A Quantitative Quality Model for Gesture Based User Interfaces

Kayne Barclay, Danny Wei
Software Engineering
University of Auckland
{kbar099, dwei023}@aucklanduni.ac.nz

Christof Lutteroth, Robert Sheehan
Computer Science
University of Auckland
{lutteroth@cs.auckland.ac.nz,
r.sheehan@auckland.ac.nz}

ABSTRACT
The technological advancement of computers and cameras over the past few years has given us the ability to control objects without touching them. There have already been a number of attempts at producing gesture based applications, but many of them have usability issues. This paper proposes a model that reflects the usability of a gesture based interface, in order to evaluate and improve a gesture-controlled system. The model defines four levels of abstraction, with the higher levels based on the lower ones. The levels of the model allow us to propose quantitative notions for 1) the parameters affecting the quality of individual gestures, 2) the overall quality of a gesture, 3) the quality of particular functionalities, or use cases, in a system, and 4) the overall quality of a system. The model was evaluated using an existing gesture-based interface for a popular media center application.

Author Keywords
Model, quality, measurement, gesture-based interaction

ACM Classification Keywords
HCI

INTRODUCTION
Imagine a world twenty years from now where home appliances can be controlled with just a single wave of the hand, without touching anything. The idea of using hand gestures to control objects and systems has been around for a long time, but never really took off due to equipment cost and technological limitations. However, consumer-level hardware such as the Xbox Kinect is now available to support such an endeavour. All that is lacking is the software that harnesses this technology's potential.

One of the most obvious applications for the Kinect is to control a television or a media centre. In the scenario of a media centre, instead of using a remote control, the user's body can be used instead. There would be no more searching for the remote as the remote is always at hand - quite literally. Using the user's own body to control computer systems has been described as more intuitive and direct (Stern et al., 2006). Furthermore, there are many situations that require the use of computer systems where it is difficult to use a computer, possibly due to dirty hands, or distance between the user and the input peripherals. The ultimate goal of gesture based systems is to allow for contactless input, thereby overcoming these problems. This could help integrate computer systems further into day-to-day living.

There have already been a number of attempts at producing gesture based applications, but many of them have usability issues. One question that requires further investigation is what factors contribute to the quality of a gesture based interface. If these factors could be identified, the quality of a gesture based interface could potentially be measured. In this paper we are investigating the following questions:

1. Which factors determine the quality of a gesture?
2. How can we create a quantitative model that reflects the usability of a gesture based interface?
3. How can such a model be used to identify usability issues?

To answer question one, we performed a literature review to see what factors were suggested by other researchers. Based on these suggestions, some more factors were identified from our own observations during an empirical study. To answer question two, we investigated various functionalities, or use cases, of an existing gesture based interface, KinEmote, which allows users to control the popular XBMC media center with the Kinect. Starting with one of the functionalities, playing a video, we constructed a state based model that describes the transitions in the system that are caused by the gestures of the user. This modeling approach was refined by modeling two more functionalities of the media centre: viewing images, and playing music. An empirical analysis of several hundred gestures recorded on video and a questionnaire were used to determine quantitative parameters for this model. To answer question three, we defined quality measures on different levels of abstraction. These measures allow interaction designers to analyze the individual parts of a gesture based user interface, as well as getting an overall notion of the quality of a system. If a system is of poor quality, a designer can drill down in the model to identify the issues that limit the quality of the overall system.

The structure of the paper is as follows. The next section discusses related work. Following this section, the factors that contribute to the quality of a gesture, and how they could be measured, are discussed. Then, our quantitative

OZCHI '11, Nov 28 – Dec 2, 2011, Canberra, Australia

model with its four levels of abstraction is introduced, and we discuss how the model can be used to identify "bottlenecks", i.e. parts of an interface that need improvement. After this, evaluation results from a gesture-based application to control a media centre, KinEmote, are presented.

RELATED WORK

In order to identify the factors determining the quality of a gesture and gesture based interfaces in general, related work about the factors involved in gesture interaction were analyzed. According to Lenman et. al (2002) hand gestures serve three functional roles:

1. Semiotic: The ability to communicate information
2. Ergotic: The ability to manipulate objects in the real world
3. Epistemic: Allows learning from the environment through tactile experience

They also categorize design factors for gestural commands into 3 dimensions: cognitive aspects, articulatory aspects, and technological aspects. Cognitive aspects refer to how easy a command can be learnt and recalled. Articulatory aspects determine how easy it is for a user to perform gestures. For example, gestures involving complicated hand or finger poses should be avoided because for some people they might be impossible to perform.

According to Baudel and Lafon (Baudel et al., 1993; Fukumoto et al., 1992) gestures should provide fast and reversible actions and not require much precision, in order to avoid fatigue. Stern, Wachs, and Edan (Stern et al., 2006; Stern et al., 2004) explored if there was a way of guessing if a gesture is intuitive and comfortable to the user beforehand without testing. From the results there is only evidence that gestures which relate to the function they invoke in a system are likely to be considered natural and intuitive. Freeman and Weissman (1995) experimented with the idea of using hand tracking as a buttonless mouse to select objects, so that the familiar point-and-click semantics of the mouse can be used.

The main advantage to gestures is the natural interaction between the user and the computer, by eliminating the need for the intermediate devices. This can lower the cognitive aspect of the user, and potentially makes the system easier to learn. Gesture control also allows for a more powerful interaction, where a single gesture can define both a command and its parameters. The main disadvantage is the problem of inputting unintentional commands, by accidently performing a gesture. One solution to this problem is having locks on gesture recognition, to prevent unintentional input of commands. Another disadvantage is that gesture based interfaces by nature require the users to recall rather than recognize the gestures. This problem can be mitigated by having a good choice of gestures that relate to the tasks that can be performed in the system.

In the study of North et al. (2009) a comparison of mouse and touch interfaces was made. A number of similar tasks were performed using both types of interactions, and the task completion time was recorded. The tasks were, for example, clicking and selecting a file, or clicking and moving a file. The result obtained by North et al. was that by using the touch interfaces, the tasks were being completed faster than using a mouse, but slower than physical interaction.

Forlines et al. (2007) compared the speed and accuracy of performing unimanual and bi-manual tasks when using touch interactions. In their experiments, a single-touch display or a single mouse was used for unimanual tasks. For bimanual tasks, two mice or two fingers on a multi-touch display were used. The results indicated that for unimanual tasks, the mouse was faster overall, whereas for bimanual tasks the dual touch was better and preferred.

Nielson et al. (2003) identified two different approaches for measuring intuitiveness: a bottom up approach and a top down approach. The bottom up approach gets the participants to match each gesture to a function, and the top down approach matches each function to the correct gesture. From the matching results a measure for intuitiveness is calculated.

DETERMINING GESTURE QUALITY FACTORS

In order to successfully construct a usable model, the factors that contribute to the level of quality of an arbitrary gesture need to be determined. With the help of related work about gesture based interfaces and the addition of our own observations, a set of factors was decided upon.

Fatigue

Fatigue seems to be one of the most important factors when it comes to measuring the quality of a gesture. The level of fatigue affects the usability of a gesture by decreasing its appeal to the user. If a user finds a particular gesture to be taxing, they will not enjoy the experience of using the gesture, and therefore be inclined to discontinue the use of the gesture, or even the system if the gesture is essential.

Unfortunately, it is difficult to measure fatigue, as it is a subjective variable. However, there are measurement instruments from sport science that are used to measure physical exertion. In sport science, the exertion to be measured is typically a lot higher than one would encounter for a gesture, but the methods can be adapted. A common empirical method is to monitor a user's heart rate, and compare the resting heart rate with the heart rate right after a task. We would measure a participant's heart rate, then ask the participant to perform a given gesture a set number of times or for a certain duration, and finally measure the heart rate again. The duration needs to be long enough so that the physical exertion is measurable. The relative change in heart rate Δh after n seconds is calculated as follows:

$$\Delta h = \frac{h_n - h_0}{h_0} \qquad (1)$$

where h_n is the heart rate of the user after performing a gesture for n number of seconds, and h_0 is the initial heart rate of the user before performing the gesture.

As a limitation, using the heart rate as a measure may pose a threat to internal validity: it is hard to make sure that an increase of heart rate is due to the performance of the gesture. The change in heart rate for gestures is not as significant as it would be for typical sports exercises; hence the measurements are more susceptible to uncontrolled factors in our case. Heart rate is affected by a multitude of factors, e.g. excitement. Furthermore, measuring the heart rate can be inconvenient for participants as heart rate monitors require electrodes to be attached to the skin.

Another common method for measuring exertion in sports science is the Borg CR10 scale (Borg, 1998), where the user rates the level of fatigue of a given gesture. This scale is a proven instrument where participants are asked to rate their exertion on a scale from 0 to 10, with standard labels ensuring relatively equidistant scale intervals. For our purposes, it is necessary to extend the scale slightly by adding additional fractional values in the lower range (e.g. 0.5, 1.5, …). The exertion caused by gestures, even when repeated a number of times, is comparatively low. With these adaptations the Borg CR10 scale is a convenient, non-intrusive tool that was used in our experiments as the preferred way of measuring fatigue.

Naturalness

The naturalness of a gesture is a great contributing factor towards the effectiveness of a gesture, as it directly affects the cognitive aspects of the user, and how memorable the gesture is (Lenman et al., 2002). If a gesture is not sufficiently memorable, more time is spent trying to recall it, resulting in less efficient usage of the system. Naturalness is often achieved through the use of symbols (Stern et al., 2006; Stern et al., 2004), i.e. making the shape of scissors for the "cut" function, or selecting something with a push movement.

The naturalness quality attribute influences the time spent on remembering the gesture to perform a particular function, before actually performing it. However, this think time is also influenced by the time spent on choosing a function in the user interface. For example, an ill-designed user interface may cause long think times due to its poor navigational structure, and not due to unnatural gestures. This makes it hard to measure naturalness purely from user session recordings.

There are methods that were proposed for measuring the intuitiveness of a gesture (Nielson et al., 2003), which could be interpreted as a measure for naturalness. However, by naturalness we refer not only to the ease of recall as is mostly referred to by intuitiveness, but also to other subjective factors such as how "natural" a gesture is perceived by a user when performing it. Hence, it was decided that naturalness would be measured using a 5-point Likert scale, with 1 being "very natural" and 5 being "very unnatural" (Stern et al., 2006; Stern et al., 2004).

Gesture duration

The time that is spent performing a gesture also affects the quality of the system. The more time is spent performing a gesture, the less throughput can be achieved, i.e. the less tasks can be completed within a certain time. Hence, gesture duration affects the efficiency of the overall system. We measured gesture duration by recording user sessions and measuring the time between the start and the end of a gesture. This is done by manually reviewing and taking samples from video recordings.

Accuracy

The accuracy of a system with regard to a given gesture is the probability that the system recognizes the gesture correctly. The accuracy of a given gesture is a good indicator of both its difficulty, and its uniqueness. If a gesture is not unique enough, i.e. similar to another gesture, it will be easy to unintentionally carry out the wrong gesture, or to cause the system to misinterpret the gesture and perform the wrong function.

To calculate the accuracy of a gesture, recorded user sessions are analyzed, and the number of correctly and incorrectly recognized gestures is counted. The number of correctly recognized gestures is divided by the total number of performed gestures of the given type. A gesture is only correctly recognized when the correct corresponding function is performed by the system in response to the user's intended gestures, i.e. if a system performs a function on an unintended gesture then we flag that as an incorrect recognition of the user's gesture. In our experiments, we obtained accuracy measurements by manually reviewing and taking samples from video recordings that included both a screen recording of the system and a video recording of the user.

A QUANTITATIVE MODEL FOR GESTURE BASED SYSTEMS

In order to effectively compare two gesture based interfaces, the measurements taken for the abovementioned gesture quality factors need to be meaningfully combined, taking the navigational structure of a system into consideration. To make sense of the numbers, it is important to clearly indentify the functionalities of a system, and relate quantitative quality measures to these functionalities. By functionalities we mean groups of functions that are necessary for a particular use case. For example, a functionality "playing a video" would encompass functions for selecting, starting and pausing a video, among others.

To properly compare two systems, only the functionalities that they have in common can be compared. If two systems do not have common functionalities, then it is impossible to effectively compare them. This does not mean that the functionalities need to have the same user interface in both applications. In fact, it is exactly the differences in the user interfaces that we intend to quantify.

In the case of our gesture based media centre scenario, the primary functionalities that have been chosen for analysis are: playing a music file, playing a video file, and viewing a picture. In order to obtain some actual data, an empirical study was carried out, using the methods proposed in the previous section. How this raw data is used in our approach to build a model is described in the following.

Abstraction 1: Quality factors of a gesture

Using the data obtained from an empirical study, normalized values for each quality factor of each gesture are calculated. First, the measurements are aggregated by calculating averages. Then, the averages are normalized by using standard scores (also known as z-values). Standard scores are an established statistical method for representing measurements relative to their distribution, considering the distribution average and the standard deviation. In the following, this will be illustrated for our gesture quality factors.

Fatigue

$$Fat_{avg} = \frac{1}{N}\sum_{i=1}^{N} Fat_i \tag{2}$$

where the fatigue values from the Borg CR10 scale of N users are $Fat_1, Fat_2, \ldots, Fat_N$. To normalize the fatigue average of a given gesture, we calculate $\widehat{Fat_{avg}}$ as the inverse z-value of the naturalness average. That is, we negate the average due to its inverse proportional relationship with quality, and then we relate it to the mean $\overline{Fat_{avg}}$ and the standard deviation σ of the distribution of the fatigue averages of all gestures:

$$\widehat{Fat_{avg}} = -\frac{Fat_{avg} - \overline{Fat_{avg}}}{\sigma} \tag{3}$$

Naturalness

$$Nat_{avg} = \frac{1}{N}\sum_{i=1}^{N} Nat_i \tag{4}$$

where the naturalness values from the Likert scale of N users are $Nat_1, Nat_2, \ldots, Nat_N$. Similar to fatigue, we normalize this value by relating the naturalness average of a gesture to the distribution of the naturalnesss averages of all gestures:

$$\widehat{Nat_{avg}} = \frac{Nat_{avg} - \overline{Nat_{avg}}}{\sigma} \tag{5}$$

where $\overline{Nat_{avg}}$ represents the average of all the gestures' average naturalness values, and σ their standard deviation.

Duration

$$Dur_{avg} = \frac{1}{N}\sum_{i=1}^{N} Dur_i \tag{6}$$

where the gesture duration values of N observations of that gesture are $Dur_1, Dur_2, \ldots, Dur_N$, obtained from observing the start and end times of gestures in a video recording of participants. Similar to fatigue, the duration quality value $\widehat{Dur_{avg}}$ uses the inverse due to the inverse proportional relationship of duration with quality:

$$\widehat{Dur_{avg}} = -\frac{Dur_{avg} - \overline{Dur_{avg}}}{\sigma} \tag{7}$$

where $\overline{Dur_{avg}}$ represents the average of all the gestures' average durations, and σ their standard deviation.

Accuracy

$$Acc = \frac{N_{correct}}{N_{total}} \tag{8}$$

where $N_{correct}$ represents the number of times across all users that the given gesture is correctly recognized, and N_{total} the total number of attempts for that gesture. Similar to the other factors, this value is normalized using standard scores:

$$\widehat{Acc} = \frac{Acc - \overline{Acc}}{\sigma} \tag{9}$$

where $\overline{Acc}$ represents the average of all the gestures' accuracy values, and σ their standard deviation.

Abstraction 2: Overall quality of a gesture

Next, we calculate an overall quality value for each gesture. The higher this value is, the higher the quality. Firstly, the relationships that each of the quality factors from abstraction 1 have with this overall quality value must be defined. All the previously defined factors are proportional to quality, however, each of these factors potentially hold a different importance for the overall gesture quality. We express this by describing the overall quality as a weighted average of the individual quality factors of a gesture. The importance of each factor is reflected in its weight.

We propose to obtain the weights through a questionnaire, which asks participants to estimate the importance of given factors as percentages. The averages of these weightings across all participants is used. Hence, the overall quality of a gesture is defined as follows:

$$GestureQuality = w_{Dur} \times \widehat{Dur_{avg}} + w_{Fat} \times \widehat{Fat_{avg}} + w_{Acc} \times \widehat{Acc} + w_{Nat} \times \widehat{Nat_{avg}} \tag{10}$$

where w_x represents the average weight of a quality factor x.

Abstraction 3: Quality of a functionality

For each functionality, a state transition model such as the one shown in Figure 1 is constructed in order to gain a clearer insight into the interaction a user needs to perform a particular use case. This model represents the gesture paths necessary for performing particular tasks. Each rounded rectangle node in the state transition model represents a state of the UI, and each transition represents a gesture. Note the state transition model for the locking-unblocking feature of KinEmote: a special state set notation is used to express that locking may occur from any of the other states in the model.

After the construction of this state transition model, a typical interaction path for a functionality is identified. This is done by observing what users typically do when using a functionality such as playing a video, and identifying a path in the state transition model that reflects this behavior. For example, in the case of playing a video, a user needs to navigate from the main menu to the video menu, then navigate through files and folders, and finally select a video file. The idea is to model the average scenario, so ideally the typical path is modeled with knowledge about typical users. For example, knowledge about the typical file-folder structure and its size should be used, or otherwise well-founded assumptions need to be made. In our example, the number of files and folders influences the number of gesture repetitions while browsing for a particular video.

The typical path is used to quantify the quality of a functionality. This quality value is obtained by calculating the average quality value for all the gestures in the typical path. The following formula represents this calculation:

$$FunctionQuality = \frac{1}{N}\sum_{i=1}^{N} GestureQuality_i \qquad (11)$$

where N is the total number of gestures in the typical path, and $GestureQuality_i$ represents the gesture at position i of the typical path. The more often a gesture occurs on the typical path, the stronger its influence on the quality of the functionality.

Abstraction 4: Quality of a system

To obtain an all-encompassing quality value for a system, we need to combine the quality values for its functionalities in a way that factors in their relative importance. This is done by calculating a weighted average, with the importance of each functionality reflected in its weight. The relative importance of a functionality depends on the user; therefore we suggest to use a questionnaire that lets users of a particular group estimate the relative importance of a functionality in percent. In many cases, the relative frequency that a functionality is used (compared to other functions) is a reasonable estimate for its importance. However, the relative frequency can only be measured precisely by collecting real usage data, which is harder to do. To sum up, the quality of a system can be modeled as

$$SystemQuality = \sum_{i=1}^{N}(w_i \times FunctionQuality_i) \qquad (12)$$

where N is the total number of functionalities being considered, $FunctionQuality_i$ is the quality of the ith function, and w_i is its relative importance. Note that this value can only be used to compare two systems if they are equivalent with regard to the considered functionalities, i.e. if they can be used to perform the same tasks, although with different user interfaces.

IDENTIFYING BOTTLENECKS – USING THE MODEL

As one carries out the process that is laid out above, information is gradually combined and thus abstracted. By looking at the constituents of the quality values, one can come to conclusions about the usability bottlenecks in a given system. Depending on the level of abstraction an interaction designer wants to deal with, they can consult the corresponding level of abstraction in the model and identify the values that limit the quality of a system. Using the abstractions described in the previous section, a designer can identify functional paths with high impact but low quality value. For functionalities of poor quality, they can look at the abstraction below to identify frequent gestures in that path with low quality values. To look even further into poor quality gestures, the features of those gestures that cause them to be of a low quality can be identified on the lowest level of abstraction.

EVALUATION

In order to evaluate the proposed modeling approach, a model for the various functionalities of an existing gesture based interface, KinEmote, was created. KinEmote allows users to control the popular XBMC media center with the Kinect. We chose three functionalities (playing a video, viewing images, and playing music) and created models for them on all the levels of abstraction. We recorded two participants for a total duration of 35 minutes, using the functionalities to complete typical tasks (such as selecting and playing a song). Both the screen, the Kinect input and the whole setup (using an external camera) were recorded.

Gesture	Quality value
Move up (mu)	0.22
Hold up (hu)	0.17
Move down (md)	0.91
Hold down (hd)	0.51
Select (s)	-0.15

Table 1. Gesture quality values.

Five different types of gestures were necessary to carry out the tasks, which are listed in Table 1. To determine the duration of each gesture type, we measured the duration of 10 gestures per participant per gesture type

	Hold up	Move up	Hold down	Move down	Hold left	Move left	Hold right	Move right	Select
Number incorrect	54	16	28	3	12	4	18	15	8
Total	116	34	116	35	35	23	33	20	22
Accuracy	0.53	0.53	0.76	0.91	0.66	0.83	0.45	0.25	0.64
Average accuracy value	0.62								
Variance	0.04								
Standard deviation	0.20								
Normalized accuracy quality factor (z-value)	-0.41	-0.44	0.69	1.46	0.19	1.03	-0.80	-1.81	0.09

Table 2. Accuracy data.

using the video recordings. To determine the accuracy, a total of 434 gestures were analyzed. The relative importance of the gesture quality factors and the functionalities were measured using a questionnaire. A state transition model was created and the measurements were used to calculate the quantitative model parameters.

	Selection	Up	Down	Left	Right
Average	5	3	3	3.5	3.5
Average fatigue value	3.6				
Variance	0.68				
Standard deviation	0.82				
Normalized fatigue quality factor (- z-value)	-1.70	0.73	0.73	0.12	0.12

Table 3. Fatigue data.

	Selection	Up	Down	Left	Right
Average	2.5	2.5	2.5	1.5	2
Average naturalness value	2.2				
Variance	0.2				
Standard deviation	0.45				
Normalized naturalness quality factor (z-value)	0.67	0.67	0.67	-1.57	-0.45

Table 4. Naturalness data.

	Move down	Hold down	Move up	Hold up	Select
Average:	0.55	0.60	0.24	0.34	0.44
Average duration value (in seconds)	0.43				
Variance	0.02				
Standard deviation	0.15				
Normalized quality factor (z-value)	-0.76	-1.13	1.31	0.64	-0.05

Table 5. Duration data.

Results

We followed the levels of abstraction that are specified in the previous section, using the measurement methods proposed in this paper. The results for the first level of abstraction, the quality factors for each gesture, are shown in the Tables 2-5. For the second level, the quality factors were combined according to the average weighting obtained from the participants to form an overall quality value for each gesture, as shown in the Tables 1 and 6.

Duration	0.1	Accuracy	0.475
Fatigue	0.2	Naturalness	0.225

Table 6. Average weightings for the quality factors.

Following on to the third abstraction, quality values for the functionalities under investigation were calculated. This was done by summing up gesture values according to a functionality's typical path, into an overall quality value which is shown in Table 7. Lastly, the quality values for each functionality were combined according to the average weighting assigned by the participants, as shown in Table 8.

	Music	Pictures	Videos
Total gestures:	30	16	77
Typical path:	1mu + 1hu + 5s + 3md + 20hd	1mu + 2hu + 3s + 2md + 8hd	3s + 2md + 72hd
Typical path average quality value:	0.42	0.38	0.50

Table 7. Functionality quality values.

Overall system quality value equation:	(0.2 + 0.5)/2 * m + 0.2 / 2 * I + (0.5 + 0.6) / 2 * v
Evaluation:	0.46

Table 8. Overall quality value for the investigated parts of the system.

Discussion

For the purposes of this discussion, only one function will be considered, as all investigated functionalities have very similar quality values. Since the Videos function has the highest weighting value, any improvements on this will result in a higher increase on the overall quality value.

Looking at the Videos function in abstraction three, it can be seen that the "hold down" gesture is the most frequently used one, being 72 out of the 77 gestures of the typical path. Therefore, the quality of this gesture would affect the quality value of the functionality the most. To further assess this gesture, the second level of abstraction for the hold down gesture must be examined.

Looking at abstraction level two, it can be seen that the "hold down" gesture is not as good as its "move down" counterpart. This further proves the importance of improving this gesture, because of its frequency in the typical path, and the fact that its quality value is not as good as it could be. These two aspects, in collaboration, contribute to the poor quality value of the Videos functionality. Hence, when improving the quality value for Videos, the "hold down" gesture should be the primary focus.

Additionally, we can further focus the improvements by identifying the quality factors that cause the "hold down" gesture to be of such a low value. We can see from looking at the data that two of the quality factors greatly affect the quality value of "hold down". Firstly, the naturalness is only 2.5 (normalized to 0.67). This value is not particularly good, and so this would have an adverse effect on the gesture's overall quality value. Secondly, the duration time is the highest for all the gestures analyzed. As the quality factors use standard scores (z-values), which are dependent on the overall distribution of values, this results in a poor value for duration (-1.13).

The other functions of the system, and their contained gestures, can also be improved with this tool. This is achieved in the same fashion as shown above; by iterating from the fourth level of abstraction, down the chain of abstractions to gradually expose more information regarding the causes of poor quality. Based on the quality values as the abstractions are traversed, it is possible to effectively prioritize where most of the improvements should be made.

Limitations

Throughout the data collection process, a few problems arose, the first of which was the time constraints. Because of the lack of time in which to carry out this data analysis, we were unable to fully analyze all of the data. Because of this constraint, we were forced to take samples of the data instead. Unfortunately, this was only realized after a fair proportion of the data had been analyzed, and so it was decided that the samples we took would be taken from this. However in hindsight, this could cause a misrepresentation of the participant's full test run, because of the lack of experience early on in the test. This could possibly result in a threat to external validity.

When analyzing our questionnaires, we came across unexpected answers regarding the naturalness of different gestures. For some gestures we thought they were quite natural, however, the participants found them to be unnatural. This may mean that our approach for measuring naturalness may need to be refined, e.g. by improving the instructions and the wording of the questionnaire. In the future, we intend to include short descriptions of each quality factor in the questionnaire rather than just naming them or explaining them verbally, to avoid misunderstandings.

Additionally, when observing the test participants, it was realized that the position in which the interface was unlocked (with a wave gesture), determined the origin of further gestures. This has potentially affected our results adversely, as it affected how the quality of the gestures was measured, resulting in a threat to internal validity.

While observing the test participants, it was noticed that when a participant was scrolling that they often overshot their goal. This could be interpreted as another quality factor that has not been considered by us yet, e.g. "controllability". The fact that overshooting was generally considered an accuracy problem meant that accuracy could potentially be misrepresented, therefore possibly resulting in a threat to internal validity.

In addition to these threats to validity, we were also faced with problems associated with our hardware. Firstly, the screen capture video software only provided video at a rate of 10 frames per second. Because of this, multiple gestures would often occur between two frames, making it extremely difficult to accurately measure the duration of gesture instances. Secondly, our external video camera did not have a strong enough microphone, and so we were not able to analyze all the comments of the participant while they were using the system.

CONCLUSIONS

During our study of gesture base interfaces, we identified important factors contributing to the quality of a gesture. These factors, as well as suggestions of how they can be measured, were presented. We proposed a model that combines measurements for quality factors into meaningful information about a system, taking into account the navigational structure of a gesture based interface and using four different levels of abstraction. This includes a quality measure for individual gestures, the functionalities of a system, and a system itself. The quantitative model helps us to understand the quality of a gesture based system, to identify the causes of low quality values, and hence improve the system by addressing quality bottlenecks.

To the best of our knowledge, the proposed quality model is the only quantitative model for gesture based user interfaces so far. As for all models, it is necessary to collect more data about this new approach in order to better understand its limitations and benefits. The study presented here is a first step in this direction.

REFERENCES

Baudel, T., Lafton, M. B. CHARADE: Remote Control of Objects using Free-Hand Gestures. Commun. ACM-Special issue on computer augmented environments: back to the real world, (1993), Vol. 36, 28-35.

Borg, G., Borg's Perceived Exertion and Pain Scales, Human Kinetics (1998).

Forlines, C., Wigdor, D., Shen, C., Balakrishnan, R. Direct-Touch vs. Mouse Input for Tabletop Displays. Proc. CHI 2007, ACM PRESS (2007), 647-656.

Freeman, W. T., Weissman, C. D. Television Control by Hand Gestures. IEEE Intl. Wkshp. on Automatic Face and Gesture Recognition, (1995), 24-31.

Fukumoto, M., Mase, K. and Suenaga, Y. Finger-pointer: A Glove Free Interface, CHI'92 Conference Proceedings, Poster and Short Talks booklet,(1992), 62.

Lenman, S., Bretzner, L., Eiderback, B. Computer Vision Based Recognition of Hand Gestures for Human-Computer Interaction. Proc. of the second Nordic conference on HCI (NordiCHI '02), (2002), 239-242.

Lenman, S., Bretzner, L., Thuresson, B. Using Marking Menus to Develop Command Sets for Computer Vision Based Hand Gesture Interfaces. Proceedings of the Second Nordic Conference on Human-Computer Interaction, (2002), 239-242.

Nielson, M., Storring, M., Moeslund, T. B., and Granum, E. A procedure for developing intuitive and ergonomic gesture interfaces for man-machine interaction. Technical Report CVMT, (2003), 03-01.

North, C., Dwyer, T., Lee, B., Fisher, D., Isenberg, P., Robertson, G., Inkpen, K. Understanding Multi-touch Manuipulation for Surface Computing. Lecture Notes in Computer Science 5727, (2009), 236-249.

Stern, H. I., Wachs, J. P., and Edan, Y. Hand gesture vocabulary design: a multicriteria optimization. IEEE SMC 2004, Inter. Conf. on Systems, Man and Cybernetics, (2004), 10-13.

Stern, H. I., Wachs, J. P., Edan, Y. Human Factors for Design of Hand Gesture Human -Machine Interaction. IEEE Intl. Conf. on Systems, Man and Cybernetics (SMC '06), (2006), 4052-4057.

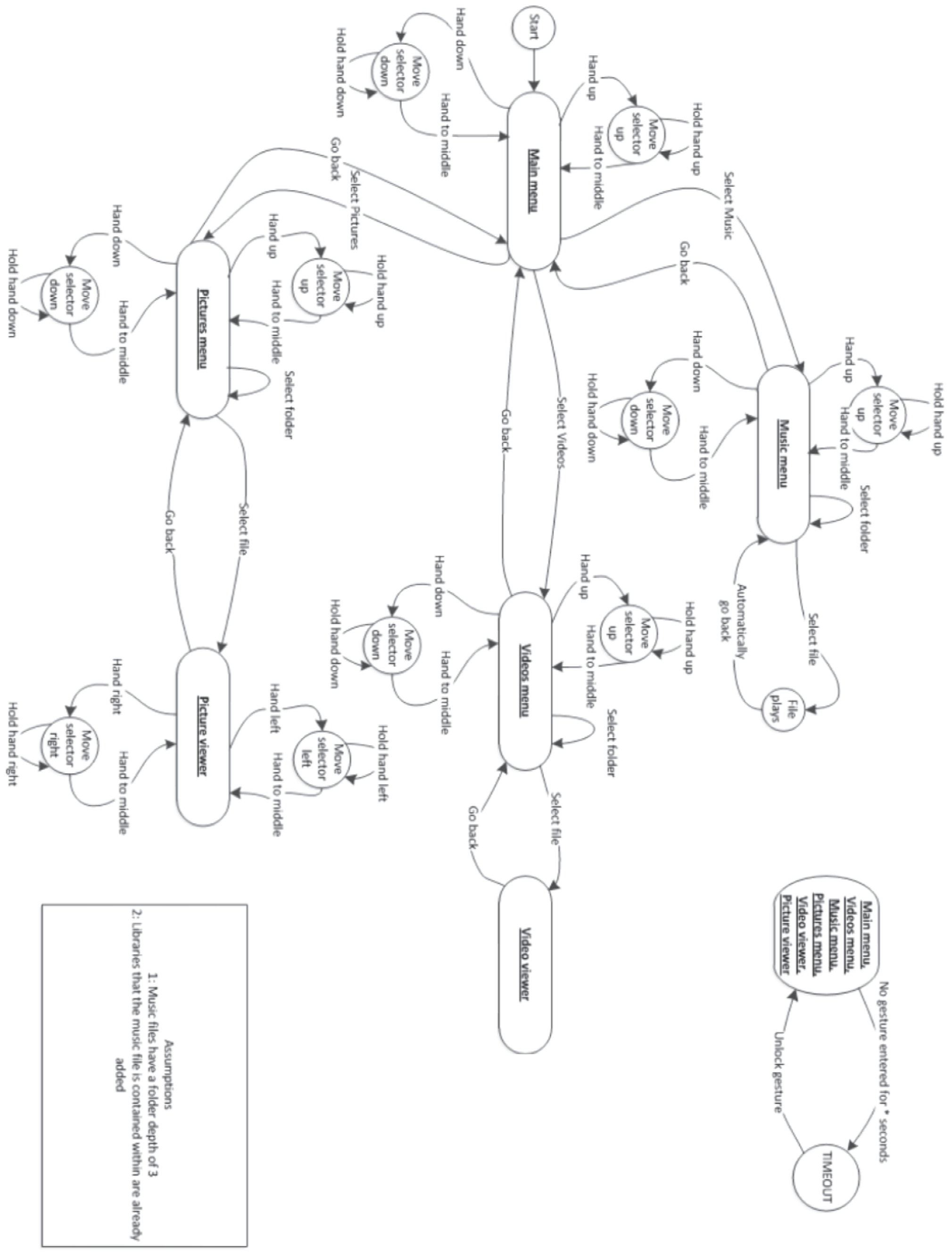

Figure 1. State transition diagram for the XBMC system with the KinEmote interface.