearly reflects the navigation itinerary with the hand closely following each turn in the path. The goal with both the one dimensional and the two dimensional menu tasks was to identify gestures that would enable navigation through any given list of options. In both cases the answer was a one handed gesture, moving up, down, left and right in order to reach the desired item in the menu.

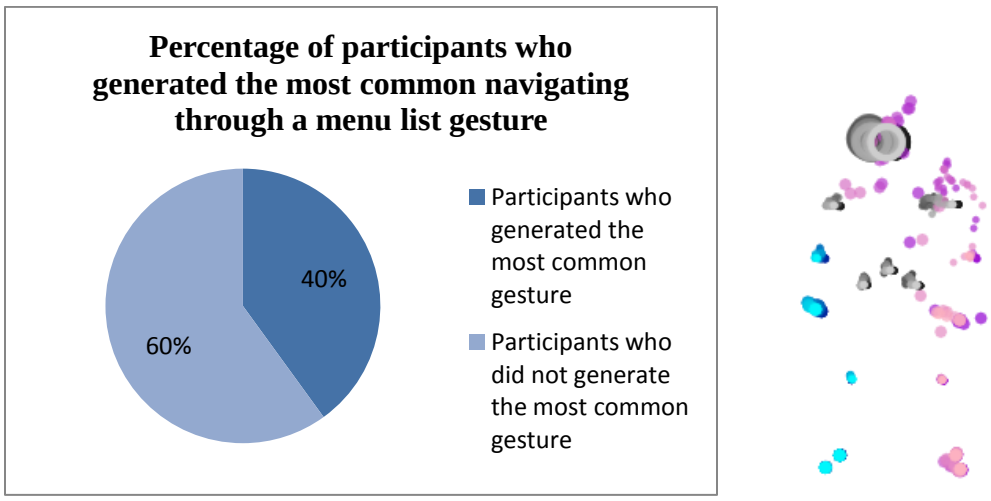


Figure 44: The most common gesture for navigating through a matrix shaped menu

We also identified another common gesture, almost as popular as the most common one, which was generated in 8 instances, two less than those of the most popular gesture. Figure 45 portrays this second most created matrix shaped menu navigation gesture, which involves raising the left arm.

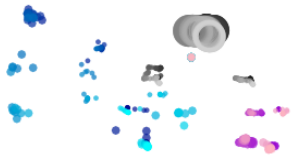


Figure 45: Another shared gesture for navigating through a matrix shaped menu

4.4 Participants' background

After identifying the most common gestures for each of the ten editing tasks, we wanted to analyze the ways in which participants' backgrounds influenced the types of gestures they were generating.

As previously discussed, the pool of participants was varied both in terms of gender and interests. Participants' academic work, occupations and leisure pursuits visibly influenced the way they performed during the study. Participant one, who has a vast passion for video games and new technologies, had a very clear vision of what his gestures were going to be. For every

task, he generated precise and concise gestures that allowed him to accomplish the editing tasks swiftly, similar to the way one needs to manipulate a controller while playing video games. Having also used gestural detection before, he understood the limits of the Kinect (for example Kinect cannot track finger movement). However, there were also some drawbacks to the fact that he had experimented with gestural detection. In various occasions he tried to recreate gestures he had used with other previous applications; for example when asked to navigate through a menu, participant one generated the same motions he usually employs when navigating through the list of songs of The Dance Central for Kinect game. Nevertheless, the fact that participant one (like all other users) had to generate multiple gestures for each of the image editing tasks did require him to create his own gestures, since after exhausting the gestures he was familiar with from previous games/applications, he had nothing but his imagination and instinct to rely on.

The third participant had a background in music. Having never worked with the Kinect and only superficially with image editing, he was very new to the ideas we were presenting to him. Additional explanations were required for almost all tasks. For example, when asked to imagine that a filter had been applied to the image he was viewing and that he had to create a gesture that would increase the effect of the filter, he became extremely confused. He didn't understand how a filter's effect could be increased and therefore correlating that idea with a gesture seemed almost impossible. We explained the task in various ways, which was fairly difficult since we did not want to give away too many hints and bias his gestural process. For instance when referring to the same task of increasing the effect of a filter as strengthening the effect of a filter, participant six took it literally and created a gesture similar to a flexing arm. It was therefore

important to try and limit our vocabulary when explaining each editing task and use more of the example slides that showed what the effects of the task were supposed to be. If users still could not comprehend what the task was and they could not think of a suitable gesture (which was the case only with participant three during the increasing the filter effect task), we asked them to create a gesture that matched even their most minimal understanding of the task.

4.5 Outliers

One of the most prominent outliers in the study was participant eight. She was particularly passionate about dancing, improvisation and choreography and her interests reflected in all of the gestures she generated during the study. The majority of her motions were intricate yet fluent, composed of a multitude of gestures and ideas.

Figure 46 illustrates how little in common the eighth participant had with the other users. The X axis marks each of the 10 editing tasks, while the Y axis represents the number of common gestures participants executed for every task. Participant 8 is represented by the blue line in the graph below, a line which spikes only once marking the two gestures that this user had in common with everyone else. The other maroon line represents the mean of common gestures of all the other 14 participants, and it illustrates how for every task almost all 14 participants shared a gesture with someone else. Surprisingly, for the more challenging task of cropping a landscape or a portrait, a task that had not as many cohesive, matching gestures within the pool of fifteen participants, participant eight had two gestures resembling those created by other users.

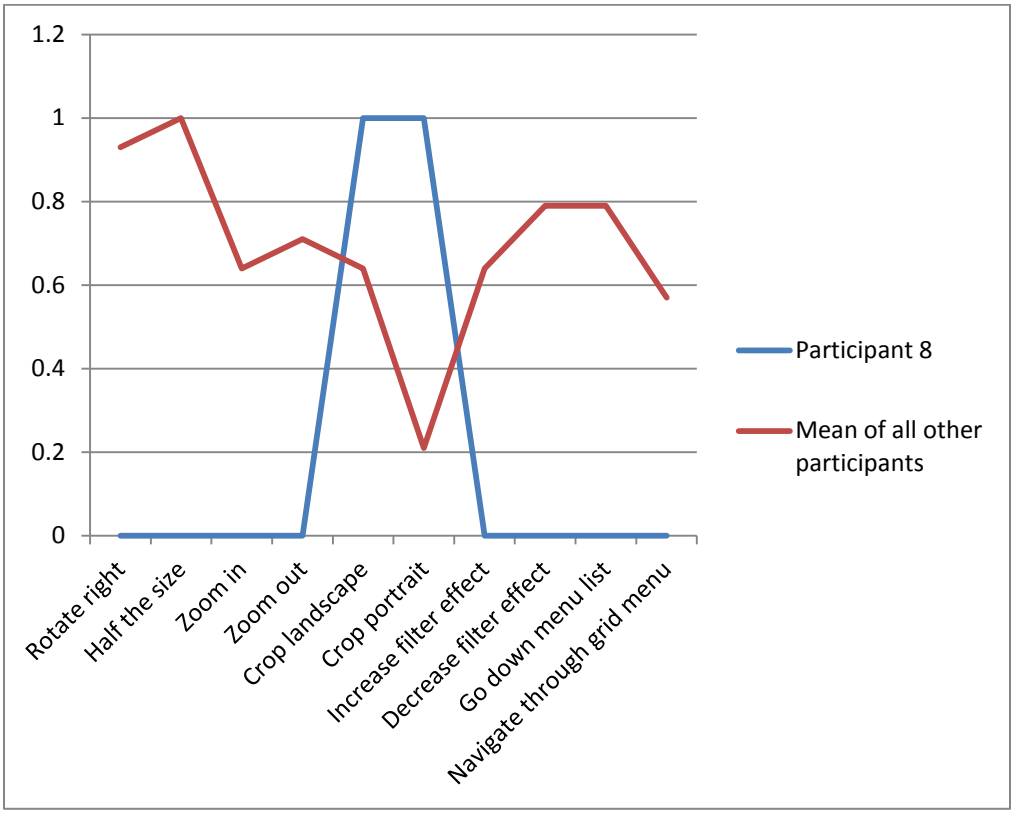


Figure 46: Participant 8 marked with by the blue line is the greatest outlier in our study. Its corresponding graph line fluctuates once to mark two gestures for cropping a landscape and a portrait which Participant 8 shared with the other participants.

Unlike other users, participant eight moved sideways and up and down frequently. She also used various body parts simultaneously in order to express one editing task, something not as regularly implemented by the rest of the users. What was even more interesting was that when we were analyzing the Kinect motion traces and we did not look at what participant was responsible for the gesture we were on, we could still tell which gestural motions belonged to participant eight. Her fluid, complicated, multiple parts movements were unique and almost no other users thought of the same gestures.

Another outlier was participant seven, but for very different reasons. This participant did not show the same excitement or interest as most of the other users. Tired and lacking motivation, participant seven generated almost the same gestures within a given task. Therefore, instead of creating four distinct gestures per task, she made a slight variation of the same gesture for the four instances of the task. This was especially noticeable when she transitioned from the standing to the seated gestures. Since participant seven was only using her upper body to create gestures while standing, when she had to repeat the study while seated, there were no challenges in recreating the same gestures she had when standing. This pattern repeated with other users, who transitioned their gestures from the seated to the standing study and vice versa. However, participant seven was the one who consistently did so for most of the tasks.

4.6 Utilization of the Skeleton Tracking Tool

When we added the skeleton tracking tool, we could not anticipate all the ways participants were going to use it. One of the main reasons for incorporating the skeleton tracker was so that users could be reassured that Kinect was correctly interpreting their gestures during the recording process. However, participant two decided to utilize the skeleton tracking tool in a unique, unexpected manner. Before starting the recording of any of his gestures, he would first test them with the skeleton tool. He would try a particular gesture several times until he was sure it looked exactly as he envisioned it. Once he was certain of his gesture and of the way it was going to look and be interpreted by the Kinect, he would signal that he was ready to start the recording process.

Participant twelve also showed an elevated interest in the skeleton tracking tool. He was very inquisitive and asked us how it was working and what exactly it was tracking. The way this tool functions is by firstly identifying each joint in the human body. While Kinect is recording, for each frame it generates these joints and draws small representative ellipses. It then connects the ellipses to simulate the way bones connect joints. When the seated control is pressed, the joints corresponding to hips, knees and ankles are not looked at and therefore not drawn. As previously mentioned Kinect does not offer finger detection and draws the whole hand as one joint. This was important to mention at the beginning of each of the participants' study case. Even so, in some cases users still tried to use finger movement, especially when they wanted to utilize pointing or during finer tasks such as cropping. They had to be reminded that they were limited to using their hand as a whole, something that did hinder their creative process.

4.7 Standard Participants – Commonalities

Apart from a couple of outliers, participants had a lot of gestures in common. Most of the commonalities surfaced especially during editing tasks that users were either accustomed to or for those whose description was very explanatory and easier to match with a gesture. The tasks that users seemed to be more familiar with and thus create more gestures in common were: zooming out and navigating through a menu (Figure 47). Tasks such as rotating an image, making an image half the size and decreasing the effect of a filter were those kinds of tasks for which users tended to exactly match the description to a gesture. For example rotating an image to the right prompted almost all users (12 out of 15) to create a clockwise rotational movement of

the arms. Cropping, while still a very familiar task, it was harder to accomplish through gestures and users generally had different, unique ways of interpreting it.

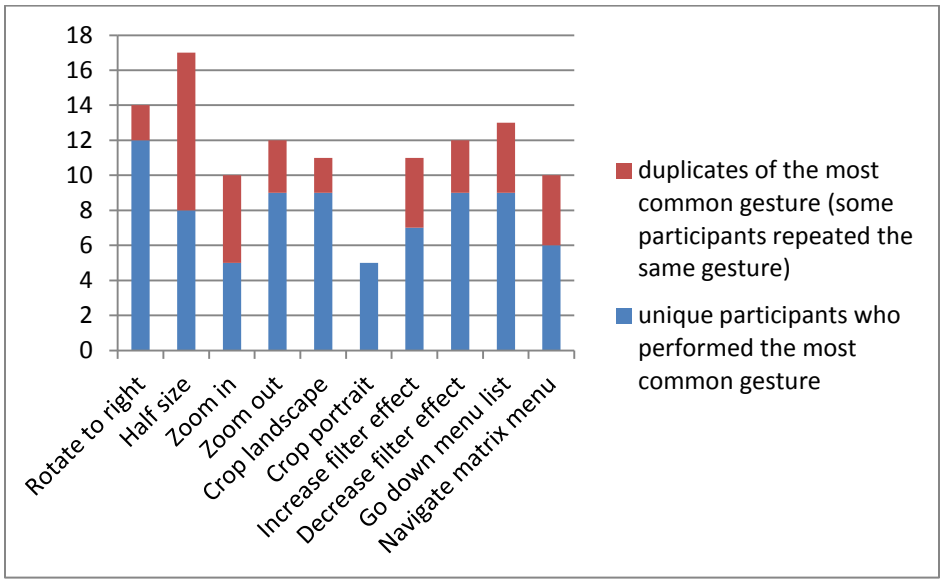


Figure 47 copy of Figure 26: The number of users that generated the most common gesture for a particular image editing task

Participants without a background in art were just as quick and resourceful as other participants who had artistic inclinations. However, participant eleven, who practiced image editing intensely, did have a valuable, insightful suggestion on how the study could be improved. He thought that a nice feature would be asking the users to input gestures not only while standing and being seated, but also while using only one hand. Not having one of the hands tracked would allow for the users to possible write things down using the free hand while still gesturing with the other, or even hold an object (e.g. a cup of coffee) or simply rest the arm if fatigue settles in.

Moreover, when looking at users who had used the Kinect gesture tracking technology before and those who had never encountered it, we could not determine any noticeable differences. Participants who had previously seen and worked with it did not require as much explaining and priming, but in terms of gesture fluidity and naturalness there weren't any evident dissimilarities.

The following chapter will be looking at how each of the shared gestures was used to assemble a proof of concept image editor, the foundations of an application which raised various interesting issues: How do we recognize complex gestures? How do we recognize the start and end of gestures? And, how do we distinguish between multiple tasks that share the same gestures?

Chapter 5

THE IMAGE EDITOR

Having identified common gestures for each of the ten editing tasks we were faced with the question of discovering if they could be implemented in a meaningful way. To explore this, we built a proof of concept image editor, which would use the gestures that proved to be the most common and intuitive.

5.1 Creating a basic Image Editor

The foundation for a gesture recognition image editor was a traditional image editor based on a WIMP interface. This basic editor was created in C# Visual Studio and relied on color vectors and matrices to perform editing tasks. In C#, as in many other programming languages, colors are represented as vectors, each composed of 5 elements (Red, Green and Blue channels and alpha and w values). Commonly, the intensity of each channel sample is defined by 8 bits and they are organized in memory in such way that a single 32-bit unsigned integer has the Alpha sample in the first 8 bits, followed sequentially by the Red, Green and the Blue sample in the last 24 bits (Figure 48) [25].

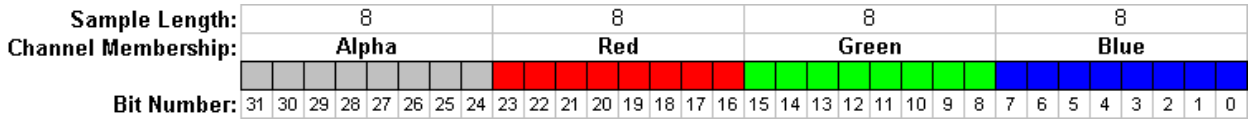


Figure 48: Representation of each color channel for a particular color

Each filter or transformation is a 5x5 color matrix that represents a linear transformation of the color vector assigned to each pixel. A color matrix defines a 5x5 matrix that contains the coordinates for the RGBA (red, green, blue, alpha) space. As basic functionality, we implemented increasing and decreasing brightness, sepia, black and white and various other filters, cropping, resizing and zooming. In order to achieve the desired filters and brightness effects we used color matrices. For a given image, the algorithm looked at every pixel, read its color vector and multiplied it with a specifically constructed color matrix. For example for the grayscale filter the color matrix was the following:

```
ColorMatrix colorMatrix = new ColorMatrix(
              new float[][]
             {
                 new float[] {.3f, .3f, .3f, 0, 0},
                 new float[] {.59f, .59f, .59f, 0, 0},
                 new float[] {.11f, .11f, .11f, 0, 0},
                 new float[] {0, 0, 0, 1, 0},
                 new float[] {0, 0, 0, 0, 1}
              });
```

The color matrix is in the RGBAW space, with the first vector of floats corresponding to the Red channel, the second to the Green channel, the third to the Blue channel, the fourth to the Alpha value and the fifth to a w value which is always 1. In this example, the alpha vector is also 1

since when transforming an image into grayscale we want to keep its initial opacity. The initial image's red, blue and green channels are being transformed in the following manner:

edited red vector= old red *.3f+ old blue*.3f+old green*.3f +0 +0;

edited green vector= old red*.59f+old blue*.59f+ old green*0.59f+0+0;

edited blue vector= old red*.11f+old blue*.11f+ old green*0.11+0+0;

Thus, taking equal amounts of the old RGB channels to create each of the new ones, transforms the image into grayscale. Similar color matrices were constructed in order to create the other filter effects. For cropping, resizing and zooming we used Visual Studio tool, which when having an image displayed could be set to stretch, scale or show only selected parts of the image. Some of the effects of the image editing tasks featured in our basic image editor are reflected in Figure 49.


[26]

Figure 49: Various editing tasks applied by our basic image editor to an uploaded image

5.2 Adding Gestures

After refining the traditional, WIMP based editor, we started thinking of ways in which gestural recognition was going to be incorporated. The first approach was trying to make use of the Kinect Toolbox; more specifically the gesture Learning Machine. The Kinect Toolbox is a set of useful tools for developing with Kinect for Windows SDK which includes helpers for gestures, postures, replay and drawing [27]. The gesture Learning Machine is a complex program which builds a statistical model based on multiple recordings of a gesture. These models can be saved to disk and reused later to recognize new gestures. The more instances of a certain gesture are recorded with the Learning Machine, the more the associated file grows in information. The Learning Machine is able to recognize slightly different cases of the same gesture and its accuracy increases with the size of the associated files. When having to identify gestures, the

Learning Machine extracts all gestural files and compares them with the gestures performed by the user. We wrote a program in C# with an integrated Learning Machine which was capable of being educated about specific gestures. We started teaching it a couple of gestures, by repeatedly performing each gesture in front of the Kinect sensors. After no more than five repeats for a gesture, the program could then reliably identify the gesture.

The problem with this approach was that the recognizer just provided a binary response of whether or not it had seen a gesture. When using the Learning Machine application one would have had to repeatedly create a gesture in order to keep editing an image according to a desired task. However, this is not what participants thought of when inputting their gestures. Users clearly relied on continuous, fluid gestures and not intermittent, repeated fragments of gestures. During our literature review, we looked at what Baudel and Lafon had to say about the issue of the segmentation of hand gestures. They highlighted the fact that a device that recognized gestures needed to be able to segment the continuous stream of captured motion into distinct lexical entities. As a result, the authors pointed out that most systems, such as the Kinect, identify steady positions also known as postures instead of dynamic motions.

Yet, all of the common gestures that we had identified were gestures in motion; the more a user performed a gesture, the greater the effects of the editing task were. For example, for increasing the effect of a filter, most participants kept raising their hand until they had reached the desired outcome. Repeated gestures or postures were therefore not a feasible approach since they were not capturing the movement, interaction and essence of the gestures generated by the participants.

5.3 Adding gestures- Second Approach

Another approach is one that uses mathematical equations which define the paths of all gestures meant to be recognized. Because using the Learning Machine proved to not be as successful since it involved gestural fragmentation, we started focusing on the second approach. Writing a program that would identify gestures based on whether the gestural path could be written as a specific mathematical function was more time consuming, but it did ultimately recognize more fluent gestures.

The program functions the following way: while the Kinect is tracking a gesture a user is inputting, each skeleton frame is sent into an algorithm that checks to see if it satisfies a mathematical equation. Every skeleton frame should differ from the previous ones according to a mathematical equation dependent on the gesture the user is trying to perform. For the left hand swipe gesture, which was already implemented by the Kinect Toolbox Gesture Viewer application, the mathematical equation was related to the way the left hand joints were supposed to move. For each frame, the left hand joints had to move along the x axis by a certain amount, and there was a very small threshold regarding how much the left hand joints could move in the y direction before the gesture was not considered a left hand swipe anymore.

Implementing some of the identified common gestures meant that we had to formulate similar equations to describe the image editing gestures. The first gestures that we started working with were increasing and decreasing the effect of an applied filter, since they both were one-handed gestures and both of them were outlining fairly simple paths.

Most participants used their right hand for the increasing the effect of a filter gesture which started by having the hand in the resting position by the hip, and then gradually bringing it up. The mathematical equation for this gesture is portrayed in the partial code below:

```
if (ScanPositions((p1, p2) => Math.Abs(p2.X - p1.X) < GestureMaximalWidth, // Width
                (p1, p2) => p2.Y - p1.Y > -0.01f, // Upwards progression
                (p1, p2) => Math.Abs(p2.Y - p1.Y) > GestureMinimalHeight, // Height
                GestureMinimalDuration, GestureMaximalDuration)) // Duration
```

The code looks at two points p1 and p2; p1 represents the position of the right hand on the previous frame and p2 denotes the position of the right hand on the current frame. By subtracting their x coordinates we get the width of the gesture at any given time during the execution of the gesture, which should be small since this is an upwards gesture not a sideways one. Therefore, we set a maximum variable `GestureMaximalWidth` which the difference of the x coordinates should not surpass. Then, we look at the upwards progression of the gesture by subtracting the two points' y coordinates. If the difference is negative then the gesture is progressing in the right direction. The gesture also needs to have a minimal height (`GestureMinimalHeight=0.4f`) in order to be considered a gesture at all, and it also needs to last a set amount of time (`GestureMinimalDuration= 250ms, GestureMaximalDuration=1500ms`), time that was set through testing and calculating what worked best with the gesture.

For the decreasing the effect of an applied filter, the most common gesture was the reverse of the increasing the effect of a filter motion, with the hand moving slowly downwards. Therefore the only change we had to implement with the mathematical equation was that the gesture had to

progress downwards rather than upwards and so when subtracting the two points' y coordinates (p2.Y- p1.Y) the difference had to be positive.

Integrating these mathematical equations with the Kinect skeleton tracking tool and the basic image editor described in Part A. 75, we were able to edit a pre-loaded image by increasing and decreasing its brightness using upwards and downwards hand gestures. While editing and testing we realized that we had to solve the problem of when the Kinect should start and end looking for gestures. Imagine this scenario: we only want to increase the brightness of an image. We raise the right hand; the brightness gradually increases as we keep raising the hand. Once we are done we bring our hand back down. What happens? The brightness decreases because bringing the hand down is correlated with decreasing the luminosity of the image. However, this is not what we intended to do, but the Kinect had no way of knowing that the gesture ended when the hand reached its final stretch up. Bringing the hand back down was not supposed to be interpreted as part of the motion.

5.4 Adding Voice Commands

As previously seen in Chapter 2: Literature Review, in the article "*Charade: Remote Control of Objects Using Free-Hand Gestures*" Baudel and Lafon talk about the "Immersion Syndrome", which is exactly what we encountered while testing the newly implemented gestures. Commonly, systems capture every gesture generated by a user, whether or not that motion was intended to be part of the end result. Baudel and Lafon thought that it was therefore important for a gestural device to provide well-defined means to detect the intention of the gestures.

Adding voice commands to clearly mark the start and end of a gesture was our proposed approach at creating a precise gesture detector prototype for image editing. Since the Kinect had a built in microphone array and voice recognition, we decide to incorporate these features into the foundations of our gestural image editor.

The Kinect Toolbox had a database of words that the Kinect could recognize. We selected the word “Record” to mark the beginning of a gesture, and the word “Stop” to signify the end of a gesture. We included these options into our C# program and whenever Kinect identified “Record” we linked that word with the part of our algorithms which started looking for gestures. When Kinect recognized “Stop”, we discontinued the process of looking for gestures (the skeleton frames were not sent for processing anymore).

After implementing the voice recognition, we were able to have Kinect only look for gestures whenever that was intended. Increasing and decreasing the effect of a filter could be executed precisely, without having anything else happen to the picture when a hand accidentally moved.

5.5 Future challenges

While increasing and decreasing the effect of a filter were one-handed gestures, some of the common gestures for the ten editing tasks we had in the main study were two-handed motions. In order to be able to identify the simultaneous motion of two hands, we started working on a new algorithm, which instead of looking at only the right hand joints, loaded the whole skeleton and then selected both left and right hand joints. This is a work in progress, but we are confident that with the right mathematical equations (for example for the rotating an image to the right, the

equation would be that of a circle) we could incorporate the two-handed gestures with the image editor. However, very complicated gestures, that for example involve crisscrossing of the arms or that are fairly irregular, could be almost impossible to code for since a corresponding mathematical equation would be very hard to identify.

Implementing two-handed gestures was not the only problem that required further work. As discussed in Chapter 4, some of the most common editing gestures were the same for different tasks (e.g. making an image half the size and zooming out). Therefore, when correlating them with a gesture recognition image editor, there would have to be a method of distinguishing between tasks. One way of addressing this problem would be to initially have a list of all tasks next to the image that requires editing. Users could then scroll through the list of tasks (using the gesture we identified to be most common for navigating menus) and select the editing task they wish to perform, e.g. zoom out. Using voice commands users could signal they are ready to begin gesturing and similarly that they have finished with that particular editing task. Therefore, when generating the gesture which involves extending both arms on each side of the body and progressively bringing them closer together, there would be no confusion whether the motion was meant to be a zoom out or half the size gesture.

Chapter 6

CONCLUSION

This research project started with the idea of exploring natural user interfaces (more specifically those based on gesture recognition) and discovering a way of making the user-technology interaction as instinctive as possible. The approach was to use gesture elicitation and ask participants in our study to create gestures which would match the given image editing tasks.

The process of having users create their own gestures rather than providing them with a set of previously assembled gestures proved to have successful results. We were able to identify common gestures for all ten image editing tasks, which signified that participants exhibited a similar way of thinking and correlating tasks with gestures during the study. More generally, since the pool of users was varied both in terms of gender and work background, our study offered insight into how gesture elicitation could be effectively used to generate common gestures for more types of applications and not just image editors.

6.1 Interpreting results

6.1.1 The image editing tasks

While we identified the most common gestures for each of our ten image editing tasks, not all of them were as equally popular. The most generated gestures for tasks such as rotating an image,

making an image half the size, zooming out, cropping a landscape, decreasing the effect of a filter and navigating through a menu list, are indeed very popular since more than half of our participants created them (also in multiple instances).

However, these editing tasks have the more strongly prevailing gestures for different reasons. Zooming out and navigating through a menu list are two tasks constantly integrated with most image editors. Therefore, when asked to generate gestures for these tasks, participants probably already had an informed idea of how their motions would look. Frequently, zooming out is represented as a pinch motion within the world of touch editors and during our study we found the most common gesture to be a pinch like motion of the arms. Also, shifting through the items of a menu list is represented by up and down swipes, something that was replaced by up and down hand movements during our study. This strengthens our theory related to the fact that the gestures for both zooming out and navigating through a menu list were influenced by pop culture.

For example, popular movies such as Minority Report where Tom Cruise executes up and down and left and right hand swipes to navigate through menus, widespread products such as the iPhone which is advertising the “pinch” finger motion as a way to zoom and enlarge images, all these cultural phenomena have influenced our thinking. This is portrayed in the results of our study, where the most common gesture for zooming out was the “pinch” motion of the arms. However, the gesture was executed using the whole hand and not just the fingers (a result probably related to the fact that users could not use just their fingers since Kinect did not have that kind of tracking accuracy). The most common gesture for navigation was using the hand as a

way of tracing a path to get to the desired item, similar to the way Tom Cruise and a lot of other characters in commercials and movies have performed it. Therefore, the more users had previously encountered comparable tasks and situations, the more it seemed possible that they would think of the same gestures and words to describe that task.

A puzzling result is that zooming in was a lot less popular than zooming out (5 users created the most common gesture as opposed to 10 for the zooming out task). Maybe as the study progressed, the users got more creative or became more aware of the pop culture influences and thus produced more similar gestures for zooming out, a task that followed the zooming in one.

Some of the other more popular gestures (for the rotating and decreasing the effect of a filter tasks) seem to owe their commonalities to the fact that their descriptions match the actual physical change. These editing tasks are not as regularly incorporated in the everyday image editor, yet participants created similar gestures to describe these tasks. One of the reasons could be the fact that the words “rotating” and “decreasing” were more revealing and helped users associate gestures with the given tasks more easily.

The task that did not have a very strong common gesture was cropping a portrait, with only 5 users creating it and no repeated instances. This is perhaps because the task of cropping was more complex and difficult to execute through gestures than other editing tasks. Usually, within the context of touch or traditional interfaces, cropping a portrait involves using a selection box to denote the area that needs to be cropped. However, when thinking of this task in the context of gestural interfaces, participants were not provided with a selection box. They either had to create

their own selection box through gestures, or generate a complete new gesture and therefore idea regarding how cropping should be represented through motions. This led to a multitude of gestures being generated by our 15 participants and finding an overall common gesture was more difficult than with any other editing tasks.

Both navigating through a list and a grid menu had similar most common gestures which ultimately described the path that one should take in order to get to a desired item. When selecting the ten image editing tasks, we were unsure whether we would find similar gestures for our two menu navigation tasks that would be independent from the shape of the menu. However, we did identify analogous gestures, which proves that participants were rationally thinking and correlating tasks and were not simply just generating gestures at random.

The users who created the most common gestures alternated and cycled the list of all fifteen participants in a fairly even fashion, all with the exception of participant 8. This made participant 8 be an even more obvious outlier. She did however produce two gestures in common with the other users, but surprisingly these gestures were describing two of the most difficult editing tasks that had some of the least common gestures: cropping a portrait and a landscape. With a background in dance, there was no doubt that this participant was very creative and outgoing in her gestures, so maybe the two cropping tasks which required a lot of imagination really suited this user's resourcefulness.

6.1.2 The skeleton tool

When designing the testing environment, we added the skeleton tracking tool as the sole feedback that users had when performing various gestures. While we think that including this feature helped our participants and enhanced their confidence (section 4.6), actually studying whether the skeleton tracking tool had a major positive impact could be the subject of further detailed investigation.

6.2 Applying results

As mentioned in previous chapters, image editing was chosen as a foundation for our study due to its ever increasing popularity. However, our research may well transition from image editing and into different fields since most of the tasks that users were prompted to generate gestures for could apply to many different environments. For example, navigating through a menu is a feature that a lot of applications could include if they were to be translated into NUI experiences: games, online shopping, web searching are just a few instances that would benefit from introducing the navigation gestures that we found to be the most common.

Considering further research, our study could be expanded into different disciplines where participants' gestures could be compared to the most common ones presented in this paper. Moreover, while we considered implementing voice commands as a way of marking the beginning and end of a gesture, different methods could be studied and applied depending on which prove to be the most efficient and popular with users. For example, very specific postures could be used to portray the start and finish of a gesture; timing could also be incorporated to

determine whether a change in the gestural path has not occurred for a given amount of time, after which the gesture is considered to be over.

With computing becoming ubiquitous, the discovery of ways of improving and easing human-computer interaction was placed at the core of our research. Technology that seemed purely fictional a few years ago is not only turning into something real but also accessible. Looking once again at the software behind the film Minority Report which has Tom Cruise moving through and organizing videos on a large screen using only hand gestures, we can see this "alien" like product breaching into our market. Oblong is actually implementing this interface as a way to sift through large amounts of video and other data [28]. In terms of accessibility, some of these applications are now tangible due to the introduction of the Kinect by Microsoft product which offers an endless list of opportunities in terms of development and research. Having something so compact and affordable has allowed scientists to really expand and explore natural user interfaces, an opportunity we seized with our study.

Looking ahead, we are excited about the possible ways in which our study could be continued and built upon. Eliciting and implementing a whole new array of gestures for different tasks, experimenting with voice and possibly other means of natural interaction, introducing finger recognition; these are all interesting, challenging tasks which could lead to new insights. "Ubiquitous computing is viable-and will soon be commercially practical," stated William Mark, SRI's vice president of Information and Computing Sciences in 2001. "The revolution is about to happen." [29] Twelve years later, the computing dream William Mark was talking about is

rapidly entering the realm of reality.

Bibliography

[1] B. Lee, P. Isenberg, N. H. Riche and S. Carpendale, "Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions".

[2] K. Hinckley, K. Yatani, M. Pahud, N. Coddington, J. Rodenhouse, A. Wilson, H. Benko and B. Buxton, "Pen + Touch = New Tools, Proc. UIST," 2010.

[3] "Apple - iPhone 4S," [Online]. Available: http://www.apple.com/iphone/features/siri.html.

[4] "Nuance - Dragon Dictation: iPhone - Dragon Dictation for iPadTM, iPhoneTM and iPod touchTM is an easy-to-use voice recognition application," [Online]. Available: http://www.nuance.com/for-business/by-product/dragon-dictation-iphone/index.htm.

[5] [Online]. Available: http://www.d-sidegroup.com/ge-gestures-frankfurt-airport/.

[6] W. A. Pike, J. Stasko, R. Chang and T. A. O'Connell, "The Science of Interaction," 2009.

[7] P. I. N. H. R. a. S. C. Bongshin Lee, "Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions".

[8] R. Francese, I. Passero and G. Tortora, "Wiimote and Kinect: Gestural User Interfaces add a Natural third dimension to HCI".

[9] Beaudouin-Lafon and T. Baudel, "Charade: Remote Control of Objects Using Free-Hand Gestures".

[10] J. Epps, S. Lichman and M. Wu, "A study of Hand Shape Use in Tabletop Gesture Interaction," 2006.

[11] P.Wachs, M. G. Jacob and Juan, "Context-based Hand Gesture Recognition for the Operating Room," [Online]. Available: http://www.purdue.edu/newsroom/releases/2013/Q1/surgeons-may-use-hand-gestures-to-manipulate-mri-images-in-or.html.

[12] M. A. Cheetham, E. Legge and C. M. Soussloff, "Editing the Image: Strategies in the Production and Reception of the Visual," *University of Toronto Press.*

[13] "BBC News Technology," [Online]. Available: http://www.bbc.co.uk/news/10569081.

[14] S. Jobs, "iPhone Presentation at the Macworld Conference & Expo," 2007. [Online]. Available: http://www.youtube.com/watch?v=Etyt4osHgX0.

[15] P. Wellner, "Digital Desk," 1991. [Online]. Available: http://www.youtube.com/watch?v=S8lCetZ_57g.

[16] M. r. lab, "DiamondTouch," 2001. [Online]. Available: http://www.merl.com/areas/DiamondTouch/.

[17] [Online]. Available: http://book.zi5.me/book/2473/OEBPS///httpatomoreillycomsourceoreillyimages829531.png.

[18] [Online]. Available: http://fc09.deviantart.net/fs13/f/2007/066/b/d/Landscape_by_baranyai.jpg.

[19] [Online]. Available: http://book.zi5.me/book/2473/OEBPS///httpatomoreillycomsourceoreillyimages829529.png.

[20] [Online]. Available: http://a8.vietbao.vn/images/vn888/hot/v2011/best_20120326-093842-3-hinh-3-18.jpeg.

[21] [Online]. Available: http://www.deviantart.com/art/Elisabeth-337220200.

[22] [Online]. Available: http://vannilaby.deviantart.com/art/Illuminated-330159505.

[23] [Online]. Available: https://www.flickr.com/photos/peterix/8175460341/in/photostream.

[24] Y. Rogers, H. Sharp and J. Preece, "Interaction Design: Beyond Human-Computer Interaction," 2007.

[25] "Pixel Samples," 11 November 2010. [Online]. Available: http://en.wikipedia.org/wiki/File:PixelSamples32bppRGBA.png.

[26] Lostoid, "Close your eyes and see," 2012-2013. [Online]. Available: http://www.deviantart.com/art/Close-your-eyes-and-see-337193329.

[27] "CodePlex Project Hosting for Open Source Software," [Online]. Available: http://kinecttoolbox.codeplex.com/.

[28] R. Lever, "Minority Report software hits the real world," July 2012. [Online]. Available: http://www.theage.com.au/digital-life/computers/minority-report-software-hits-the-real-world-20120724-22ln3.html.

[29] R. Buderi, "MIT Technology Review: Computing goes everywhere".

[30] "Alpha compositing," 20 February 2013. [Online]. Available: http://en.wikipedia.org/wiki/Alpha_compositing.

[31] "Lossless Compression," 7 3 2013. [Online]. Available: http://en.wikipedia.org/wiki/Lossless_data_compression.

[32] M. Corporation, "ColorMatrix Class," [Online]. Available: http://msdn.microsoft.com/en-us/library/system.drawing.imaging.colormatrix(v=vs.71).aspx.

[33] M. Corporation, "Introduction to Kinect for Windows Audio," 2013. [Online]. Available: http://www.microsoft.com/en-us/kinectforwindows/develop/tutorials.aspx.

[34] A. Danielescu, S. Drucker, D. Fisher and m. s. Meredith Ringel Morris, "Gestures for Faceted Browsing: Findings from a Creativity Elicitation Approach".