Institute for Visualization and Interactive Systems

University of Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Masterarbeit Nr. 75

Investigating the Characteristics of Unistroke Gestures using a Mobile Game

Johannes Wolf

Course of Study:	Softwaretechnik
Examiner:	Jun.-Prof. Dr. Niels Henze
Supervisor:	Dipl.-Inf. Sven Mayer, Dipl.-Inf. Dominik Weber, Dipl.-Inf. (FH) Valentin Schwind
Commenced:	December 1, 2015
Completed:	June 1, 2016
CR-Classification:	H.5.2

Kurzfassung

Touch-Gesten sind momentan die am weitesten verbreitete Eingabemethode für die Interaktion mit Smartphones. Insbesondere Unistroke-Gesten, welche aus einem einzelnen Strich bestehen, werden für die Texteingabe mittels Touch-Keyboards und als Shortcut für Funktion auf mobilen Geräten verwendet.
Im Rahmen dieser Arbeit wird die Genauigkeit untersucht, mit der der Nutzer Unistroke-Gesten auf mobilen Geräten ausführt. Zu diesem Zweck wurden zwei Studien durchgeführt, die die Fähigkeit des Nutzers untersuchen, verschiedene vorgegebene Gesten wahrzunehmen und diese zu reproduzieren.
Eine Kontroll-Studie zielte darauf ab zu untersuchen, wie genau der Nutzer Gesten wiedergeben kann, die aus einzelnen geraden Linien oder aus zusammengesetzten geraden Linien bestehen wobei die Winkel und die Längen dieser Linien in dem Experiment variiert wurden.
Um die Ablenkung durch äußeren Einflüsse im gewohnten Nutzerumfeld zu untersuchen, wurde eine größer angelegte Studie mittels eines Mobile-Games durchgeführt. Das Spiel wurde so entworfen, dass der Nutzer motiviert war die vorgegebenen Gesten möglichst genau durchzuführen. Basis für das Spielprinzip war die Kontroll-Studie, bei der die Probanden die Gesten wahrnehmen und reproduzieren mussten. Um eine große Menge an Touch-Proben zu sammeln wurde das Spiel in einem App-Store für mobile Geräte veröffentlicht und die Ergebnisse über das Internet eingesammelt.
Die Analyse beider Studien (Kontroll-Studie und Spiel-Studie) ergab, dass die Rotation und die Winkel innerhalb einer Geste eine Auswirkung auf die Genauigkeit der Ausführung haben. Dieser Effekt wurde an Variationen in der Biegung, der Winkel und der Form der Gesten beobachtet. Dabei wurden die Gesten von den Nutzern des Spiels (d.h. in der Spiel-Studie) ungenauer durchgeführt als in der Kontroll-Studie.

Abstract

Touch gestures are today's main input method for the interaction with smart phones. In particular, unistroke gestures, which are gestures consisting of one articulated line, are commonly used for text input (touch keyboards) and short cuts referring to functions on mobile devices.

This work investigates the user's accuracy of articulating unistroke touch gestures on mobile devices. Therefore, two studies were conducted which focused on the user's ability to perceive different variations of unistroke gestures and reproduce them accurately. First, a control study aimed at analyzing the user's touch accuracy during the articulation of single line and composed line gestures by varying gesture properties like the rotation or angles within the gestures. To analyze the influences of the user's distraction in his natural environment, a large scale study was conducted which used a mobile game as apparatus. The game was designed in a way that the user was motivated to articulate the given gestures precisely. The game principle was based on the control study procedure of perceiving and reproducing gestures. To gather a great amount of touch samples the game was published on a mobile app store and the samples were collected trough the internet.

The analysis of both studies showed that the gestures orientation and the angles within the gestures affected the articulation accuracy in terms of the deviations made by bending, rotating, and varying the shape of the gestures. Furthermore, the gestures articulated in the game study tended to be more error-prone compared to those being articulated in the control study.

Contents

List of Figures

List of Tables

List of Acronyms

AGATe Agreement Analysis Toolkit

CLC Curves, Line Segments, and Corners

CSD Corner Shape Distance

DTW Dynamic Time Warping Algorithm

GECKo Gesture Clustering Toolkit

GHoST Gesture Heatmaps Toolkit

GRANDMA Gesture Recognizers Automated in a Novel Direct Manipulation Architecture

GREAT Gesture Relative Accuracy Toolkit

MDS Multidimensional Scaling

NASA-TLX NASA Task Load Index

1. Introduction

Nowadays we see a shift from personal computers to the use of more and more mobile devices as a main computing device. All major mobile devices are using touch input as the main input technique. However, even the latest generation of laptops are often equipped with touch screens.

The interaction through unistroke and multistroke touch gestures is becoming more important in the area of human computer interaction. This type of interaction gives the user a shortcut to the functions of his touch device like a smartphone or a tablet computer in an often intuitive way. These functions are for example launching an app from the home screen or using functions inside a particular app. Also, touch keyboards which enable text input by doing stroke gestures are getting more popular.

Previous work in the context of stroke gestures has examined different kinds of stroke gesture recognition algorithms. Depending on the use case, there are algorithms that cover the recognition of unistroke and multistroke gestures in a directed and undirected way. Further research has shown which factors have to be considered when stroke gesture sets are designed. These factors are for example the memorability of the gestures, the modality of the user feedback and the overall complexity of the gesture sets. Studies have shown the human accuracy and performance for articulating the gestures depending on the gesture complexity. The researchers analyzed the difference between the different input methods like thumb, index finger and pen in terms of the gesture completion time and accuracy. They also examined the consistency between users and within users when performing the gestures.

However, little research has been done in understanding how the gesture characteristics relate to the articulation accuracy of the user. For instance, such characteristics are the number of corners or the rotation of a gesture.

The objective of this work is to investigate how these characteristics affect the articulation accuracy of the user. By doing two studies, we like to answer questions like: Is the user able to distinguish if a gesture is rotated by 15° degrees and reproduce it? Thereby we focus on unistroke gestures which consist of straight lines.

For this purpose we conduct two studies. A large scale study which uses a mobile game as apparatus and a control study which used a visual less distractible version of the game. In order to get a large number of participants, the game is released to a mobile app store. By comparing both studies, were able to see how the distraction of the user affects his articulation accuracy.

The basic procedure for both studies is the same. The participants is shown a sequence of gestures which has to be reproduced by them. In the game study the participants get rewarded by animations and scores for accurate gestures to stay motivated.
By varying the gestures in their orientation, number of lines, and the angles within the gestures we are able to analyze how these gesture properties relate toe the articulated accuracy.

Structure

This work is structured as follows:

Chapter 2 – Related Work: This Chapter presents the current state of unistroke gesture research with focus on the user behavior and gesture recognition.

Chapter 3 – Analyzing the Articulation Accuracy of Unistroke Gestures: This Chapter presents the focus of the two performed studies and the methodology of the analysis.

Chapter 4 – Control Study: In this Chapter, the control study apparatus, procedure, and the results are described.

Chapter 5 – Game Study: This Chapter presents the concept of the large scale game study and the results of the analysis.

Chapter 6 – Study Comparison: In this Chapter we compare and discuss the results of the large scale study and the control study.

Chapter 7 – Conclusion: This Chapter summarizes the two performed studies and the findings. Also, this Chapter gives an outlook for future work in the scope of this work.

2. Related Work

This Chapter gives an overview of previous investigations in touch gestures with unistroke gestures on the main focus. First, Section 2.1 reviews descriptions of stroke gestures in form of stroke gesture features and models. Next, Section 2.2 shows all aspects related to the gesture production process which includes perception, cognition and motor control. Taking these aspects into account, Section 2.3 will look at the gesture design process including user-defined gestures. From the technical point of view, gesture recognizers enable applications the use of gesture interaction. The different strategies of stroke gesture recognition algorithms are shown in Section 2.4. Last, Section 2.5 focuses on previous experiences made with large scale studies using app stores like applied in this work.

2.1. Gesture Models

To describe and understand the properties of stroke gestures, metrics and models have been created in previous work. This Section will look at these investigations. Thereby we first describe gesture features and metrics and then predictive models for the gesture complexity.

2.1.1. Gesture Features and Metrics

The common representation of a stroke gesture is an ordered list of coordinates. Additionally, the list can be extended by a timestamp and the pressure for each coordinate (cf. Zhai and Kristensson [ZK12]). However, to understand the gestures properties in detail and analyze the user's behavior when articulating gestures, there's the need for further descriptions. In the context of stroke gesture research, these metrics are called gesture features.

Rubine [Rub92] defined a set of gesture features which are used for gesture recognition. These features are for instance the sine of the initial angle, the angle of the bounding box, the sharpness or the size of the bounding box. Long et al. [LLR99] extend the set of features amongst other things by the curviness and the aspect.

In contrast to the previously reviewed gesture metrics, the work of Vatavu et al. [VAW13] introduces a set of relative gesture metrics. The relative gesture features differ from the absolute features in that they describe the accuracy of an articulated gesture in relation to a predefined gesture task axis. The task axis is a template-like representation of the gesture. This enables the comparison of gestures. Depending on the use case, the task axis can be described in three ways: A perfectly shaped gesture of primitives (geometric), an average gesture based on several user samples (average) or the closest user sample to the average gesture (template). The relative metrics describe the articulated gesture in terms of geometric accuracy, kinematic accuracy and articulation accuracy. The geometric accuracy is the geometric deviation from the task axis as well as the errors in length, size, or bending. The kinematic accuracy describes the deviation in production time and speed. The articulation accuracy describes the count and the order of strokes when performing multistroke gestures (also described by Anthony et al. [AVW13]). The metrics were applied to a dataset of articulated gestures with the result that gestures performed in medium speed had a smaller shape error than fast and slow articulated gestures. A further outcome was that the length and the bending error increases with the articulation speed.

2.1.2. Gesture Complexity Models

A simple model to estimate the production time of unistroke characters was presented by Isokoski [Iso01]. The model is based on the assumption that each straight line has a similar production time. Each character can be approximated by a simple representation consisting of only lines. The resulting production time of the character is accumulated by the sum of the articulation time for the characters' lines. The model was chosen deliberately simplistic to be easily applicable and not specific for the Roman alphabet. The model provides an accuracy of $0.85R^2$ by reason for the simplicity.

An approach to estimate the gesture production time is the Curves, Line Segments, and Corners (CLC) model presented by Cao and Zhai [CZ07]. The model predicts the production time depending on the count of curves, line segments, and corners. Each gesture is decomposed into these three types of segments. The production time for each of these segments is described by a sub-model. The sum of all sub-segments describes the estimated production time for the gesture. The model was evaluated in a study in which the participants used a stylus as an input device. Except for familiar gestures (like letters) and individual differences between the participants, the model delivered a high correlation to the empirical data (greater than $0.9R^2$). Additionally, the study showed that diagonal lines between 45° and 225° were the fastest but also more likely to be error-prone. However, the model only concentrates on the human motor system and doesn't consider the mental demand when the gestures are performed.

In contrast to the CLC model, Vatavu et al. [Vat+11] described a model to rate unistroke gestures into five groups of execution difficulty (75% accuracy) and rank them (90% accuracy) with two simple rules. The authors ran a study in which the participants had to articulate a set of gestures multiple times. Afterwards, they had to rate the difficulty of the gestures and rank them. Their study showed that the perceived complexity highly correlated with the production time for familiar and unfamiliar gestures as well. Surprisingly, the attempt to use the CLC model for estimating the production time in this context delivered poor results with 67% for the rating and 28% for the ranking. On the other hand, the prediction model of Isokoski [Iso01] could be used to predict the ranking with an accuracy of 82%. Furthermore, the evaluation showed that only a few participants (1-5) are necessary to get a good accuracy for the prediction (five participants lead to an accuracy of 91.3% in ranking and 79.8% in rating).

2.2. Gesture Production

The process of gesture production includes the perception of gestures, cognitive aspects like the learning of gestures, and motor aspects when executing the gestures. This Section will cover all these aspects relevant to gesture production.

2.2.1. Gesture Perception

To understand the visual similarity of pen gestures, Long et al. [Lon+99] studied the user's perception of gestures and built a model to predict visual similarities between gestures. Therefore, a study was performed which analyzed different gesture sets. Each gesture set focused on different gesture features used by the Rubine gesture recognition algorithm (cf. Rubine [Rub92]). A gesture set with widely varying features, a set with changing absolute angle and aspect, a set with changing gesture length and area, a set with changing rotation of the gestures, and a combination of all of them were tested. The participants were shown three-gesture-blocks where they had to decide which gesture differed the most from the other ones. The analysis showed that the judgements for similarity differed among the participants. Surprisingly, the length and the bounding box size of the gestures had no significant influence to the perceived similarity of the gestures. An analysis Multidimensional Scaling (MDS) showed five compressed dimensions of visual similarities. For instance, the dimension angle/distance correlated with the gesture of curviness. Furthermore, regression analysis showed that the angle of the bounding box, as well as the alignment to the x-axis and y-axes, are significant features to determine the gestures similarities. The resulting model is able to predict the perceived similarity with a correlation of 0.56 with reported user judgements. The model was also applied in a tool which supports gesture designers to find similarities while designing gesture sets.

2.2.2. Cognition

An important aspect in gesture design is the memorability of gestures. Zhai and Kristensson [ZK03] analyzed the memorability and the learning effect of stroke gestures. For this reason, participants had to learn gestures which represent words on a gesture keyboard. The study consisted of four testing sessions to analyze the learning effect. The average number of newly learned gestures was around 15 and nearly constant per session.

Appert and Zhai [AZ09] made a study where they compared the memorability between keyboard shortcuts and gesture shortcuts. During the study the participants were pleased to select menu items either by using the mouse, keyboard shortcuts or gesture shortcuts. The experiment was repeated a day after and the participants had the freedom to use their preferred input method. The experiment showed that the gesture shortcuts had a significant advantage to the keyboard shortcuts in production time, error rate, and the ability to recall the gestures. Moreover, about 77% of the participants preferred the gesture input on the second day.

Further investigations in terms of memorability and the guessability of gestures have been done in relation to user-defined gestures which will be reviewed in Section 2.3.2.

2.2.3. Motor Control and User Behaviour

To understand the differences and similarities between pen and finger stroke gestures Tu et al. [TRZ15] compared the different input methods. The authors compared the input with the index finger, the thumb and a pen. To characterize and compare the gestures, the authors used previously mentioned features like articulation time but also the aperture between the starting and the ending point, the indicative angle between the start point and the centroid as well as the shape distance. They also designed two new features: The size ratio between the bounding boxes and the CSD, which describes the geometric distance of the gesture's corner points. This is done by extracting the points of the corners, resampling them to a fix count and building the average to all point distances. A similar technique is also used by the $-unistroke recognizer (See Section 2.4). During the study, the participants had to articulate a set of gestures under different conditions. In each experiment, they used a pen, the index finger and the thumb. The experiments were executed on a stationary device, a mobile device in a sitting position, and a mobile device while walking. Because of the mobile focus we consider the latter two cases. In most experiments the pen input performed the best while the index finger gestures performed better than the thumb. This is explained by the inclusion of more muscles and joints when using a pen than a single finger. In sitting position the drawing implement had a significant effect on the articulation time: the fastest input was produced by the pen followed by the index finger and at last the thumb. Furthermore, the pen gestures tended to be smaller than the finger gestures but there was no significant difference between the index finger and thumb gestures in size. The finger gestures also tended do be less accurate than the pen gestures with regard to the shape distance and the CSD. One exception is the indicative angle difference which was similar for all implements in each experiment. Another outcome relevant to this master thesis was a significant bias in the gesture's orientation by about 3°. The last experiment, in which the participants walked while entering the gestures, proved it to be a benefit when the thumb was used as an implement compared the thumb's performance in sitting position: The reduction of the production time in relation to the sitting experiment was the smallest at the expense of accuracy. But the overall performance of the thumb input was still the worst.

Investigations in understanding the consistency within and among users have been done by Anthony et al. [AVW13]. The authors focused on multistroke gestures and their production features like stroke directions as well as the execution variations expressed in geometric and kinematic descriptors. With a correlation of 0.91, the authors discovered a high consistency within users. Additionally, certain gesture types like less geometrically complex ones showed a high consistency. The findings of this study resulted in a set of design guidelines to support gesture designers. Furthermore, the authors developed a tool called Gesture Clustering Toolkit (GECKo) which clusters similar gestures and provides several visualizations to understand the users' gesture articulation.

2.2.4. Feedback on Gesture Articulation

In the following, we'll have a look at the impact of different feedback modalities according to the gesture articulation. Depending on the use of the visual, acoustic or haptic feedback, there are significant influences in the resulting gesture.

Andersen and Zhai [AZ08] analyzed the influence of visual and acoustic feedback by doing a user study. The participants were pleased to practice gestures under changing conditions in terms of the visual and acoustic feedback. The visual feedback was implemented by visualizing a digital ink while the acoustic feedback was realized by playing sound patterns. The sound patterns provided information about the changes in position, direction, and speed. This was done by mapping the vertical movement to the sound volume and the vertical movement to the direction of the sound source by using stereo sound.

Focusing first on the visual feedback, the analysis showed that articulated gestures with visual feedback were significantly smaller. The average production speed and the aperture on closed gestures was smaller compared to no visual feedback. On the other hand, the visual feedback had a small impact to the completion time and no influence to the shape distance as well as the directions of the gesture segments. Hence, the authors recommend to not use the latter two features as critical recognition information when designing gesture sets.

The acoustic feedback aims to provide easy to learn and remember informational sound patterns to support the user during gesture production. However, participants were able to remember gestures better from visual feedback than from acoustic one. Like the visual feedback, the acoustic feedback had no significant impact on the shape distance. Also, the other features behaved similar to the visual feedback but to a much smaller extend. The authors argue that the feedback during gesture production may to slow to affect the user's behavior because the gestures are rather drawn from memory.

Based on haptic feedback, Zahai et al. [ZK12] discuss approaches to support the user during gesture articulation. For instance, haptic feedback could provide anchor points on the touch screen. They also note that from observation dry fingers tend to move smoother on touch screens than moist ones.

2.3. Gesture Design

When it comes to the design of gesture sets for user interfaces, the previously shown findings have to be considered to create gestures that are easy to learn and execute . Like Vatavu et al. [Vat+11] argue, the design process can be supported either by predictive models like CLC or by involving users to the design process. The latter is recommended by the authors but needs more time.

2.3.1. Design Principles

Based on some previously reviewed studies and models, Zhai et al. [ZK12] summarized principles which should be considered during the gesture design process. However, the authors note that the principles can be conflicting and its up to the gesture designers to find a suitable trade-off between them. In the following these principles are described.

First, the authors recommend to "make the gestures analogous to physics or conventions" [ZK12]. As an example they explain that a question mark gesture could indicate the need of help. Another presented example is the use of text or single numbers and letters. There's a need for a bridge between the conventions which are indeed easy to remember but not designed for efficiency to the more efficient ones which are herder to remember. An approach for this is a modified version of the gesture alphabet like EdgeWrite. Also, the gestures should be "guessable and immediately usable" [ZK12] which makes them immediately usable. For this reason, the users have to be included in the design process. On the other hand, it should be considered that users tend to vary when defining gestures for particular functions. Third, the gestures should be "as simple as possible" [ZK12] in terms of motor complexity. Therefore, predictive models like CLC can be used to predict the complexity. Fourth, designers should "make [the] gestures distinct" [ZK12]. This leads to a better recall rate and gives a better recognition accuracy than the use of similar gestures. Gesture design tools can support the gesture designers to indicate similarities between gestures (see Section 2.3.3). On the other hand, Long et al. [LLR99] argues that similar gestures can be beneficial for the memorability if they are used for similar functions. This leads to the next Principle. "Gestures [should be] systematic" [ZK12] so that the conceptual model is clear to the user. The authors show a word-gesture-keyboard as an example for a mnemonic to link words to gestures. This makes the "gestures [also] self-revealing" [ZK12] as the keyboard is constantly shown. Other self-revealing approaches are for example to show "crib sheets" [ZK12] on demand or further, to guide the user if he hesitates. Furthermore, the interface designers have to "support the right level of chunking" [ZK12] which is explained as the grouping of tasks. As a named example, the user could either enter a phrase with a single gesture or word by word using more gestures. While the first method has its benefits by a reduced motor effort, the second method has an increased motor complexity paired with reduced cognitive effort and fewer constraints. Last, the interface designers have to "support progression" [ZK12] from an easy to learn interface (visual-based) to an efficient interface (recall-based). Interfaces of customer products are often designed for the ease of learning which has drawbacks in the efficiency of the interface. On the other hand, efficient user interfaces which make the use of shortcuts are heavier to learn. It's up to the interface designers to find a moderate way to support this progression. The procession can be achieved by a consistent mapping between the stimulus and response.

An example for this is a system which supports gesture input (recall-based) and a guided gesture input (visual-based) [ZK12].

2.3.2. User-Defined Gestures

An alternative to pre-designed gesture sets is the use of user defined gestures. Here, the users are free to design their own gestures or are involved in the gesture design process which can have advantages in memorability of gestures. This was shown by Nacenta et al. [Nac+13], where they compared the memorability of user-defined gestures, pre-designed gestures, and random gesture sets. As a result, the user-defined gesture set had a higher recall rate even a day after the design process. They also found out that the error rates were a result of erroneous associations.

Wobbrock et al. [Wob+05] defined the guessability of gestures and offered a formal measure for this. The guessability is an important property of gesture sets when the user lacks of knowledge about the gestures. The authors argue that it's unrealistic to expect that the users have the intention and time to learn new gestures. The process for achieving a high gueassability of the gesture set is described as follows. First, the participants are shown a set of referents, which are the referencing functions to the gestures. The participants then have to define a gesture for each referent. After that, the gestures are linked to the referents if there are no conflicts. Conflicts are raised if the same gesture is linked to different referents. To solve the conflicts, a rating function determines which referent links to which gesture. The evaluation of this process resulted in an improved EdgeWrite alphabet which increased the guessability from 51% to 80.1%. Further research to solve conflicting gestures has been done by Vatavu et al. [VW15] which resulted in a toolkit to find agreements within user gestures.

On the other hand, Oh and Findlater [OF13] considered the challenges which come with the user design process. A major outcome was that users tend to create familiar gestures which they already know from their known touch interfaces like smart phones even if they had no constraints for the gesture design. Additionally, the study showed that users consider features like the finger count, stroke count, or the stroke order to distinguish between gestures. In contrast, features like speed, scale or pattern repetition seem to be less important for the distinction. Nevertheless, the author noticed that the results differed from another similar user study by Wobrock et al. [WMW09] where the users rarely cared about the number of used fingers. According to these results, Oh and Findlater [OF13] later recommend for a mixed-initiative customization supported for the design process. As an example, the users could be supported by recommended modifications of the gestures.

2.3.3. Tool Support

Several tools have been developed to support the gesture design and analysis. We'll give a short overview of the existing tools focusing especially on symbolic stroke gestures.

A first attempt to enable tool support in adding gesture interaction to applications was done 1991 by Rubine [Rub91]. The tool called Gesture Recognizers Automated in a Novel Direct Manipulation Architecture (GRANDMA) includes a unistroke recognizer described in the next Section 2.4 and was created to add easily unistroke gesture interaction to applications by using training data. Another tool which aims to add gesture interaction to existing Java user interfaces is the *Stroke Shortcut Toolkit* by Appert et al. [AZ09]. The tool enables interface designers to add gesture shortcuts as they define new gestures or manipulate a predefined gesture set. Furthermore, there are advanced options like visual feedback while gesture articulation as virtual ink or help functions for the user. While these tools are designed for easy to implement gesture interaction, there are further tools which offer additional metrics for the gesture sets. Long et al. [LLR01] developed a gesture design tool which warns the interface designer of similarly looking gestures. Using this information, the gesture sets can be optimized in terms of memorization and recognition. Singer et al. [SKN07] created a tool which supports not only the creation and testing of gesture sets, but also the function to test gesture sets with different recognizers to find a suitable recognizer for each gesture set.

In opposite to gesture design tools, there are tools which support developers by discovering and understanding how users articulate gestures. These tools are especially useful for exploring user gestures. In this area, Anthony and Vatavu developed a set of tools. First, the tool called GECKo by Anthony et al. [AVW13] clusters datasets of user gestures with respect to gesture production patterns like the stroke order of multistroke gestures. Next, the Gesture Heatmap Toolkit Gesture Heatmaps Toolkit (GHoST) by Vatavu et al. [VAW14] helps in understanding the user dependent and independent accuracy while gesture articulation. The tool creates heatmaps and provides accuracy additional metrics using Gesture Relative Accuracy Toolkit (GREAT) (cf. Vatavu et al. [VAW13]) developed by the same authors. Like mentioned in Section 2.3.2, there can be conflicts when designing guessable gesture sets. Agreement Analysis Toolkit (AGATe), a tool by Vatavu and Wobbrock [VW15], was designed to help resolving these conflicts by providing agreement metrics and statistical tests.

2.4. Gesture Recognizers

The algorithms for stroke gesture recognition can be divided into two approaches: On-line algorithms, which compute recognized gestures while articulation and off-line algorithms, which deliver the recognized gesture after articulation. As an example, in Kristensson and Danby [KD11], the authors present an on-line algorithm recognition algorithm which is also able to predict the users intended gesture. However, in the context of this master thesis, we will focus on off-line algorithms for unistroke gestures. We also exclude multi-touch gesture recognizers because they are outside the scope of this thesis.

One way to recognize gestures is done using feature extraction. Like Rubine [Rub92] described, the algorithm extracts features from a set of gestures and matches them with the features of an articulated gesture. Therefore, the recognizer has to learn the gesture's features by some training examples. An alternative to the feature-based algorithms are template-based algorithms. Here, the gestures are described by an ordered list of points. After the user has articulated the gesture, the recognition algorithm searches for the best match of the articulated gesture and the set of gesture templates. For instance, a lightweight and easy to implement (needs around 100 lines of code) unistroke recognizer of this kind is the $-Algorithm (cf. Wobbrock et al. [WWL07]). To find the closest match, the articulated gesture and the gesture templates are resampled with a fix count of samples, scaled to a square and rotated to the indicative angle. After that, the algorithm calculates the optimal angle, based on the Golden Section Search, to get an optimal score. Now the algorithm compares the gestures to their path distance and the template with the shortest distance is the result. This algorithm is rotation and scale invariant which means that the shape is recognized independently of orientation of the articulated gesture. Instead of comparing points of gestures to find the best match, the Dynamic Time Warping Algorithm (DTW) interprets the gestures as signals over time. This algorithm is also used in other research areas like speech recognition or general gesture recognition with cameras. An implementation of DTW by Holt et al. [HRH07] recognizes the gestures by applying a smooth filter and warps the signals of the articulated gesture for the best alignment to the template gestures. After that, the recognizes gesture is chosen by the nearest neighbour algorithm. Another unistroke recognizer that is also based on the nearest neighbour approach is called protractor presented by Li [Li10]. In contrast to the $-Algorithm, this recognizer can be rotation sensitive and rotation invariant implemented. Additionally, it's able to distinguish and recognize 1D gestures (lines) in contrast to the $-Algorithm. The shape distance of the articulated gesture and the compared gesture template is computed by the inverse cosine between the two vectors of the gestures points. For this reason, the magnitude of the vector's distance has no impact on the resulting shape distance. Comparing to the $-Algorithm, the protractor is significantly faster, uses less computational space and is more tolerant to gesture variation. To cope with variations of unistroke gestures, Lü et al. [LFL14] developed a recognizer with tool support. The recognizer combines the recognition of example-based gesture descriptions with a textual description of the gesture articulation so called gesture script. The gesture script describes the sub-components of the gestures as well as their order. Comparing to the previous recognizers, this one delivers additional attributes as a recognition result like the orientation of an arrow or the count of zigzags inside a spring symbol. A similar approach of describing the gestures in sub-components is used by the SiGeR-Algorithm (Simple Gesture Recognizer) by Scott Swigart [Swi05]. Here, the gesture is defined as an ordered list of directions and points where the velocity is nearly zero (stop points). The usage of the stop points is necessary to distinguish edges from curves. The directions are only expressed in a simple manner like “U” to move up or “LD” to move left and down. Thanks to the tool support, which extracts the gesture definitions by a set of example gestures, it is easy to add new gestures to the recognizer.

If a gesture consists of more than one stroke, so called multistroke gesture, other recognitions strategies are needed. Such extended recognitions strategies can also be used for unistroke recognition. One attempt, like the $N-algorithm does, is to permute all strokes of a multistroke gesture in the production order and direction. In a second step, an unistroke recognizer (here the $-Algorithm) is used to find the best match. Because the of its rotation invariance and its missing capability to recognize 1D gestures, the $-Algorithm has to be exstended. To recognize 1D gestures for example, the algorithm checks the aspect ratio and normalized the gesture in relation if it’s a potential 1D gesture. In addition, the algorithm has to do some speed optimizations because of the potentially huge amount of variations of the gesture. A significant improvement in recognition speed is reached by an improved version of the $N-Recognizer, which uses the $N-Protractor for the unistroke recognition. To reduce the combinatoric complexity of composed strokes, the $P-Recognizer by Vatavu et al. [VAW12] ignores the order of points and segmentation of strokes and instead interpretes the gestures as point clouds. So the algorithm is invariant of the stroke order and direction. The minimum matching distance then is derived from the assignment problem known from graph theory using the Hungarian algorithm. Therefore, the articulated gesture and the gesture templates are represented as bipartite graphs and the recognized gesture is the solution with the smallest weight. Concentrating on basic shapes, Fonseca and Jorge developed a gesture recognizer which is able to deal with variations in the line style. Thus, the gestures can be also drawn dashed and as bold lines. This is done by extracting features like the convex hull and the usage of fuzzy logic. Therefore, each gesture has to be defined by a training set of all line styles.

2.5. Research in the Large

Part of this work is the gesture data acquisition by doing a large scale. For this reason, we'll have a look at previous studies using an app store to publish an app apparatus.

Henze et al. [PH11] reported from their experiences regarding user studies which make the use of mobile app stores. Although the smart phone users, in their case especially Android users, are far from a perfect sample of the world population, the use of app stores enables the acquisition of huge datasets in the users real life context rather than isolating them which provides better external validity than lab studies. However, the researchers can not ensure the correct use of the apps. Therefore, the collected data has to be filtered by removing strokes which obviously are not related to intended gestures. For motivating the user, the app can be designed as a game with high score lists. The user's motivation is also relevant if the study investigates their long term behavior. For legal and ethic reasons, the users have to be informed that they participate a study which includes data collection. The best rate of collected data in relation to the app installations was achieved when informing the user at the first start of the app. While the collected quantitative data in the presented apps was mostly useful (except for the investigation of long term behavior of a navigation app), the collected qualitative data was nearly useless. An approach to get meaningful qualitative date is to reward the user with in-game badges. Like commercial apps, the study apps aim to reach as many users as possible. Henze and Boll [Hen11] discovered that the best time to publish a game app is Sunday evening because at that time the fewest apps get released to the google play store. Taking this into account, the authors expect to get 4 times more installations as the app is listed at the most recent apps in the store.

2.6. Summary

This Chapter presented the research in relation to unistroke gestures which was relevant to this master thesis. Models like CLC help in estimating the gestures complexity expressed in production time. The gesture production process includes the perception and memorization of gestures. In this context, including the users into the design process can have a positive effect on the memorability and guessability of gestures. Furthermore, there are several gesture design principles, which should be considered by gesture interface designers when designing user-friendly interfaces.
The user's accuracy is affected by the used implement like (Pen, index finger or thumb) and whether the user is sitting or walking. From the technical point of view, gesture recognizers like the $-recognizers are needed to add gesture interaction to applications. However, depending on the gesture set, a suitable recognizer has to be chosen to fit the

applications needs. For this the recognizer's limitations like rotation invariance have to be considered.

Previous work in relation to large scale studies using app stores showed benefits and challenges using this study method. If possible, designing the study as a game can have its advantages in the user's motivation and long term usage using high scores. Furthermore, this kind of study investigates the users' behavior in its natural environment.

While there were several investigations to predict the complexity and production time of unistroke gestures, few research has been done in order to predict the articulation accuracy. To make the users' gesture production accuracy predictable, there's the need to analyze their touch behavior and which factors affect the articulated accuracy. Therefore, this work aims to analyze the articulation accuracy of line gestures and how the distraction of users affects their touch behavior.

3. Analyzing the Articulation Accuracy of Unistroke Gestures

This work aims to analyze the accuracy in which users are able to reproduce proposed unistroke touch gestures on mobile devices. Understanding the accuracy of the gesture production (including visual perception and motor control) is necessary for the creation of robust gesture recognizers and the design of well usable gesture sets.
Therefore a large scale study and a control study is performed. The large scale study uses a mobile dance game as apparatus where the users have to articulate several unistroke gestures. If the gestures are performed precisely, the player is rewarded by a graphical dance animation. The users of the game are in their natural environment which includes external influences. The control study uses a changed version of the game apparatus which excludes the graphical background as well as the feedback for accurately performed gestures. Further, the users participating the control study are in a controlled environment which includes their sitting position and using only the thumb as implement.

Comparing the two studies gives insight in how the environment and external influences affect the accuracy of the users' touch behavior for unistroke gestures. Additionally, the gestures performed in these experiments are varying in the following characteristics: Orientation, angles within the gestures, number of stroke segments, and aspect ratio. Analyzing the effect of varying these characteristics (also called features in the following) with respect to the gesture articulation accuracy shows how these features affect the users touch accuracy.
The result of the analysis shows the weakness and strength in terms of the user's articulation accuracy. In particular, the results present the users' touch accuracy with respect to the external environment and the gestures' features. Figure 3.1 illustrates the investigation focus by comparing some example gestures used by the two studies.

This Chapter presents how the gestures of the two studies are analyzed in detail. Starting with Section 3.1, the focus of the studies analysis and the varying gesture features are described. Next, Section 3.2 defines hypotheses for the users touch accuracy depending on the changing conditions which are basis for the discussion in Chapter 5 and Chapter 4. Last, Section 3.3 describes the analysis methodology of the gesture analysis including

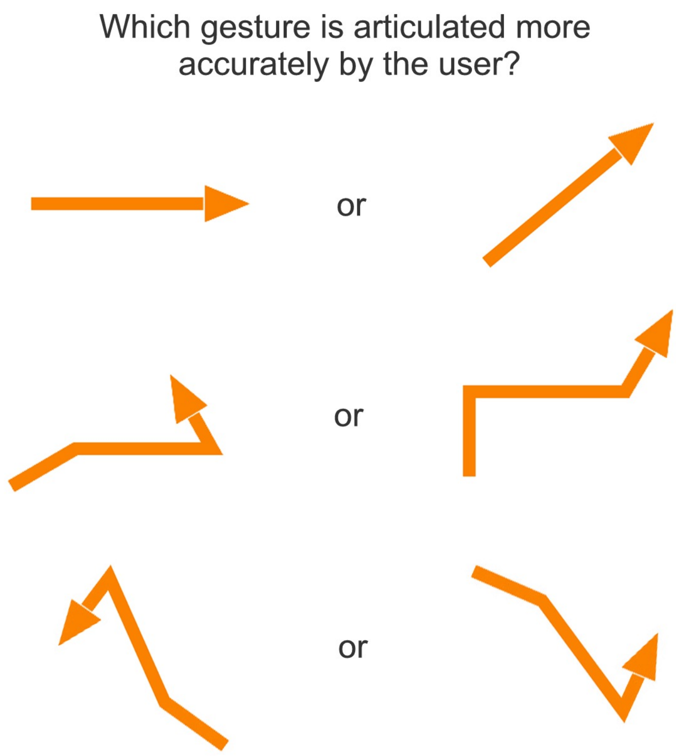


Figure 3.1.: Investigation Focus: Which of these proposed gestures are articulated more accurately?

the explaination of gesture metrics and the segmentation of the composed gestures used in the studies.

3.1. Investigation Focus

The analysis of the articulated gestures investigates the effect of external influences as well as the effect of changing gesture features with respect to the articulation accuracy of unistroke gestures on mobile devices. Figure 3.2 illustrates the investigations of the large scale study and the control study.

3.1.1. Articulation Accuracy: External Influences

The comparison of the two studies gives insight in the effect of the environment and external influences to the touch accuracy. The large scale study (game) provides a higher

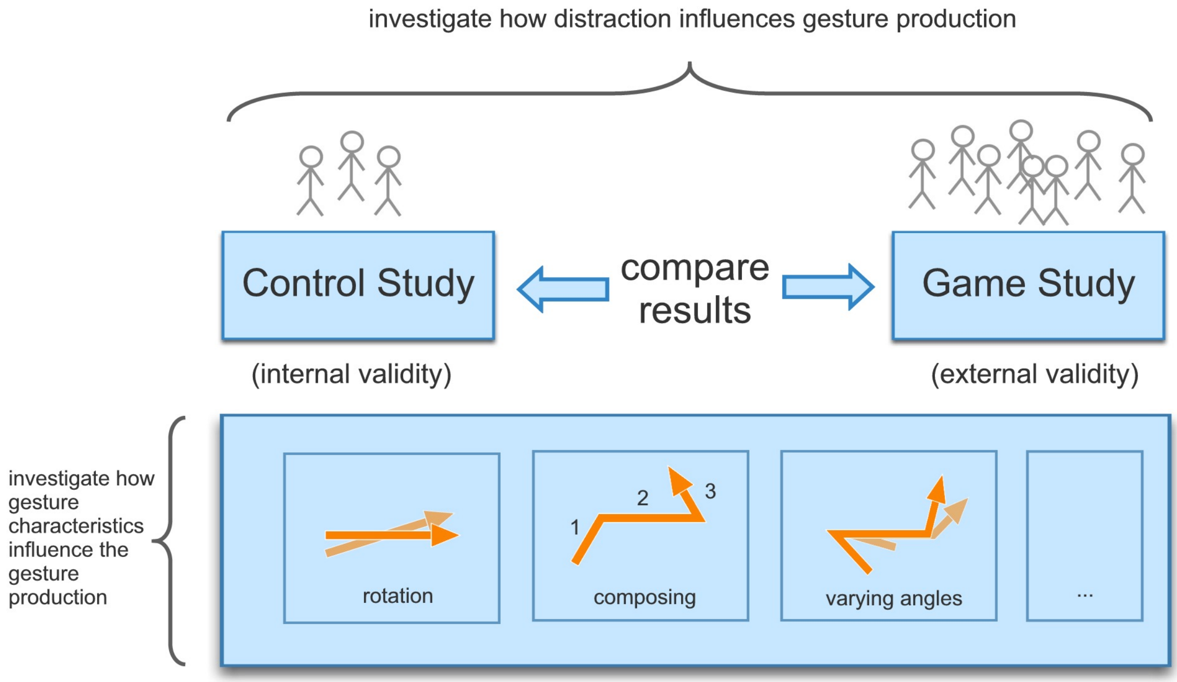


Figure 3.2.: Investigating the articulation accuracy depending on external influences and gesture characteristics.

external validity compared to the control study which has a higher internal validity. We assume that the users play the game in their natural environment. Hence, we are not able to control the used implement (thumb, index finger, or pen) for the gesture articulation as well as the distraction of the user during the game study. Furthermore, the position on the users is not controlled which means that the users can play the game in positions like sitting or walking. The effect of distraction is also caused by the visual and acoustic appearance (background animations on the screen and music) of the game. Furthermore, the user is given a feedback for inaccurate articulated gestures. We assume that the users who are playing the game are distributed across different countries, vary in their age, use different screen sizes, and are left- and right-handed. That's why the large scale game study covers a broader group of smart phone users compared to the control study.

3.1.2. Articulation Accuracy: Gesture Features

As shown in Figure 3.2, the gestures are varying in their features across both studies. The analysis focuses on unistroke gestures consisting of straight lines. The gestures are analyzed scale invariant which means that the size of the articulated gestures has no effect on the measured accuracy. In the following, the varying features are listed.

Orientation. The gestures are rotated globally in their orientation. The analysis shows how the orientation of the whole gesture as well as the global orientation of the gestures' segments affect the articulation accuracy.

Angle. The angles within the gestures are varying from 30° to 150°. This gives insight into the articulation accuracy with respect to sharp and obtuse angles.

Articulation Order. The analysis investigates the effect how the execution order of the gesture segments relate to the users articulation accuracy. Therefore, we compare the accuracy of starting, in between, and ending line segments of the gestures.

Number of Line Legments. We analyze the differences in the accuracy of single line gestures compared to the line segments of composed gestures consisting of multiple lines.

3.1.3. Articulation Accuracy: Accuracy Measurement

When analyzing the accuracy of articulated unistroke gestures, there is the need to define how the articulation accuracy is measured. In the following, the factors relevant to the accuracy measurement are described (see also Subsection 2.1.1). Hence, the term "error" is defined by the deviation between a proposed gesture and the actual articulated gesture. The inclusion of multiple accuracy definitions is also necessary because the results of this work should be relevant to the creation of robust gesture recognizers which concentrate on these different aspects of the gesture. The particular metrics used for the analysis is described in Subsection 3.3.1.

Shape Errors. The deviation of a gesture shape during articulation is expressed by the shape error. Furthermore, there are shape error metrics focusing on specific regions like the corners to express the user's deviation in these areas.

Bending Errors. The users tendency to articulate gestures bended (deviation from the proposed directions) is expressed by the bending errors. We also distinguish from bending errors in specific regions from smooth bending errors across the whole gesture.

Orientation and Angle Errors. We analyze the users accuracy of line gestures with respect to the global angle error (orientation) and the relative angle error between the line segments of the gestures.

Aspect Ratio Errors. This measurement expresses the stretching accuracy of the articulated gestures. Although a gesture has a low shape error, it can be faulty by being stretched.

Length Ratio Errors. The user may articulate specific parts of a gesture longer or shorter compared to the parts of the intended gesture. Because we analyze the gesture accuracy independently from the absolute gesture size, the articulated length accuracy is expressed as a ratio to the gesture path length or the length of a particular segment.

3.2. Investigation Hypothesis

The next hypothesis show the main focus of the gesture evaluation. Note that the term "accuracy" used in the questions is described by the bending error, the angle error, length error and the shape error compared to the proposed gesture. This metrics will be described in detail in the next Subsection 3.3.1.

Hypothesis 1: The orientation (global rotation) of single line gestures affects the articulation accuracy. The assumption is that vertical and horizontal lines are articulated more accurately than diagonal ones.

Hypothesis 2: The angle within gestures affects the articulation accuracy. Is there a tendency that the users tend to make more deviations during gesture articulation in regions of sharp angles or obtuse angles? We assume that obtuse angles are drawn more curved than sharp angles but sharp angles tend to be more shorten. We are also interested how accurate the angles articulated in terms of the difference between the axis's angles and the articulated angles.

Hypothesis 3: The orientation of composed line gestures affects the articulated accuracy. We define the orientation of the composed line gestures by the counter clockwise rotation of the mid line segment.

Hypothesis 4: Gestures consisting of only one line are articulated more accurately than composed line gestures. We expect single lines to be more accurate than the line segments of composed gestures because of their lower complexity.

Hypothesis 5: The position (articulation order) of a line segment within a gesture is relevant to the articulation accuracy. We assume that gesture segments at the beginning of the gesture are articulated more precise than segments at the end of the gesture.

Hypothesis 6: The accuracy of the articulated gestures is affected by external influences such as distraction. We assume that distracted users tend to articulate gestures more error prone compared to not distracted users.

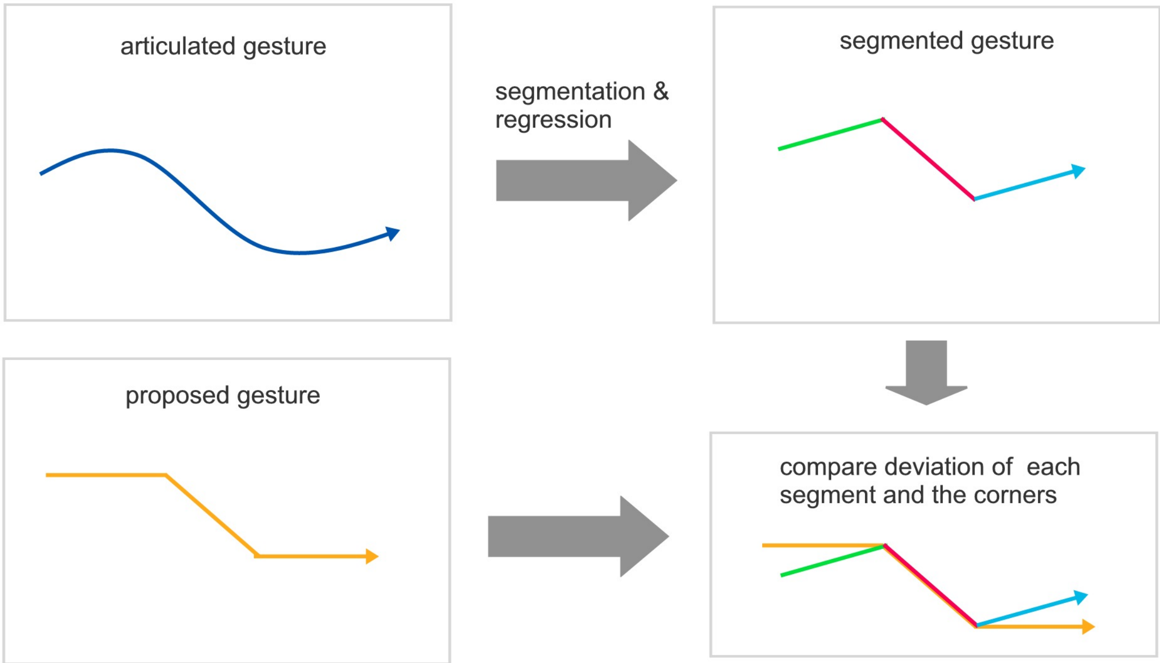


Figure 3.3.: Compare how the articulated gesture deviates from the proposed gesture (angle and orientation errors).

3.3. Analysis Methodology

The logs of the control study and the large scale game study are evaluated in the same way to make the findings comparable. Therefore, the script language python is used which enables the automated gesture evaluation for both studies. To test statistic differences between the articulated gestures (and gesture segments), Welch's t-test was applied with respect to the metrics of Subsection 3.3.1.

3.3.1. Metrics

To determine the users' accuracy when articulating a gesture, each gesture is represented by a set of spatial and temporal metrics. These metrics are divided into two groups. First, absolute metrics (features) which describe the spatial and temporal features of the gesture. Second, relative metrics describe the deviation between the articulated gesture and the intended gesture.

Absolute Metrics. Table 3.1 shows the absoute gesture metrics (features) and on which gesture type or gesture segment they are applied. First, the characteristics of the gesture bounding box are computed. This includes the angle of the diagonal to the x-axis

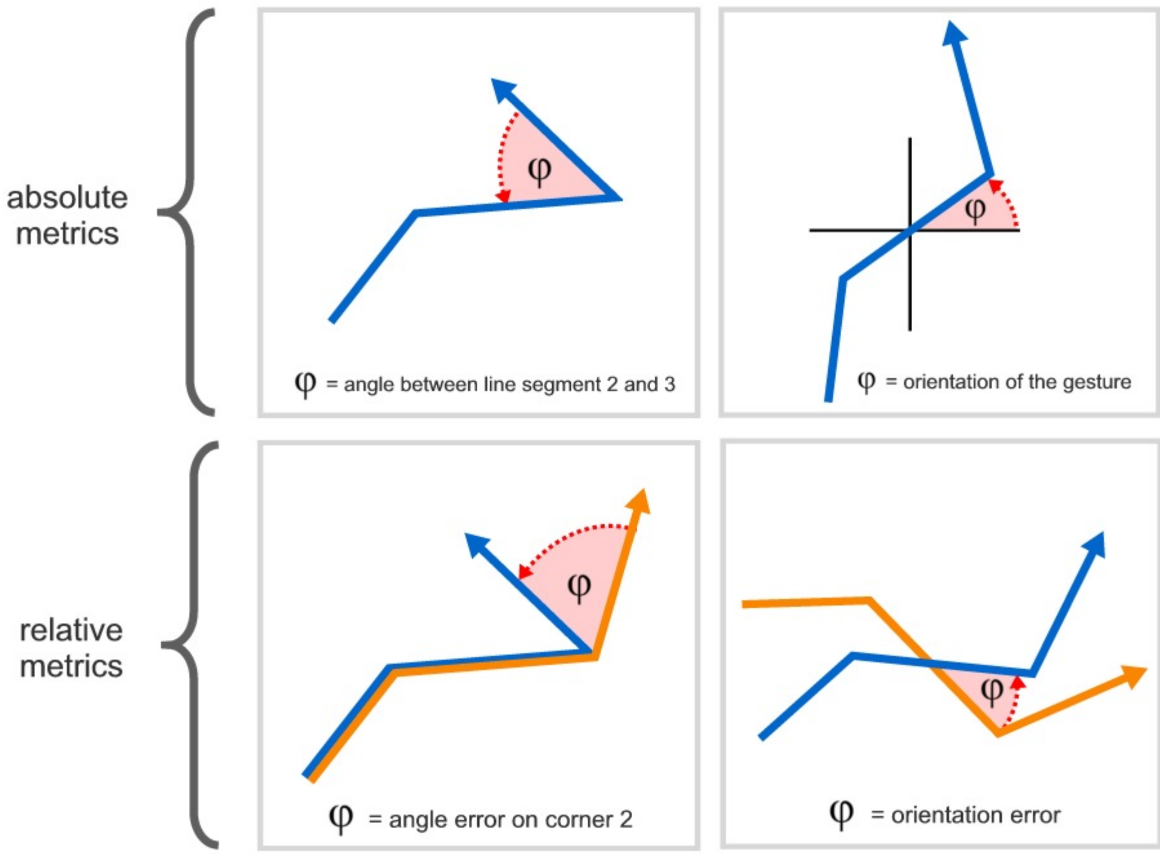


Figure 3.4.: Angle and orientation metrics illustrated on a composed line gesture. The proposed gesture is colored orange and the articulated gesture after segmentation and regression is colored blue. Note that the line segments are enumerated by their articulation order.

(*bounding box diagonal angle*), the center position of the bounding box normalized by the screen size of the device (*bounding box center position normalized*), and the aspect ratio (*bounding box ratio*). Second, the gesture length features are computed by the length of the stroke path (*path length*) and the *aperture*. The *aperture* is defined by the Euclidean distance between the starting and the ending point of the gesture and describes the closeness of the gesture (see rubine feature 5 in Rubine [Rub91]). For all length metrics it has to be considered that during the game study devices with varying screen sizes and resolutions are used. To make the the metrics comparable, the Euclidean distance units have to be converted from pixels to the actual length in mm by using the screen's dpi information. Next, the gesture features in terms of its curvature is described by the *total turning angle*, the *curviness*, and the *sharpness*. The total turning angle is the sum of the angles between each line segment of the gesture (cf. rubine feature 9 in Rubine [Rub91]). Also, the *curviness* is computed by adding up the angles between the gesture's resampled line segments, except for the sharp angles which are above the treshold of 18° (cf. Long et al. [Lon+99]). The *sharpness* metric (rubine feature 11) is used to distinguish between smooth gestures and gestures with sharp angles (cf. Rubine [Rub91]). The orientation of the gesture is defined by the indicative angle which is the angle between the centroid (geometric center) and the starting point if the gesture (cf. Wobbrock et al. [WWL07]). Next the gesture's dynamic features are described by the duration of the gesture articulation (*duration*)and the average speed (*speed*). Additionally, the single line gestures are characterized by their length (*line length*) the lines' global orientation (*line orientation*), and the angle between the line segments within the gesture *line segment angle*. The *line length* is the Euclidean distance between the start and the end point of the gesture and describes the length of the line without considering its curvature. The *line orientation* is computed by doing a linear orthogonal regression through the gesture points and measuring the angle between the x-axis and the resulting regression line. Additionally, the direction of the drawn gesture has to be considered to determine angles over 180 degrees. For the gestures consisting of composed lines, the length (*line segment length*), the orientation (*line segment orientation*), and the curviness (*line segment curviness*) are computed for each line segment (see Figure 3.3). Therefore, the gesture has to be divided into its three line segments and afterwards each metric is computed like described before. Because an accurate segmentation auf the gesture has an significant impact on the further evaluation of the line segments, an adequate segmentation algorithm has to be chosen. An error-prone line segmentation has the effect that the actual articulation accuracy of the participant can not be analyzed because of the inaccurate representation of the line segments. The used segmentation algorithm is described in Subsection 3.3.2.

Relative Metrics. The relative metrics express the deviation of the articulated gesture by the participant and the proposed gesture. As the evaluation of the gesture accuracy is independen from the absolute gesture size, the length errors have to be normalized with respect to the gesture size. Table 3.2 shows all relative metrics used for the gesture evaluation. The metrics *bounding box diagonal angle error*, *bounding box ratio error*, *indicative angle error*, *total turning angle error*, *curviness error*, *sharpness error*, *line orientation error*, *line segment orientation error* and *line segment angle error neighbors* are the absolute value of the difference between the metrics of the articulated gesture and the proposed gesture. The *relative aperture error* the aperture of the articulated gesture and the proposed gesture are normalized by their *path length* and after that the absolute value of their difference is computed. The error made in the gestures shape is expressed by the *shape error*, the *local shape errors*, and the *shape variability* (cf. [VAW13]). The *local shape errors* describes the spatial deviation from the articulated gesture and the proposed gesture for each resampled point of the articulated gesture. This metric is computed by scaling the articulated gesture and the proposed gesture to a square of fixed size, resampling it to a fix count of samples, rotating it by the reverse *indicative angle* and

Name	Gesture Type
bounding box diagonal angle	all gestures
bounding box center position normalized	all gestures
bounding box ratio	all gestures
path length	all gestures
aperture	all gestures
total turning angle	all gestures
curviness	all gestures
sharpness	all gestures
indicative angle	all gestures
duration	all gestures
speed	all gestures
line length	single line
line orientation	single line
line segment length (segment 1-3)	composed lines
line segment orientation (segment 1-3)	composed lines
line segment angle ($\sphericalangle$ corner 1 and $\sphericalangle$ corner 2)	
line segment curviness (segment 1-3)	composed lines

Table 3.1.: Absolute gesture metrics and the corresponding gesture segments.

translating the gesture centroid to the origin. After that, the Euclidean distance for each point is computed. For this evaluation the algorithm was adapted for 1D gestures (lines) by scaling them with maintaining the aspect ratio. The *shape error* is the average value of all *local shape errors* and expresses the spatial deviation of the articulated gesture and the proposed gesture. The *shape variability* is the standard deviation of the *local shape errors*. As Vatavu et al. [VAW13] describe, a low value for the *shape variability* indicates a consistent shape error across the whole gesture while a high value shows that the shape errors were made in a specific segment of the gesture. Furthermore, the metrics *bending error*, *local bending errors*, and *bending variability* describe the direction errors made during gesture articulation (cf. Vatavu et al. [VAW13]). Therefore, the articulated gesture and the proposed gesture have to be resampled and after that the angle between each line segment within the gesture is computed. The differences between each angle of the articulated gesture and the proposed gesture is expressed by the *local bending errors* metric. The *bending error* is the average of all *local bending errors* and the *bending variability* is the standard variance of the *local bending errors*. For the gestures consisting of composed lines, the *line segment length ratio error* expresses the relative length error of each line segment in relation to the gestures *path length* and the *line segment length*

Name	**Gesture Type**
bounding box diagonal angle error	all gestures
bounding box ratio error	all gestures
shape error	all gestures
local shape errors	all gestures
shape variability	all gestures
relative aperture error	all gestures
bending error	all gestures
local bending errors	all gestures
bending variability	all gestures
indicative angle error	all gestures
total turning angle error	all gestures
curviness error	all gestures
sharpness error	all gestures
line orientation error	single line
line segment orientation error (segment 1-3)	composed lines
line segment length ratio error (segment 1-3)	composed lines
line segment length ratio error neighbors (l1/l2 and l3/l2)	composed lines
line segment angle error ($\sphericalangle$ corner 1 and $\sphericalangle$ corner 2)	composed lines
line segment shape error(segment 1-3)	composed lines
line segment bending error (segment 1-3)	composed lines
CSD (corner 1 and 2)	composed lines

Table 3.2.: Relative gesture metrics and the corresponding gesture segments.

ratio error neighbors describes the length error between the neighboring line segments. Like for the single lines, the length for each line segment is computed by the Euclidean distance between the first and the last point of the line segment. Furthermore, the *line segment shape error* and the *line segment bending error* are the *shape error* and the *bending error* for each line segment. Last, the CSD as described by Tu et al. [TRZ15] expresses the *shape error* in the corner regions of the gesture. Therefore, the corner is extracted by identifying the corner point (see Subsection 3.3.2), resampling the left and the right leg of the corner and then computing the *shape error*.

3.3.2. Gesture Segmentation

A prerequisite for a valid line segment evaluation is the accurate line segmentation of the composed line gestures. The main issue of the line segmentation is finding the two corners of the gesture. Therefore an adequate segmentation algorithm has to be chosen with respect to the following requirements. First, the algorithm has to be robust for length errors within the gesture (Requirement *Length Ratio Robustness*). For example, the algorithm should be able to detect all corners even if the length ratio (*line length/path length*) of a line segment differs strongly from the length ratio of the proposed gestures' corresponding line segment. Second, the algorithm has to be robust for local variations within a line segment (additional corner or additional line) so that the error does not affect the recognition of the other corners (Requirement *Global Robustness*). Next, a deviation in the angle between the line segments must not affect the correct recognition of the corners (Requirement *Angle Variance Robustness*). Last, the algorithm has to detect corners even if there are none, which is the case for obtuse angles (Requirement *Numb Angle Robustness*). The gesture logs showed that the obtuse angles were mostly drawn as arcs. In this case the algorithm has to cut the arc at the position which matches the corresponding proposed gesture the most. The following algorithms were tested with a subset of the control gestures and rated (see Table 3.3).

Min Speed K-Means Algorithm. As presented by Tu et al. [TRZ15], this corner detection algorithm is used for the calculation of the CSD. This algorithm is based on the fact that the user has to slow down his finger movement when a corner is drawn. So the list of the gestures points is sorted by the speed of the points in ascending order. After that, the algorithm clusters the points by their position by applying a K-Means algorithm for the fastest points of the list. Each cluster is the defined as a corner of the gesture (starting and ending points are ignored). Testing this algorithm by the composed line gestures showed good results for sharp angles (because of the strong speed difference) but performed bad for obtuse angles. Moreover, the algorithm detected additional corners within badly performed line segments. On the other hand, the algorithm was robust for differences in different line lengths.

Template Matching Algorithm. This algorithm normalizes the articulated gesture and the proposed gesture as described for the *shape error*. Next the articulated gesture is cut at the bisector of the proposed gesture. The intersection of the bisector and the articulated gesture is defined as the corner. While this algorithm is robust for local variations of the line segments, it performed generally badly because of the length differences in relation to the proposed gesture.

Point Set Registration Algorithm. This algorithm aligns two points sets by performing rotation, translation and scaling in multiple iterations (cf. Myronenko and Song [MS10]). After matching the articulated gesture with the proposed gesture, the algorithm is

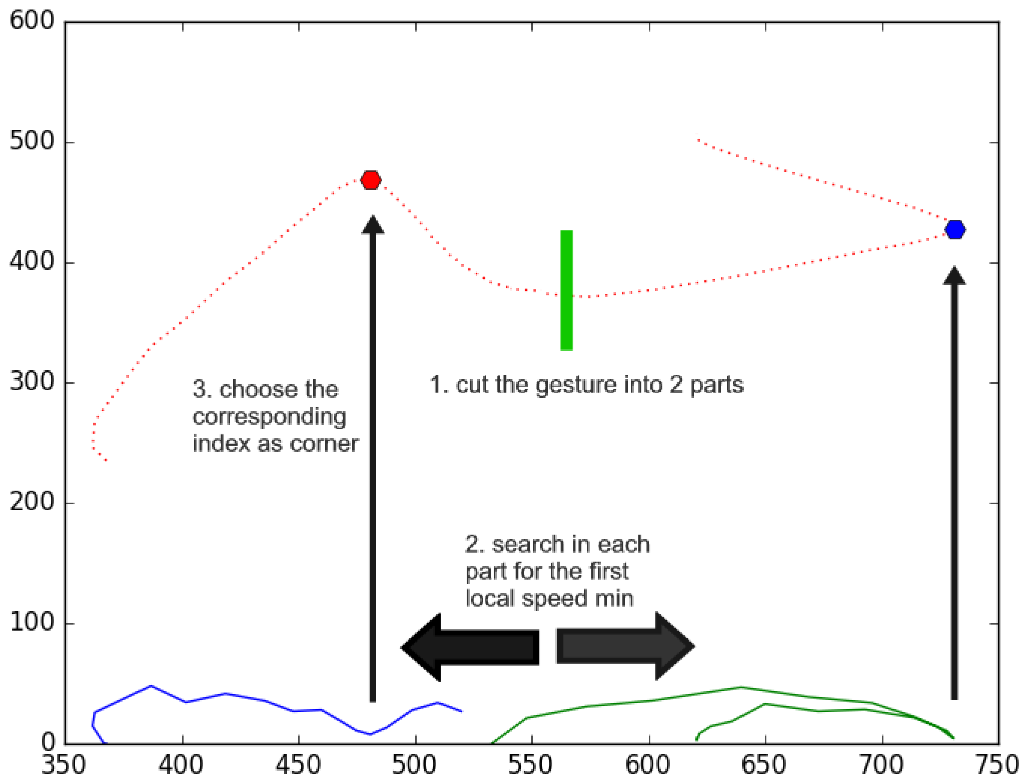
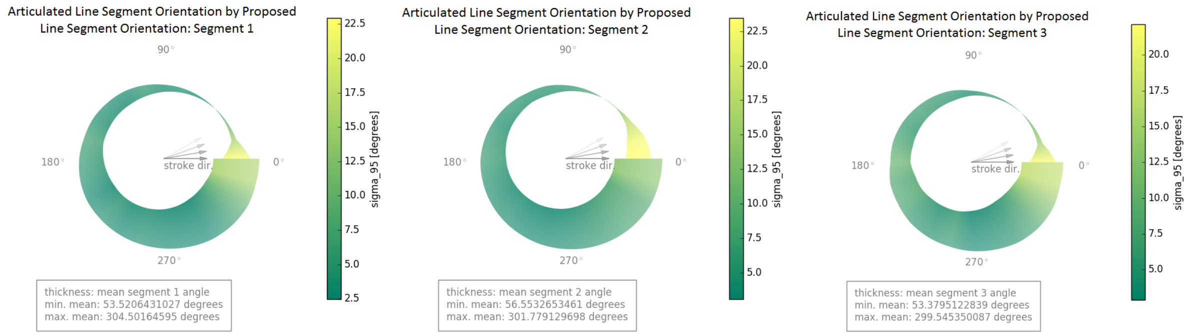


Figure 3.5.: The segmentation algorithm segments the gesture on the red and blue dot. The corresponding speed of the gesture points is displayed by the blue (first gesture segment) and the green (second gesture segment) line. The articulated gesture is displayed dotted.

extended so the gesture is cut as in the *Template Matching Algorithm*. However, this algorithm had no acceptable improvement to the *Template Matching Algorithm*.

Template Direction Matching Algorithm. This method rotates the gesture by the angle between the first two line segments of the proposed gesture so that the first line segment is parallel to the x-axis. Then the distance from each point to the x-axis is computed until the center of the gesture is reached. The corner is defined by the point where slope the distance function reaches a defined treshold. Then the algorithm is repeated for the last line segment to get the second corner. This algorithm is robust for differences in the length of the line segments and also for small variations within a line segment. On the other hand, the test showed that the algorithm performed badly for varying angles within the gesture because the slope of the distance was too slow to detect the corners. Besides, obtuse angles were not recognized well.



(a) Line segment 1 orientation **(b)** Line segment 2 orientation **(c)** Line segment 3 orientation

Figure 3.6.: Test results for the gesture segmentation algorithm with respect to the orientation of the line segments. After the segmentation, the orientation of each line segment is computed. The figures (a) to (c) display how the orientation of the proposed line segments and the articulated line segments correlate.

Two Sided Local Min Speed Algorithm. Like the *Min Speed K-Means Algorithm* this algorithm is based on the fact that the speed within a line segment is higher than the speed on edges. First, the algorithm resamples the gesture and then splits it at the point which is the nearest to the half-path length. After that the speed for each point of the two gesture segments is computed. Next, the algorithm steps from the center point in the direction of the starting point on the path (and in the direction of the ending point for the second gesture element) and searches for a local speed minimum. The first minimum is then selected as the corner of the gesture segment. This algorithm assumes that the path center point of the gesture is the path of the middle line segment. So the speed decreases when getting near to the corner and increases when the next line segment is drawn. Like the *Min Speed K-Means Algorithm*, this algorithm works well for sharp edges and by splitting the gesture into two segments a badly articulated line segment does not affect the corner detection of the other corner. Also, variations at the start and at the end of the gesture have no effect on the corner detection. Furthermore, the algorithm performs better in detecting obtuse angles compared to the other algorithms because it considers the local speed minimum instead of the global minimum compared to the *Min Speed K-Means Algorithm*. However, the algorithm tends to cut the gesture into its line segments closer to the center point than the *Min Speed K-Means Algorithm*, which has to be considered when evaluating the gesture segments (the middle line segment may be more accurate than the starting and ending segment). Furthermore, this algorithm is tailored only for this specific type of gesture and cannot applied for other gesture types. Figure 3.5 illustrates the segmentation algorithm applied on an example gesture.

Algorithm	Length Ratio R.	Global R.	Angle Var. R.	Obtuse Angle R.
Min Speed K-Means	good	moderate	good	poor
Template Matching	poor	moderate	poor	moderate
Point Set Registration	poor	moderate	poor	moderate
Template Dir. Matching	good	moderate	poor	poor
Two Sided L. Min Speed	moderate	good	good	moderate

Table 3.3.: Line Segmentation Algorithms and their rating with respect to the segmentation requirements (qualitative rating based on the test data set).

As shown in Table 3.3 shown, the *Two Sided Local Min Speed Algorithm* has the best results in relation to the segmentation requirements. Therefore this algorithm is chosen for the line segmentation. Figure 3.6 illustrates the test results with respect to the orientation of the line segments.

3.4. Summary

This Chapter presented the research goal of this work and the analysis methodology. To understand the user's accuracy when articulating unistroke touch gestures on mobile devices, a large scale study (game) and a control study is executed. The comparison of the two studies gives insight into how external influences affect the users' articulation accuracy. Further we analyze the effect of varying gesture features to the articulation accuracy. The accuracy is described by different metrics which express the articulation errors with respect to the shape, bending, stretching, orientation, angles, and length of the gesture. The analysis of composed line gestures requires a robust segmentation to provide a valid analysis of its line segments.

4. Control Study

In order to analyze how external influences affect the users gesture accuracy on mobile devices, there is the need for a control study which ensures a higher internal validity than the large scale study. In the study, we eliminated the effect of different visual appearing, distraction trough the back screen and the effect of different context and context changes of the player. First, Section 4.1 describes the study design including the apparatus and the study procedure. After that, the results of the study analysis are described in Section 4.2. Last, Section 4.3 discusses the results with respect to the investigation focus.

4.1. Method

This section presents the study apparatus and how the study was executed. Additionally, the participants which conducted the study are described.

4.1.1. Apparatus

In order to reduce the external influences and collect additional data during the study, a visual simple version of the game study's apparatus was developed. The background of the screen consisted of a white box where the proposed gestures were shown and a gray box at the bottom where the gestures are articulated. In the white area, the gestures are falling from the top to the bottom of the screen. Because the falling animation was basis of the game principle of the large scale study's game apparatus, the falling animation was also shown by the control study's apparatus. Therefore, both studies are comparable. Furthermore, the participants get no feedback how accurate the gestures were performed. Figure 4.1 displays the study's resulting screen.

Additionally, the control study apparatus includes a questionnaire which collect's the participant data and a non-weighted NASA Task Load Index (NASA-TLX) test (test described by Hart [Har06]). The questionary asks for the users name, age, physical limitations, the dominant hand, and if he uses a smart phone daily. Furthermore, the

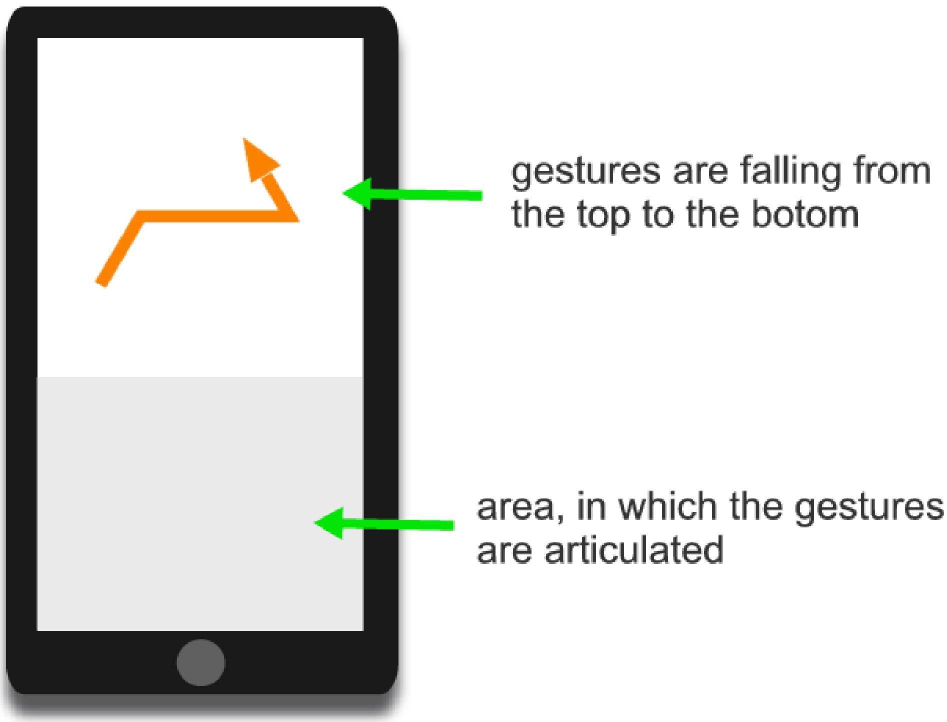


Figure 4.1.: The control study's apparatus. The gestures are falling from the top to the bottom of the Screen.

study tutorial was changed in a way that the participant is shown the score of the performed gesture immediately after articulating (on a scale from 0 to 10).

The used hardware was a Nexus 5X (LG V10) smart phone. It is running on Android 5.1.1 and has a 5.7" screen (2560×1440 Pixel).

4.1.2. Design

The control study aims to provide internal validity by excluding external influences. Furthermore, we're interested in the participant's performance when doing single lines and composite lines in varying orientations and angles inside the gestures.

Reduced Distraction. By reducing all unnecessary visual stimuli which could distract the participant, only shown a white screen is shown with the falling gestures and a gray box where the gestures have to be drawn. The gray area is shown for reasons that the participant is not mislead to articulate the gestures directly over the shown gesture. Otherwise, the gesture image would be visually covered and the articulated gesture would be biased because its moving down the screen. Furthermore, the participant is not given any feedback about how good he performed a gesture (in terms of the shape distance). So he articulates the gestures in a way he feels accurate enough and does not adapt to the gesture recognizers rating.

Limitations. In contrast to the large scale study, the control study has limitations in the participants and the used implement. First, the participants have to perform the gestures only by using their right thumb. Furthermore, the participants have to be in sitting position on a table and hold the mobile phone in their right hand.

Fatigue Effect. The order of the gestures which have to be performed by the participant are permuted randomly for each participant. Therefore, the order of the shown gestures has no impact on the articulated accuracy with respect to the concentration and fatigue of the participant. Also, to reduce the fatigue effect the study is splitted in to 5 levels, which gives the participant the chance for a short break. To observe the participants fatigue, the participant has to do a non-weighted NASA-TLX test. Here, the participant rates his mental demand, physical demand, temporal demand, overall performance, frustration level, and effort by using sliders. This ratings are combined to an index without weighting them.

Gesture Sets. The control study focuses on two gesture sets: single lines and composed lines. Like in Section 5.1.3 described, these gestures are varying in their orientation and the angles between the composite lines. Like in the game, also the gesture sets from previous studies are used, but without varying rotation. However, in contrast to the game, each gesture has to be performed only once by the participant. So the learning effect is not covered in this study. The single lines are rotated by 360° in 1° steps. The composite lines are rotated by 360° in 15° steps and the angles between the lines are varying in 15° steps in a range of 180°. This leads to a total amount of $(360 + 5 \times 5 \times 24 + 76)$ 1036 gestures for each participant. Consequently, each level consists of 207 gestures.

4.1.3. Procedure and Participants

Procedure

Before starting the study, it has to be ensured that the experiment won't be interrupted by push notifications or a call on the mobile phone. Therefore, the devices mode is set to "plane mode" which disables the mobile internet connection and disables incoming phone calls. The study starts with introducing the participant into the study's objective. Especially, the participant is requested to articulate the gestures as accurate as he can do. Then, has to fill in a questionnaire on the mobile phone . When the participant is in sitting position, he is allowed to start the tutorial level. Like in the game, the proposed gestures are falling from the top of the screen. The tutorial level only consists of a few gestures so that the participant gets the idea of the game. Additionally, the participant is guided by a text on the screen which explains how and where to perform the gestures. The text also displays the rating of his accuracy (which will be disabled in the following levels for not distracting) to get an immediate feedback of his performance. During the tutorial the participant will be corrected by the study's conductor if he does not act in the intended manner (for example uses the index finger instead of the thumb). When the participant completed the tutorial, he starts playing the levels 1 to 5. After each level the participant has to fill in the NASA-TLX test. Then a screen is shown which shows the players score of the level.

Participants

Nine participants took place on the control study. Their age was between 25 and 64 and none of them had any physical limitations. Six of them were right handed and three were left handed. Seven of them stated themselves as daily smart phone user and two of the participants used a smartphone every second day. On average the experiment took 25 minutes for each participant. During the experiment ($9\ participants \times 1036\ gestures$) 9324 gestures were recorded. After filtering out wrong gestures with an recognizer score less than 0.5 and gestures which were drawn on a low frame rate (under 60 FPS), there were 9271 recorded gestures left to be analyzed.

4.2. Results

This Section presents the results from the control study with the focus on single line gestures, composed line gestures, and stretched gestures.

4.2.1. Single Line Gestures Analysis

This Subsection reports the results of analyzing the single line gestures. The analysis focuses on how the gesture metrics change by varying the line orientation. However, no significant differences in the gesture metrics were found depending on the lines orientation. On the other hand, there are orientation regions which seem to be more error-prone than others with respect to the gesture metrics. These variations are approximated by a last square sine regression (see Table 4.1).

Metric	Mean	σ_{95}	Sine Regression
path length	22.3 mm	0.4 mm	M=22.0 mm A=2.3 mm T=186,7° PH=44.5° $R^2 = 0.30$ (moderate)
total turning angle	52.9°	1.9°	M=8.2° A=43.0° T=351.2° PH=21.7° $R^2 = 0.64$ (strong)
sharpness	59.5 $degrees^2$	5.4 $degrees^2$	N/A
duration	240 ms	5 ms	N/A
line length	21.7 mm	0.3 mm	M=21.8 mm A=2.3 mm T=195.6° PH=59.2° $R^2 = 0.34$ (moderate)
b. box dia. angle error	8.4°	0.3°	N/A
b. box ratio error	0.0	0.0	N/A
shape error	0.0017 mm	6.3e-5 mm	N/A
shape variability	0.0011 mm^2	3.6 mm^2	N/A
relative aperture error	0.32	0.002	N/A
bending error	0.97°	0.03°	N/A
bending variability	3.4 $degrees^2$	0.1 $degrees^2$	N/A
indicative angle error	9.7°	0.3°	N/A
line orientation error	10.4°	0.6°	M=9.6° A=1.6° T=178.5° PH=108.6° $R^2 = 0.09$ (weak)

Table 4.1.: Aggregated metrics of single line gestures. Each metric was computed with a sample size of 2819 lines. N/A is used for $R^2 < 0.1$ or the metric was not analyzed in detail.

Gesture Features. The analysis shows an average line length of 21.7 mm (σ_{95}=0.3 mm). The line gestures were performed with an average turning angle of 52.9° (σ_{95}=1.9°). The positive value indicates that the gestures were likely to be bended counter clockwise or the counter clockwise bended lines were stronger bended than the clockwise bended ones. Furthermore, the single line gestures were articulated with an average orientation error of 10.4° (σ_{95}=0.6°).

Orientation Dependency. As shown in Figure 4.3a and Figure 4.3b, the diagonal lines in the first and third quadrant tended to be longer than the lines in than in quadrant two and four. The sine regression shows a weak correlation of the experimental data with $R^2 = 0.34$. Because the participants used their right thumb when performing the experiment, we assume that the lines which are articulated in the diagonal left and right direction of the thumb were articulated 2.3 mm longer than the lines produced by a push or pull movement of the thumb (with respect to the sine regression).
Figure 4.4a and Figure 4.4b are showing the total turning angle depending on the line orientation. The lines articulated to the left and in upward direction (quadrant 1 and 4) tended to be bended stronger in one direction while the lines in the area of 180° had a low turning angle near zero. Figure 4.2 shows the arc like bending of the gestures performed with an intended orientation of 74°. The sine correlation for the total turning ange has a strong correlation with $R^2 = 0.64$. The orientation error of the line gestures showed to have a no significant correlation of ($R^2 = 0.09$) according to the sine regression.

4.2.2. Composed Line Gestures Analysis

The analysis of the composed line gestures focuses on the gesture orientations and the angles within the gestures. Furthermore, the length variations and the shorten of corners is shown. Table 4.2 reports the results of the analysis. Figures 4.5a to 4.5c are showing the aggregation of the composed lines gestures with respect to the local shape and bending errors.

Order Dependency. The order of the line segments and corners had a significant effect according to their accuracy. Figure 4.14b shows that the first line segment had an average bending error of 1.26° (σ_{95}=0.03°) which was significantly lower than the bending error of the last line segment (p<0.001). The low bending error of the second line segment is explained by the behavior of the segmentation algorithm which tends to cut the line segments near to the center of the gesture. For this reason the start and end segment consists of an arc which connects the line segments to the mid segment.
The first corner was articulated with an average angle error of 26.2° (σ_{95}=0.8°) which

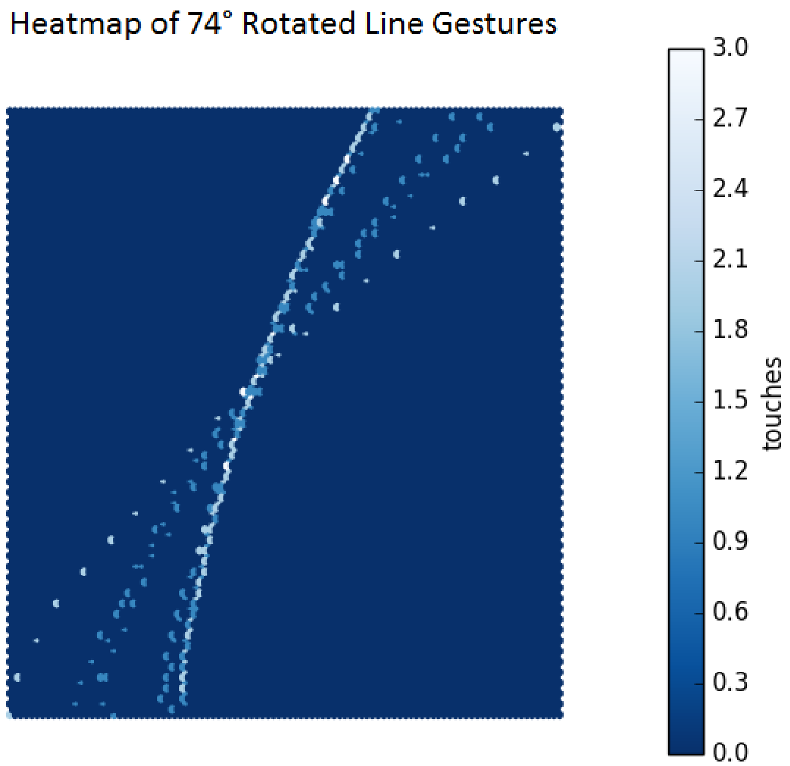
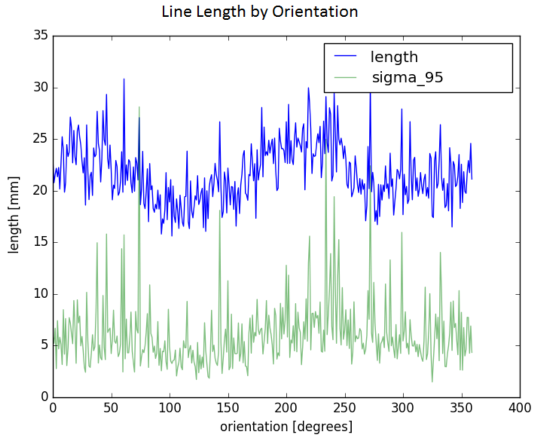
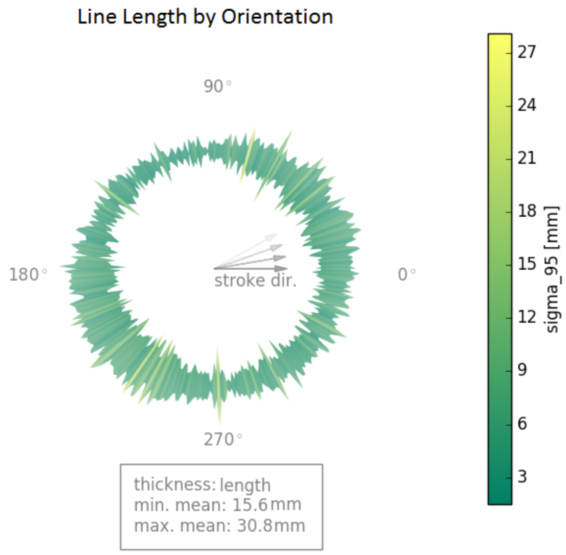


Figure 4.2.: Bending of single line gestures

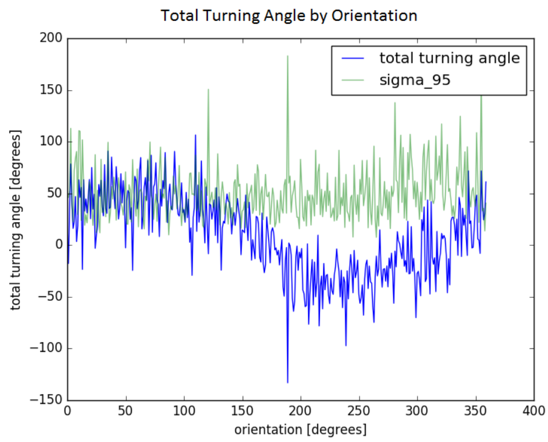
was 3.1° (σ_{95}=0.8°) lower than the average angle error of the second corner (see Figure 4.13b). Also, the global orientation error increased significantly (p<0.001) depending on the articulation order (see Figure 4.13a).
The average CSD was significantly (p<0.001) lower on the first corner compared to the CSD on the second corner (see Figure 4.13c). This shows that the second corner was articulated less accurate than the second corner. Also, the shape error (see Figure 4.14a) was significantly higher for the last line segment compared to the shape error of the first line segment (p<0.001). Like for the bending metric, we assume that the low value of the second line segment is explained by the segmentation algorithms behavior.
The length ratio error for the second line segment was significantly higher compared to the length error of the first and the last line segment (p<0.001). This result is explained by the fact that all line segment were articulated with the similar length (see Figure 4.12a) but the proposed gesture is defined by with the length ratio 0.25:0.5:0.25 (the second line segment has the double length than the first and the last line segment).

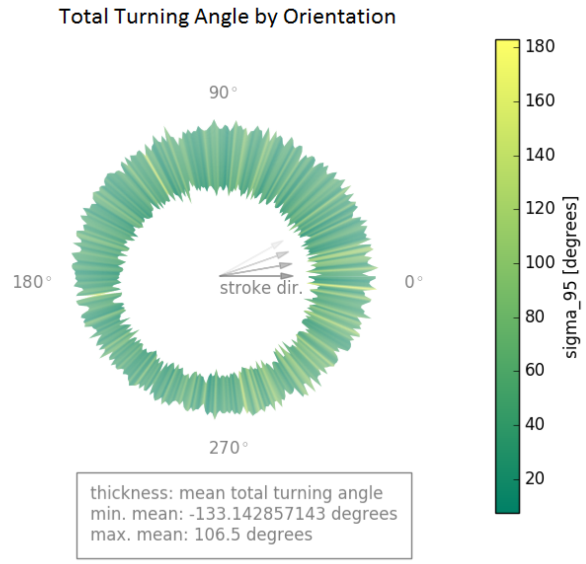




(a) Line length depending on the line orientation **(b)** Angular representation of the line length depending on the line orientation

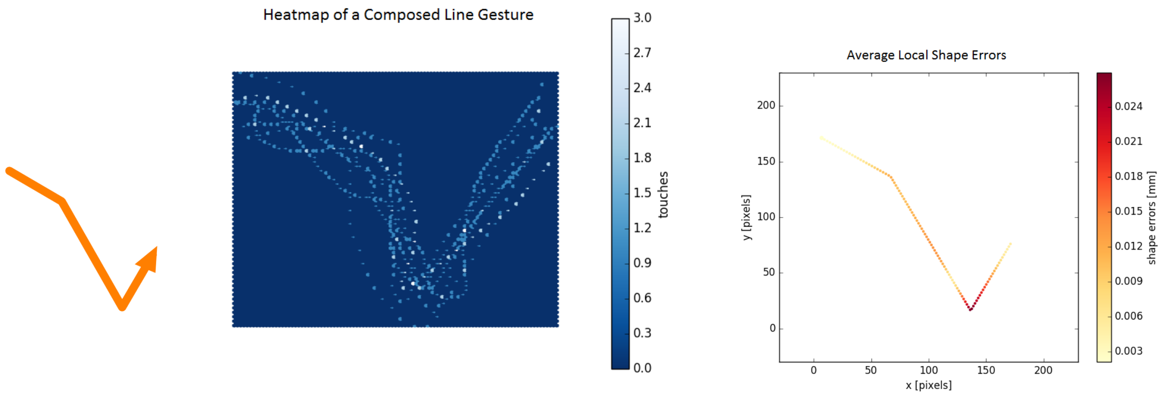
Figure 4.3.: Line length depending on the line orientation





(a) Total turning angle depending on the line orientation **(b)** [Angular representation of the total turning angle depending on the line orientation

Figure 4.4.: Total turning angle depending on the line orientation



(a) Proposed Gesture

(b) Heat map of the articulated gestures

(c) Average local shape errors mapped to the proposed gesture

Figure 4.5.: Example composed line gesture: Visualization of the local error metrics

Gesture Orientation Dependency. The orientation (angle between the x-axis and the mid-line segment) of the composed line gestures had a significant effect on the angle error and the length ratio error between the neighboring line segments.

For gestures with an orientation of 45° the average angle error of the second corner was higher compared to the gestures with an 180° orientation (see Figure 4.6). An absolute sine regression for this metric showed to correlate strong with R^2=0.63 with a global minimum near 180. Also, the first corner showed this trend an correlated with R^2=0.69. The length ratio error between line segment one and two showed to be significantly higher when the gestures were rotated by 105° (up direction) compared to gestures parallel to the x-axis with left or right direction ($p<0.001$). The same effect was observed for the length ratio error between the mid- line segment and the last line segment. While the length ratio error between the first two line segments was 0.7 (σ_{95}=0.05) for non rotated gestures, a rotation by 105° resulted in an increased error of 1.6 ($p<0.001$).

Figures 4.7a and Figure 4.8b are showing that there was also the trend that the 180° rotated gestures were articulated more accurate than gestures in other orientations. Note that the error values differ from the orientation of 0° because the direction of the articulated gesture is considered (0° orientation corresponds to a horizontal line articulated in right direction, 180° in left direction). For example the gestures with an orientation of 195° were 1.2° more bended than the gestures of 45° orientation ($p<0.001$). Also, the gesture orientation (indicative angle) was less error-prone in the regions near the orientation of 180°.

Further, the shape error was significantly lower for 180° rotated gestures than in the orientation of 45° and 225° ($p<0.001$). Last, the relative aperture error, which indicates the closeness error of a gesture, was significantly higher in the orientation of 45° and 225° than in the orientation of 0° and 180° ($p<0.001$).

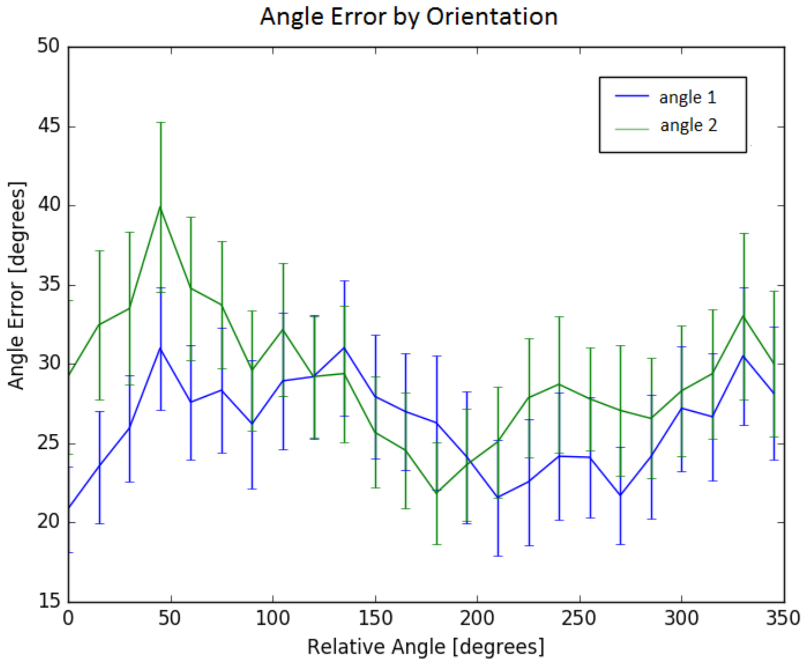
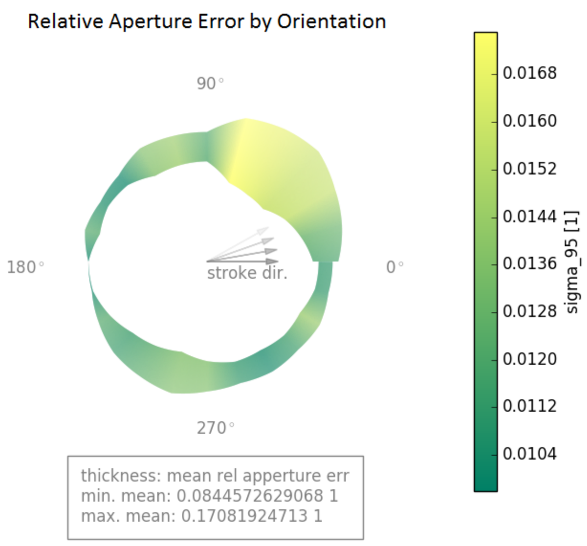
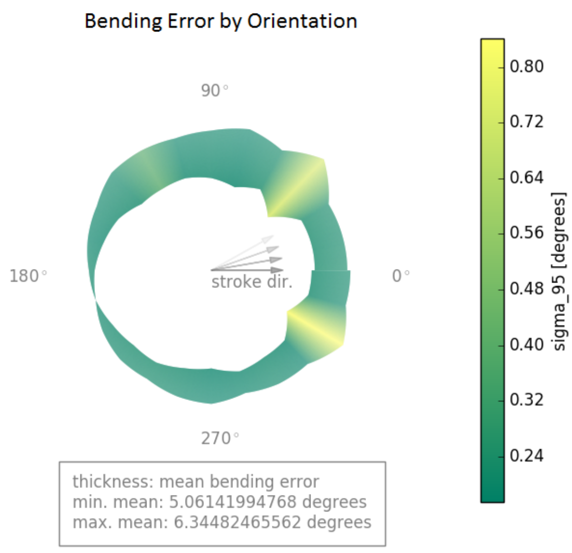


Figure 4.6.: Angle error depending on the orientation of the gesture.

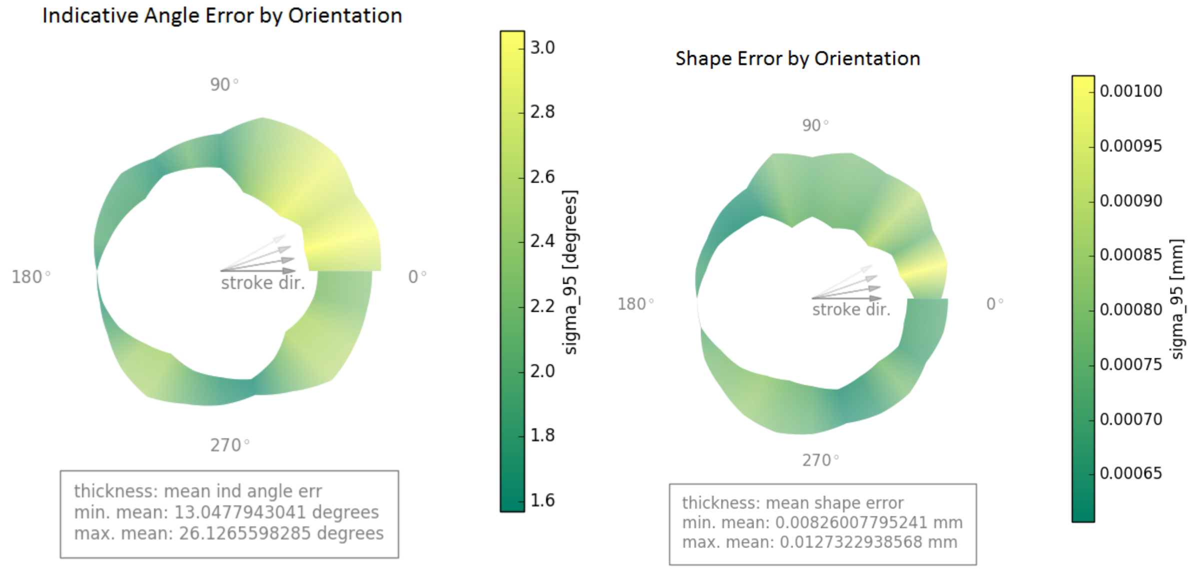


(a) Relative aperture error depending on the orientation of the gesture.



(b) Bending error depending on the orientation of the gesture.

Figure 4.7.: Relative aperture error and bending error depending on the gesture orientation

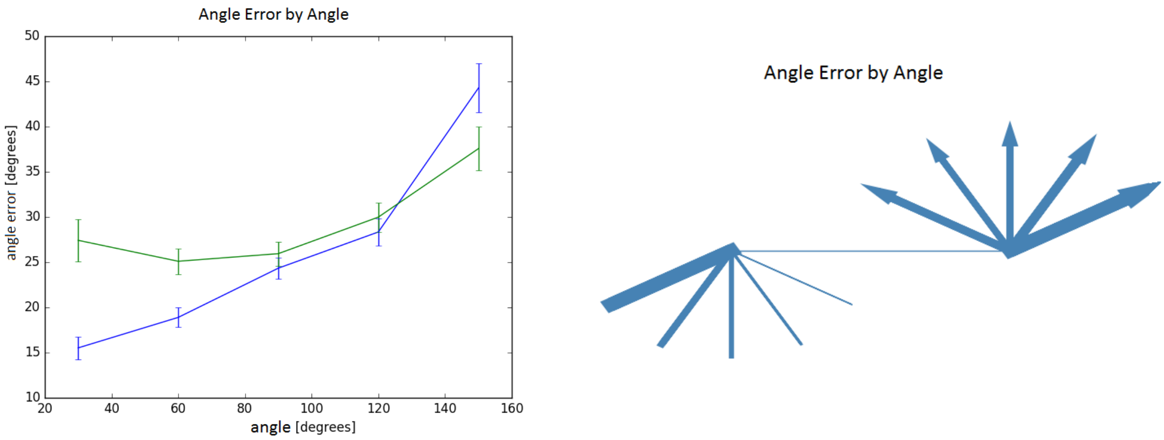


(a) Indicative angle error depending on the orientation of the gesture

(b) Shape error depending on the orientation of the gesture.

Figure 4.8.: Relative aperture error and bending error depending on the gesture orientation

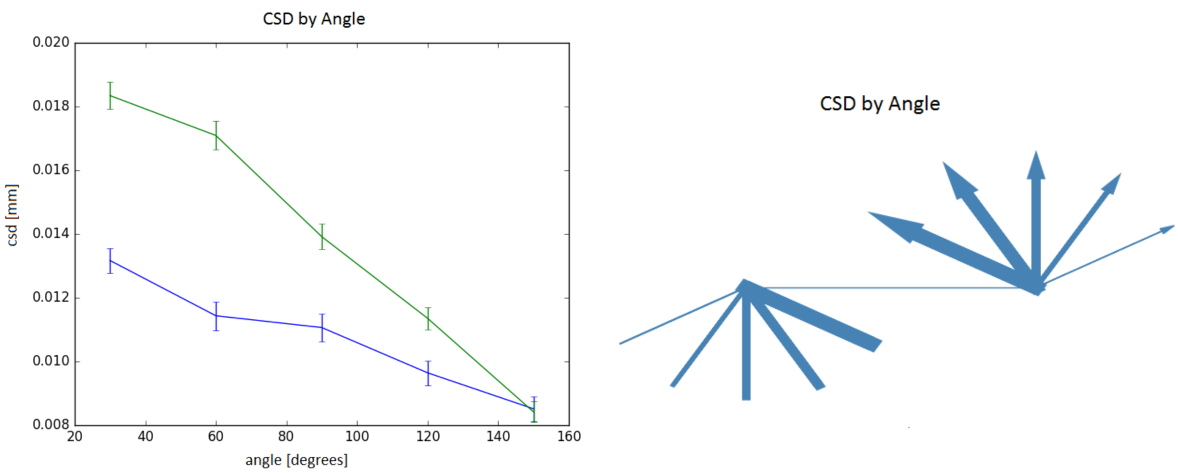
Angle Dependency. The variation of the angles within the gestures had a significant effect on the accuracy of the gesture segments. First, sharper angles tended to be articulated more accurate than obtuse angles. Figure 4.9a shows a significant increase in the angle error with an increasing intended angle between the line segments of the first corner ($p<0.001$). The average angle error by an angle of 30° increased from 15° to 42° for angles of 150°. Also, the second corner showed to be articulated less accurately with an increasing angle in the area of 60° to 150°. Figure 4.9b illustrates the articulated angle errors for the composed line gestures.
In contrast, CSD decreased with an increasing intended angle between the line segments on the first and second corner (see Figure 4.10a). This indicates that the participants were more likely to shorten sharp angles than obtuse angles. This effect can also be seen in Figure 4.5c which shoes the aggregated local shape errors of all participants mapped to the gesture proposed gesture. Last, Figure 4.11a shows that the length ratio error between the neighboring line segments of the first corner was affected significantly by the angle between them. While the line first two line segments were articulated with an average length ratio error of 0.78 for 30°, the length ratio error increased up to 1.62 ($p<0.001$).



(a) Angle error depending on the angle between the line segments. The first angle is represented by the blue line and the second angle is represented by the green line.

(b) Angle error depending on the angle illustrated on the composed line gesture. The thickness of the lines represents the values of the left chart.

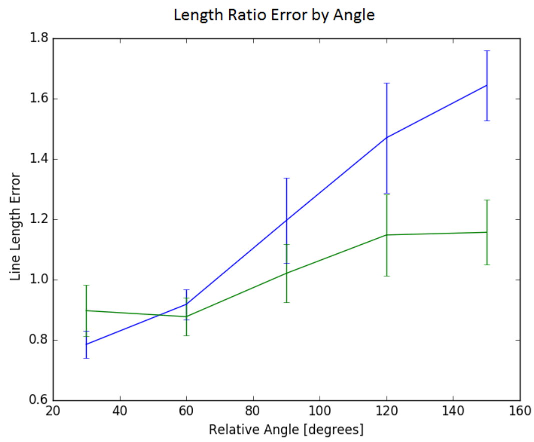
Figure 4.9.: Angle error depending on the angle within the gestures.



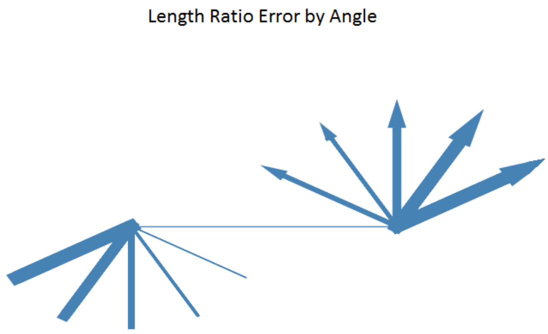
(a) CSD depending on the angle. The first angle is represented by the blue line and the second angle is represented by the green line.

(b) CSD depending on the angle illustrated on the composed line gesture. The thickness of the lines represents the values of the left chart.

Figure 4.10.: CSD depending on the angle within the gestures.



(a) Neighbor length ratio error depending on the angle. The first angle is represented by the blue line and the second angle is represented by the green line.



(b) Neighbor length ratio error depending on the angle illustrated on the composed line gesture. The thickness of the lines represents the values of the left chart.

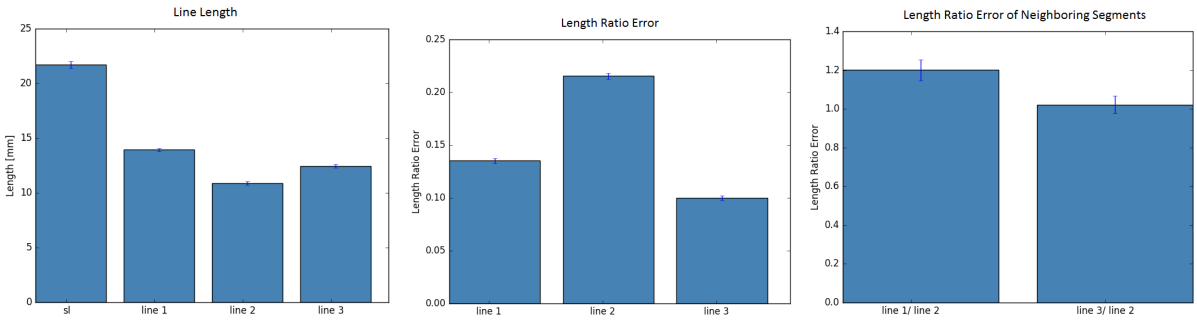
Figure 4.11.: Neighboring length ratio error depending on the angles within the gestures.

4.2.3. Composed Line Gesture vs. Single Line Gesture

The average bending of the single line gestures was significantly ($p<0.001$) lower compared to the corresponding values of the first and third line segments of the composed line gestures (See Figure 4.14b). We assume the lower bending error of the second line segment compared to the single line depends on the behavior of the segmentation algorithm. This is also the case for the low shape error of the mid-line segment (see Figure 4.14a). However, the average shape error of the single line gestures was significantly lower than the shape error of the first and last line segment of the composed gesture ($p<0.001$). Furthermore, the single line gesture was articulated significantly ($p<0.001$) longer in terms of its length compared to all line segments of the composed gestures (see Figure 4.12a). As shown in Figure 4.13a, the mean orientation error of the single line was significantly lower than the orientation error of the composed gestures' line segments ($p<0.001$). Comparing to the last line segments of the composed gesture, the single line were articulated more accurate ($p<0.001$).

Metric	Gesture Segment	Mean	σ_{95}
len. ratio error	line segment 1	0.135	0.002
len. ratio error	line segment 2	0.215	0.003
len. ratio error	line segment 3	0.100	0.002
len. ratio error neigh.	line segment 1 and 2	1.2	0.05
len. ratio error neigh.	line segment 2 and 3	1.0	0.05
angle error	$\sphericalangle$ corner 1	26.2°	0.8°
angle error	$\sphericalangle$ corner 2	29.3°	0.8°
orientation error	line segment 1	15.6°	0.6°
orientation error	line segment 2	20.0°	0.6°
orientation error	line segment 3	23.2°	0.9°
shape error	line segment 1	0.0033 mm	9.8e-5
shape error	line segment 2	0.0021 mm	6.9e-5
shape error	line segment 3	0.0064 mm	0.2e-5
bending error	line segment 1	1.26°	0.03°
bending error	line segment 2	0.62°	0.03°
bending error	line segment 3	1.56°	0.03°
CSD	corner 1	0.012 mm	1.9e-4 mm
CSD	corner 2	0.014 mm	2.0e-4 mm

Table 4.2.: Aggregated metrics of composed line gestures (3699 samples for each line segment). The second last columns contain the value *yes* if at least one value of the metric is significant different from another value of the metric under a different orientation or angle.

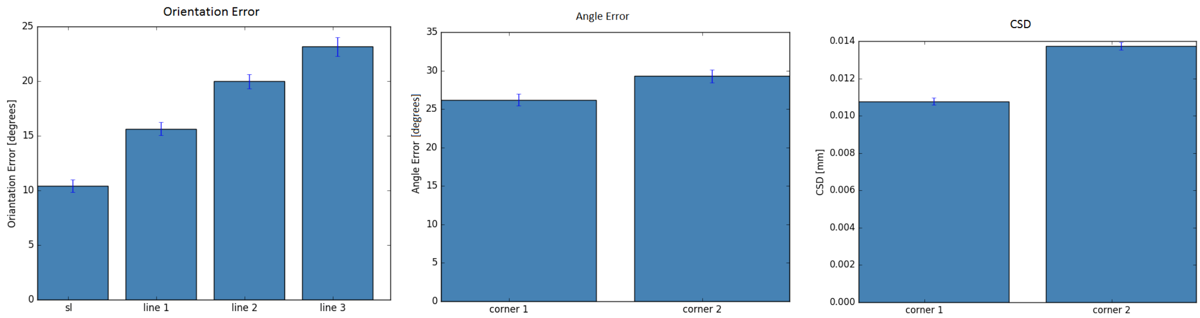


(a) Length of the line segments and the single lines.

(b) Length ratio error of the line segments and the single lines.

(c) Length ratio error of the neighboring line segments.

Figure 4.12.: Line Segment Comparison: Length and Ratio Errors.

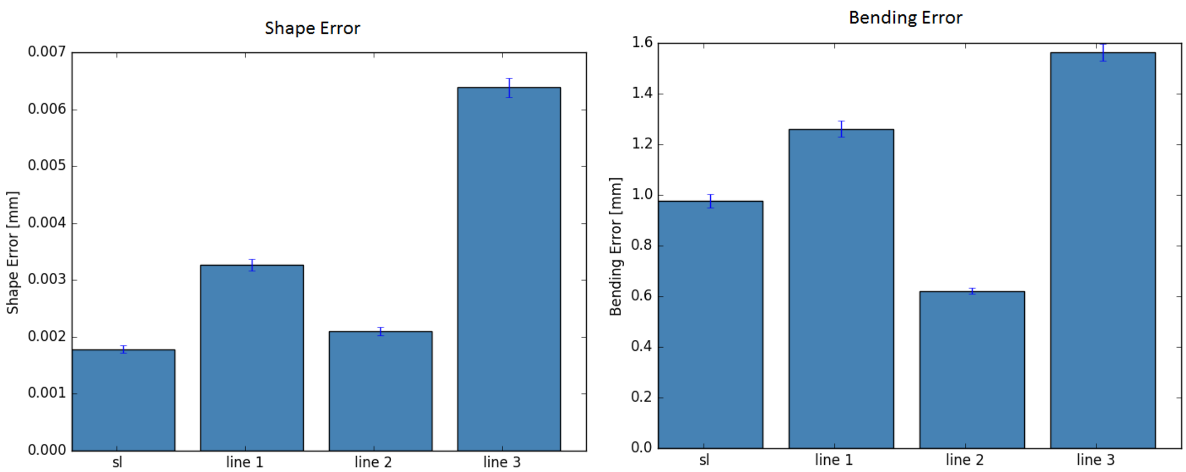


(a) Orientation error of the line segments and the sinle lines.

(b) Angle error of the angles within the gestures.

(c) CSD of the corners.

Figure 4.13.: Line Segment Comparison: Orientation error, angle error, and CSD.



(a) Shape error of the line segments and the single lines.

(b) Bending error of the line segments and the single lines.

Figure 4.14.: Line Segment Comparison: Shape error, bending error, and curviness.

4.3. Discussion

In this Section we discuss the results of the gesture analysis by focusing on the hypothesis 1-5 defined in Subsection 3.2.

Hypothesis 1: The orientation (global rotation) of single line gestures affects the articulation accuracy.

The participants tended to articulate the horizontal lines with another shape but with a similar bending compared to the articulation of vertical line gestures. The average

bending of the gestures was 0.97°
The gestures were articulated with an orientation error of 10.4° which varied by 1.6° depending on the gesture orientation (with respect to the sine regression). However, the weak correlation shows that there is no strong dependency of the to line orientation to the articulated orientation error.
Conclusion: The hypothesis 1 is statistically not verified by our study. We assume that this is caused by the strong variation in the articulation behavior between the participants.

Hypothesis 2: The angle within gestures affects the articulated accuracy.

The analysis showed that sharp corners tended to be more accurately articulated in terms of the articulated angle error between the line segments comparing to obtuse angles. On the other hand, the sharp corners were articulated less accurate than obtuse angles with respect to the shape distance in the corner area. This indicates that the participants were likely to shorten the sharp corners or articulated them more curved than the obtuse angles. Further, the obtuse articulated angles resulted in a higher length ratio error of the neighboring line segments.
Depending on whether the articulation accuracy is defined by the accuracy of the angle or by the shape in corner regions of a gesture, the articulation accuracy differs depending on the angle. This has to be considered when defining gesture sets or gesture recognizers which are either sensitive to the shape or the angle in corner regions.
Conclusion: The hypothesis 2 is verified by the study.

Hypothesis 3: The orientation of composed line gestures affects the articulated accuracy.

We discovered the trend that the horizontal oriented composed line gestures (non rotated or 180°) rotated mid-line segment) were articulated more accurately than gestures in other orientations. The gestures rotated by 180°, which is a line gesture with a horizontal mid-line segment and a start and end segment in down direction, showed to be the most accurate. The most affected metrics were the bending error, the shape error, the gestures overall orientation (indicative angle), and the relative aperture (closeness).
On the other hand, the gestures with a diagonal or vertical mid-segment tended to be more error-prone which could be a result of an incorrect perception of the proposed gesture by the participant or the effect of the limited flexibility of the thumb.
Conclusion: The hypothesis 3 is verified by the study.

Hypothesis 4: Gestures consisting of only one line are articulated more accurately than composed line gestures.

The single lines were articulated significantly more accurate than the line segments of the composed line gestures. The Orientation error of the single lines was significantly lower than the average orientation error of the composed line segments.
Further, the composed line segments were more bended. Also the average shape error of

the composed line segments was significantly higher than the shape error of the single line segments.
Conclusion: The hypothesis 4 is verified by the study.

Hypothesis 5: The position (articulation order) of a line segment within a gesture is relevant to the articulation accuracy.

The line segments and corners tended to be articulated less accurately depending on their order except for the mid-line segment. We expect that the higher accuracy of the mid-line segment compared to the starting and ending segment depends on the segmentation algorithm which tends to cut closer to the mid-segment. However, comparing the first and the last line segments shows an significantly higher articulation accuracy of the first line segment with respect to the bending error, the shape error, and the orientation error. Also the first corner was articulated more accurately than the second one with a lower CSD and a lower angle error. On the other hand, all line segments were articulated with nearly the same length which was not depending on the length ratio of the proposed gesture. Therefore we recommend the use of equal length lines when designing composed line gestures. However, the difference of the corners articulated accuracy was not that high as assumed.
Conclusion: The hypothesis 5 is verified by the study for the bending error, the shape error, and the orientation error, but not for the length ratio error.

4.4. Summary

This Chapter explained the apparatus, design, procedure, and the results of the control study. The objective is to analyze the articulation accuracy in a control environment (controlled experiment). This is done by excluding external influences and reducing distractions. The study focuses on single lines and composed lines which are varying in orientation and the angle between the lines. Also, all gestures used in the large scale study (game) are tested to make the studies comparable. This enables the analysis of external influences.

The accuracy of the single line gestures was not significantly depending on the lines orientation. However, there was a tendency for a varying accuracy depending on the gesture orientation relating to the type of shape and the orientation accuracy.
Composed line gestures lead to significantly decreased articulation accuracy with respect to the orientation and the angles within the gestures. Regarding to the orientation, the composed gestures with a horizontal mid segment were performed more accurately compared to other orientations. The sharp angle within the gesture tended to be articulated more accurately than the obtuse angles with respect to the angle error but

less accurately relating to the CSD and the length ratio error.
Also the order of articulating the line segments showed to influence its articulation accuracy. Comparing the single line gestures and the segments of the composed line gestures there was a higher articulation accuracy for the single line gestures.

5. Game Study

In order to investigate how the distraction of users influences their articulation accuracy, we ran a large scale study. This Chapter describes the large scale study which was conducted by publishing a mobile game. First, Section 5.1 describes the game principle and how the gestures were logged. After that, Section 5.2 presents the results of the gesture analysis. Last, Section 5.3 discusses the results by focusing on the investigation hypothesis.

5.1. Method

We wrapped our study design into a game for mobile devices. First, this enables the use of an app store (Google Play Store) which reaches many android users. Second, the large scale study has a higher external validity than a control study and therefore impact of external influences on the gesture performance is included in this study. These influences are relevant for the user's touch behavior (cf. Tu et al. [TRZ15]).

This describes the game apparatus which we developed to investigate the articulation accuracy of unistroke gestures starting with the conceptual game design in Section 5.1.1. Next, the logging of the user's touch gestures will be described in detail (Section 5.1.2), following by the used gesture sets in Section 5.1.3. Last, the implementation of the game will be explained in Section 5.1.4.

5.1.1. Game Description

The aim of the game apparatus is to understand the user's touch accuracy on mobile devices depending on the gestures features and external conditions. In particular, we focus in the relation of the user's accuracy to the gesture's orientation and the angles within the gestures. Furthermore, the effect of learning gestures over time is analyzed in this context. In the following we describe, how this study requirements are met using a mobile game as apparatus.

Gameplay

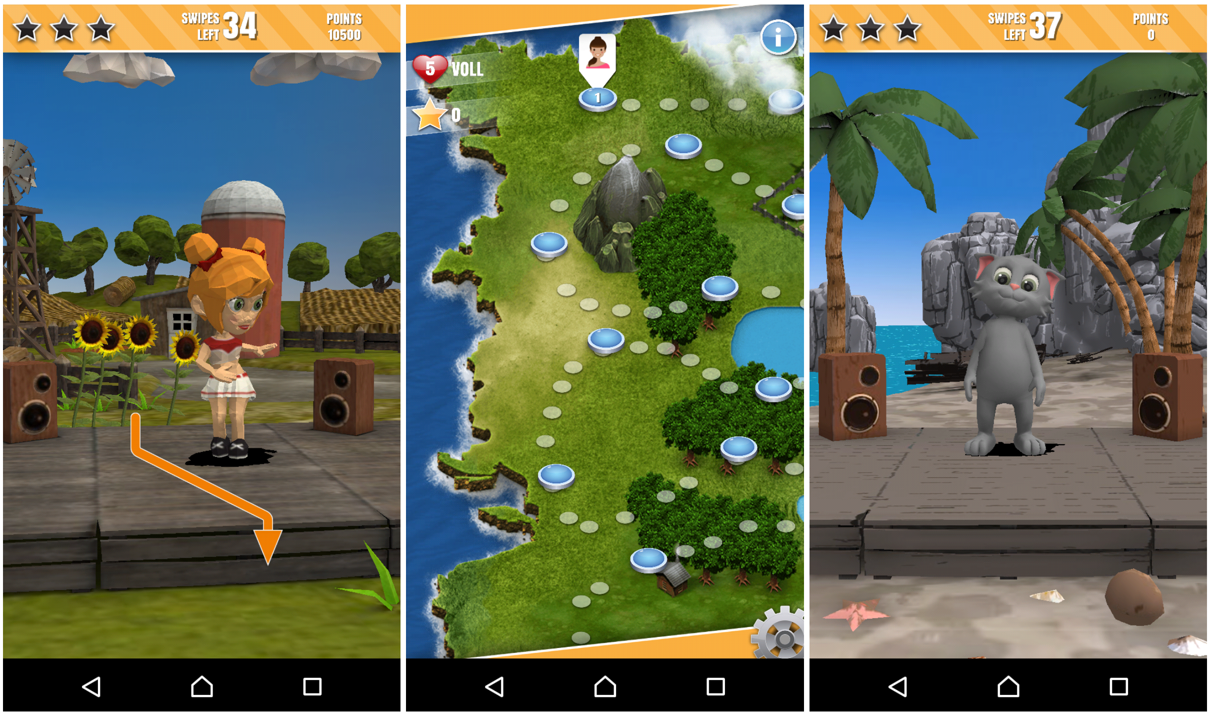
We use a method which was previously used by other authors (cf. Tu et al. [TRZ15]) for testing the user touch behavior. Such method shows the requested gesture graphically on the screen and then has to be performed by the participant.

This process includes the visual perception (see gesture), cognitive effort (plan the gesture execution), and the motor control (articulate the gesture). This is an outside-in process while the stimulus comes from the visual perception in contrast to inside-out studies where the memorability of gestures is tested (cf. Zhai et al. [ZK12]). Using this method, the gesture perception, the motor limitations, and learning effects can be discovered. The analysis of gesture memorability is not included in this study design.

The principle of showing a gesture and then let the participant articulate it is the base for the gameplay design. Because this study procedure is not that entertaining for the participants the study has to be adjusted. In short, the app is a dance game, where the player gets rewarded for successfully articulated gestures. This is done as follows: Starting from the top of the screen, a gesture image “falls” to the bottom of the screen. An arrow indicates the direction in which the gesture has to be drawn. Now, the player has to articulate the gesture on the screen. If the player articulates the gesture successfully, which means that the articulated gesture is similar to the proposed gesture, the player gets rewarded by a dancing character (visual feedback) and the gesture disappears. The similarity of the articulated gesture and the proposed gesture is defined by the shape distance. After that, the next gesture appears at the top of the screen. If the gesture is not performed successfully by the player, the gesture image judders (visual feedback) and the device vibrates (haptic feedback). Now the player has the chance to retry the gesture or wait until the gesture reached the bottom and the next one appears. The game consists of several levels, where each level has a predefined amount of gestures and a limited number of attempts to articulate a gesture. Each time the user articulates a gesture, whether it’s successful of not, the number of attempts is decreased by one. The game is finished if no attempts are left. After each successful level, the user has to do a short survey about his satisfaction and how he feels about the graphical conditions. After that, a summary of the level’s achievements is shown. Figure 5.1a shows the basic principle of the game.

Levels

Each level is defined by a set of gestures, the number of attempts and graphical conditions. If the player successfully finishes a level by achieving one star, he unlocks the next level. With the increasing number of unlocked levels also the gestures are getting more complex. The gesture complexities are defined as described by Vatavu et al. [Vat+11]. This challenges the player while he gets better when playing the game for a longer amount of time. The gestures in each level are randomly permuted so the order of the



(a) Principle of the game. The proposed gestures are falling from the top to the bottom.

(b) The level map. The levels consist of different gesture sets.

(c) The characters and the backgrounds are changing in different levels.

Figure 5.1.: The game principle (a), the level map screen (b), and changing character and background (c).

gestures has no influence to the user performance. The speed of the falling gestures should be selected moderate so that the player will not get in a hurry but also will not get bored. Furthermore, the player is't rewarded if a gesture was performed fast. So the time factor is not relevant to the gesture articulation. Furthermore, splitting the game into levels leads to a short break for the player after each level which reduces the fatigue effect.

The graphical conditions are changing from level to level. The game consists of different stages and characters. This allows the analysis of the user's performance independently from the visual appearing of the game. One can argue that a player has the intention to perform better if a character is more appealing to him than another one. Furthermore, the quality of the 3D scenes changes in terms of texture quality and the polygon quality (polygon count). The user also has the option to customize the app by disabling the acoustic and haptic feedback as well as the background music. However, these settings have to be considered when analysing the touch data.

Player Motivation

In order get the player motivated to articulate the gesture's accuracy the player will be rewarded by visual feedback. Furthermore, a score will be increased for each successfully articulated gesture. Depending on the shape distance between the articulated gesture and the proposed gesture the increase of the score can be higher or lower. The game differs between false, ok, good, and excellent performed gestures. Another representation of the score is a star count. If the user reaches at lest one star, the level was successfully completed and the next level will be unlocked. Otherwise, the player has the opportunity to retry the level. To prevent the player from retrying the level many times without trying to get better, the player loses a life (using a heath metaphor) if he fails at a level. If all hearts are lost, the player has to wait for 30 minutes or do a survey to get more hearts. Furthermore, the player is shown the maximum number of successfully performed gestures in a row. To get as much gesture samples as possible and to analyze the players learning effect over the time, the player has to be motivated to play the game over a longer period. Therefore, the game provides a global high score in which the player can compare his level score to his friends level scores. If a player wants to improve his score of a level, he is free to retry every level that he has played before by providing a level map. Furthermore, the player is motivated by the fact that he is able to unlock new stages an characters. These described factors should ensure the overall motivation and satisfaction while playing the game.

5.1.2. Logging

In order to analyze the user touch behavior, its necessary to log the touch events during the game and also information according to the device and the player's success. Therefore, the user has to be informed about the data collection which is done at the first start of the game like recommended by Henze et al. [PH11]. This Section presents in particular the collected information and their use.

Each log contains the device's android-id which is mandatory to assign the logs to a specific device. Additionally, a timestamp gives the information when the log was created. Furthermore, the facebook-id is logged which is needed to save the user's scores and analyze the user's behavior if he changes the device.

Gesture Log. The log of each gesture contains an ordered list of each touch point while articulating a gesture. The touch point is described by the touched coordinates x and y, a timestamp, and the pressure. By logging these information, we're able to discover the accuracy and the speed while articulating a gesture. By considering the timestamp, also the learning effect can be analyzed if a gesture is repeated many times. Furthermore, the gesture log contains the id of the shown proposed gesture, the score of

Log	Description
Gesture Log	Describes a touch event by the player
Device Log	Describes the device parameters
Settings Log	Describes the user's acoustic and haptic settings
Survey Log	The answers of the questionnaires
Level Log	The users' success after playing a level

Table 5.1.: Logged information while playing the game.

the gesture recognizer (shape distance) and the time needed to recognize the gesture by the algorithm. Additionally, the framerate while articulating a gesture is recorded to exclude juddering gestures which can confuse the player. Last, the gesture log provides information about the level's graphical conditions (polygon quality and texture quality) and the actual sound volume of the device.

Device Log. The log of the device information is created every time the app starts to guarantee that the device's information are up-to-date (in terms of the android version). Also relevant to the gesture log are information about the display size. This should be considered when analysing the gesture logs with respect to the screen size. A listing of all information logged can be found in Table 5.1.

Level Log. The level log describes the success of a finished level. This includes the level score, the reached stars, and the successful gestures in a row. Furthermore, the log gives information about how active the player is using the app and analysing the improvement when retrying levels. Furthermore, by logging the level-id, the graphical conditions as well as the shown stages and characters can be considered during the analysis. Last, the star count is used to provide a global high score.

Settings Log. The shows when the player changes the default settings for the acoustic an haptic feedback as well as the background music.

Survey Log. The survey log captures a user's answer to a survey question. The questions are asked in a way that they can answered by using a slider. So the answer is represented by a value between 0 and 1. Depending on the question, the answer gives information about the users satisfaction while playing the game or how he likes the graphical conditions.

5.1.3. Gesture Sets

The gesture sets, which are used for this game, are created and chosen with respect to analyze the user's accuracy depending on the gesture's orientation and the angles within a gesture. Furthermore, we investigate if there's a relation to the accuracy of a gesture's segment, depending if the segment is at the beginning, in between, or at the end of a gesture.

Single Lines. First, the game contains a gesture set consisting of rotated lines. There are 360 lines which differ in an angle of 1°. These gestures are used to analyze the user's accuracy depending on the gesture's orientation. These gestures are intentionally chosen simple so we're able to discover the accuracy of the line segment independently from the position inside a gesture. Furthermore, the measured accuracy is the benchmark for the accuracy of more complex gestures. This is based on the assumption that a user cannot reach a higher accuracy when articulating a composed gesture compared to his performance of single line gestures.

Composed Lines. This gesture set contains gestures consisting of three line segments. Using this, were able to analyze the accuracy of line segments inside composed gestures. The whole gestures are rotated from 0° to 360° in 15° steps. Furthermore, the starting lines and the end lines are rotated by 180° in 15° steps. This leads to a total number of $(24 \times 11 \times 11)$ 2904 gestures.

Sets from other Studies. We make use of gesture sets from previous gesture studies. This includes the gesture set of Appert and Zhai (cf. [AZ09]), which was used for gesture shortcuts. Second, we added the gesture Grafitti unistroke alphabet which was created for Palm OS. Last, the test set of the $-Algorithm was included in the game (cf. Wobbrock et al. [WWL07]). The latter two gesture sets were also used for analysis of a recognizer by Kristensson and Denby [KD11]. Some selected gestures from theses sets were also varied in curvature, length, and ratio. Additionally, some gestures were rotated in steps of 90°.

Besides the testing of the described variations, the additional gesture sets are selected to challenge the user in terms of the gesture complexity. All gestures are grouped in five complexity classes (cf. Vatavu et al. [Vat+11]). Additionally, the letter two gesture sets (single lines and composite lines) are tested in the control study which is described in Chapter 4. In Figure 5.2 you can see an abstract of the game's gestures.

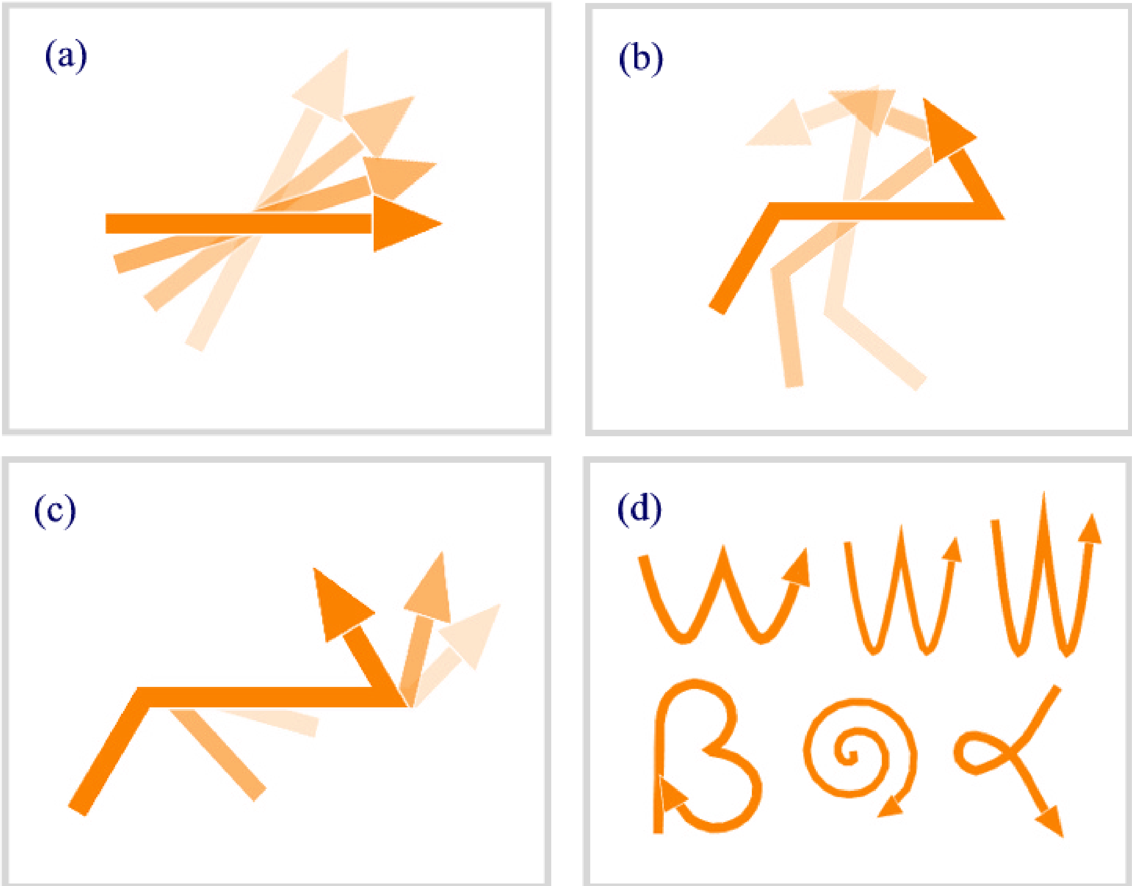


Figure 5.2.: Gesture sets which were used by the control study and the game study. Single rotated lines (a), composed lines varying in their orientation (b) and angles (c), and various gesture sets (e.g. the $-Algorithm test set and Grafitti).

5.1.4. Implementation

This Section describes the implementation of the game as well as the architecture of the logging system.

Architecture. The game is based on the game engine LibGDX which supports the android platform. When the user performs a gesture, a gesture log is created and send to a server via HTTP. If there's no internet connection, the app saves the log in a local database and will be sent at the next app start. On the server side, a PHP script processes the gesture log (and also the other logs described in Section 5.1.2) by saving the log to the server's database. If the database is not available, the server responses the app to save the log locally.

App Implementation. The cross-platform and open-source game engine LibGDX was chosen to support 2D and 3D scenes. The user interface is implemented in 2D scenes except for the game screen which is a combination of a 2D scene (foreground) and a 3D scene (background). The game screens's 2D scene displays the user's score and the gesture images which are drawn from a SVG graphic. By defining the gestures as SVG graphics, the gesture appearance and the point data are described by a single file. This makes the app easily extendable for further gestures. For the 3D scene contains the stage model and a model of the character. The character is moved by several key frame animations. The 3D model is textured dynamically during runtime in order to change the texture quality as needed by the scene and to reduce the apps size for storage. Figure 5.1a shows a selection of the game's user interface including the game screen. All screens as well as the different levels and characters of the game are listed in the Appendix A.

The rating of the user's accuracy when articulating a gesture is done by a modified version of the $-Algorithm. The algorithm was chosen by reasons of simplicity and the requirement to be direction sensitive. The algorithm has to be modified to support 1D gestures (lines) and to be rotation sensitive. Therefore, the gestures are scaled by keeping the ratio and the step of rotating the gestures to its indicative angle is skipped.

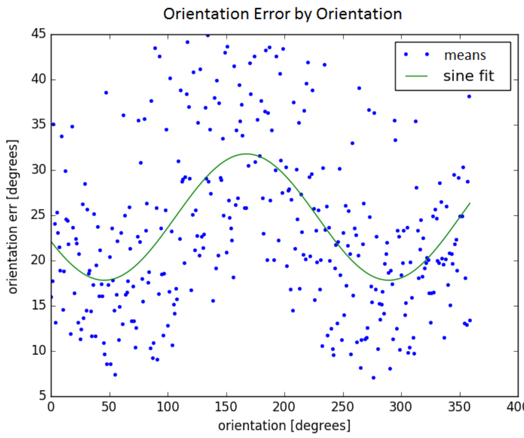
5.2. Results

The game was available on the Google Play Store for 3 weeks. During this time, the game was installed on more than 900 devices and more than 1800 levels were played. The game was played by players of various countries like Germany, USA, Great Britain, and Russia. During this time, over 69000 gestures were logged. After filtering out gestures which were articulated on a frame rate below 60 FPS and short gestures, around 21000 loges were left to be analyzed.

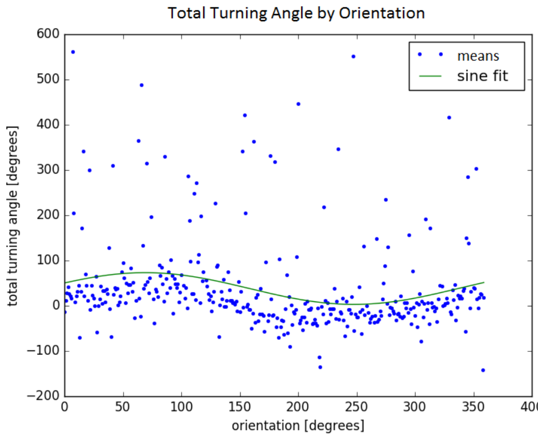
5.2.1. Single Line Gestures Analysis

Table 5.2 reports the single line gesture analysis of the large scale study. Therefore, 2988 line gestures were analyzed.

Gesture Features. The average line length of the single lines was 34.4 mm (σ_{95}=03 mm). In average, the participants articulated the single line gestures with an orientation error of 25.0° (σ_{95}=0.6°). The bending error of the single line gestures was 2.4° (σ_{95}=0.15°). Last, the average total turning angle of the single line gestures was 65.3° (σ_{95}=19.4°).



(a) Sine regression of the orientation error.



(b) Sine regression of the total turning angle error.

Figure 5.3.: Sine regression analysis of the large scale study.

Orientation Dependency. The orientation error made while gesture articulation showed to be affected by the lines global orientation (angle between the line gesture and the x-axis). However, the means of the orientation error differed not significantly. With respect to the sine regression ($R^2 = 0.26$), the participants made the biggest orientation errors when articulating horizontal line gestures (see Figure 5.3a). In these orientations, the orientation error was 7° higher than the regression's mean of 24.7°.
The average line length tended to be higher for diagonal lines in the orientations near 45° and 225°. The sine regression showed a weak correlation (R^2=0.25) for the line length depending on the orientation.

5.2.2. Composed Line Gestures Analysis

As for the control study analysis, it has to be considered that the results of the composed line gesture analysis depend on the accuracy of the segmentation algorithm which has a strong impact on the validity of the analysis. In the following the lines and corners of the composed line gestures are enumerated by their execution order (e.g. line segment two = second executed line segment). Table 5.3 displays the results of the composed lines gestures analysis. Because of the small sample size, the angle and orientation dependent (statistical) analysis of the composed lines gestures was not possible.

Order Dependency. According to the bending error of the line segments, there was no significant difference between the first and the last line segment. On the other hand the average bending error of the mid-line segment was significantly lower compared to the other line segments (p<0.001). Also, the shape error of the mid-line segment was

Metric	Mean	σ_{95}	Sine Regression
total turning angle	65.3°	19.4°	M=38.0° A=35.1° T=357.2° PH=20.3° $R^2 = 0.05$ (weak)
sharpness	653.2 $degrees^2$	95.0 $degrees^2$	N/A
line length	34.4 mm	0.3 mm	M=34.3 mm A= 2.5 mm T=238.1° PH=125.2° $R^2 = 0.25$ (weak)
b. box dia. angle error	13.3°	0.2°	N/A
b. box ratio error	0.0	0.0	N/A
shape error	0.0039 mm	0.0001 mm	N/A
shape variability	0.0024 mm^2	7.2e-5 mm^2	N/A
relative aperture error	0.098	0.004	N/A
bending error	2.4°	0.15°	N/A
bending variability	6.0 $degrees^2$	0.2 $degrees^2$	N/A
indicative angle error	22.3°	0.5°	N/A
line orientation error	25.0°	0.6°	M=24.7° A=7.0° T=243.7° PH=202.5° $R^2 = 0.26$ (weak)

Table 5.2.: Aggregated metrics of single line gestures. Each metric was computed with a sample size of 15081 lines. N/A is used for $R^2 < 0.1$ or the metric was not analyzed in detail.

significantly lower than the other line segments. Analyzing the order of the articulated corners, there was no significant difference in the CSD as well as the angle error between the line segments. Also the orientation error differed not significantly between the line segments. In contrast, there was a difference in the average line segment length ratio error depending depending on the segments' position. The second line segments had a significantly higher length ratio error compared to the first and the third line segments. This is explained by the fact that the mid-line segment of the proposed gesture is as double as the starting and ending segment but the articulated line segments had the similar length.

Metric	Gesture Segment	Mean	σ_{95}
len. ratio error	line segment 1	0.334	0.006
len. ratio error	line segment 2	0.217	0.008
len. ratio error	line segment 3	0.150	0.007
len. ratio error neigh.	line segment 1 and 2	1.126	0.177
len. ratio error neigh.	line segment 2 and 3	1.360	0.261
angle error	∢ corner 1	44.8°	2.7°
angle error	∢ corner 2	41.7°	2.6°
orientation error	line segment 1	36.0°	2.8°
orientation error	line segment 2	41.0°	3.0°
orientation error	line segment 3	36.0°	2.8°
shape error	line segment 1	0.0059 mm	5.4e-4 mm
shape error	line segment 2	0.0033 mm	4.1e-4 mm
shape error	line segment 3	0.0050 mm	4.6e-4 mm
bending error	line segment 1	1.9°	0.4°
bending error	line segment 2	0.9°	0.2°
bending error	line segment 3	1.5°	0.4°
CSD	corner 1	0.018 mm	8.8e-4
CSD	corner 2	0.019 mm	8.8e-4

Table 5.3.: Aggregated metrics of composed line gestures (958 samples for each line segment). Because of the small sample size, the column *orientation dependency* and *angle dependency* from the control study analysis are removed.

5.2.3. Composed Line Gesture vs. Single Line Gesture

The length of the single lines was nearly as double as the length of the composed line segments. With an average of 34.4 mm, the single lines were articulated 17 mm longer than the line segments of the composed gestures.
The single lines were articulated with a smaller orientation error than the line segments. While the average orientation error of the single line gestures was 25.0° (σ_{95}=0.6°), the average orientation error of line segment one was 36.0° (σ_{95}=2.8°), line segment two was 41.0° (σ_{95}=3.0°), and line segment three was 36.0° (σ_{95}=2.8°).
According to the shape error, the users tended to articulate the single line gestures more accurate than the first and the second segment of the composed line gestures. In contrast, the line segments two and three of the composed gestures showed to be significantly less bended than the single line gestures ($p<0.001$).

5.3. Discussion

In this Section we discuss the results of the large scale study with focus on the investigation hypothesis' defined in Section 3.2. Because of the small amount of collected composed line gestures the hypothesis 2 and 3 are not discussed. However, they are discussed in the previous Chapter (see Subsection 4.3). Hypothesis 6 which focuses on the users distraction is discussed in Section 6.2.

Hypothesis 1: The orientation (global rotation) of single line gestures affects the articulation accuracy.

During articulating the single line gestures the users made an average an orientation error of 25.0°. Depending on the orientation of the line gestures the error varied by 7.0°. Horizontal lines in left direction showed to be the most accurate. However, the weak correlation of the sine regression showed that the orientation error was less sensitive to the line gesture orientation than assumed.
The bending of the lines showed to be not affected by the lines orientation. On average the lines were bended by 2.4°. However, the moderate sine correlation of the total turning angle showed that the shape of how the lines are bended instead of the bending intensity depended on the line orientation. This is because the total turning angle is sensitive to the bending direction. We assume that the low turning angle but nearly constant bending error for horizontal lines indicates "S"-shaped lines while the high values of the total turning angle indicated the articulation of arc-shaped lines. The variation of the shape type but constant bending depending on the line orientation is also shown by a nearly constant shape error for rotated line gestures.
Conclusion: The hypothesis 1 is statistically not verified by our study. We assume that this is caused by the strong variation in the articulation behavior between the participants.

Hypothesis 4: Gestures consisting of only one line are articulated more accurate than composed line gestures.

As assumed, the single line was articulated more accurately comparing to the line segmets of the composed line gestures. The shape error of the single line segments is significantly lower than the shape error of the composed line segments which is explained by the additionally articulated arc which connects the line segments (instead of a sharp corner). In contrast, the second and third line segment showed to be less bended than the single line segment.
The single lines showed to be articulated more accurately compared to the composed line segments with respect to the orientation error. We assume this is the result of the more complex shape of the composed line gestures comparing to the simple single line gestures.
The single lines were articulated significantly longer compared to the composed line

segments. Because we analyzed the accuracy independently from the absolute gesture sized, this finding is not related to the gestures accuracy.
Conclusion: The hypothesis 4 is verified by the study.

Hypothesis 5: The position (articulation order) of a line segment within a gesture is relevant to the articulation accuracy.

There was a significant difference in the shape error and the bending error between the mid-line segment compared to the start and end segments. We assume that this result is explained by the gesture segmentation rather than by the user articulation behavior. Comparing the first and the last line segment with respect to the bending and shape error shows no significant difference.
The users tended to articulate all line segments with a similar length. This effect lead to an significantly higher length ratio error for the mid-line segment which had the double length than the other line segments.
The accuracy of the articulated corners was not depending on the order of them (with respect to their shape error). Also, the angle error between the line segments was independent of the order in which the line segments were articulated.
Conclusion: The hypothesis 5 is verified by the study.

5.4. Summary

This Chapter described the game apparature and the logging of the user's gestures as well as the results of the large scale study. During the game, the player has to articulate gestures which are falling from the top to the bottom of the screen. The player is rewarded by a dancing character animation and a higher score for accurate articulated gestures. The app logs the users touch behavior to the server, where it's stored for analysis. Additionally, technical device specific information and survey questions are logged. Each level is defined by its gesture set and its graphic conditions like changing levels, characters and model qualities. Each level gesture set is a combination of gestures from five complexity classes and the amount of more complex gestures is increasing with the level number. The gesture sets consist of lines, composed lines and gestures which where used in previous studies. The gestures are varying in orientation, angles, and curviness in order to analyze the users gesture accuracy depending on these parameters.

Analyzing the gesture logs showed that single lines tended to be articulated more accurate than the lines which were composed of three line segments. On the other hand, the orientation of the single lines had no significant effect on the articulated accuracy but there was a tendency for vertical lines to be the most accurate and vertical lines less

accurate.
The order in which the composed line segments were articulated affected the articulation accuracy in terms of the bending the first and the last line segment which connects the mid-line segment with an arc. Also all line segments were articulated with the similar length which resulted in an length ratio error.

6. Study Comparison

In this Chapter the results of the large scale study (game study) are compared to the findings of the control study. In particular we focus on how the external influences affected the users' accuracy when playing the game during the large scale study compared to the control study.
First, Section 6.1 presents the results of comparing the two studies. After that, Section 6.2 discusses the these findings with respect to the investigation hypothesis.

6.1. Results

In the following, we present which metrics differed significantly and which metrics were not affected by external influences and the users' distraction.

Single Line Gestures

Regarding the bending of the single line gestures, the gestures of the large scale study were articulated with a bending error of 2.4° while line gestures of the control study were bended by 1.0°. The lower accuracy of the game gestures is also shown by the increased shape error. With a shape error of 0.0039 mm the error was more than double high compared to the control study (0.0017 mm)
Furthermore, the users articulated the line gestures with a significantly ($p<0.001$) higher angle error (35.0°) compared to the participants of the control study (10.4°). Focusing on the gestures' length features, the gestures articulated during the game had path length of 87.7 mm and the lines of the control study were articulated with a path length of 22.3 mm. Also the line length (distance between the starting and the ending point) was significant longer for the articulated line gestures of the game study (34.4 mm during the game study and 21.7 mm during the control study). Our assumption is that the lines were articulated longer because the users may also have used their index finger as implement.

Composed Line Gestures

Gesture Features and Accuracy. There was a significant difference in the bounding box ratios error (difference between the task axis' bounding box ratio and the articulated

bounding box ratio). The gestures of the game study were articulated with an error of 1.0 and the game gestures were articulated with an error of 0.5.
In the game study the gestures were articulated with nearly three times the path length compared to the control study (game study: 137.7 mm, control study: 44.9 mm)

Analyzing the gestures bending features showed that the gestures articulated during the game study were bended by 1.1° more compared to the control study ($p=0.004$). Also, the bending variability differed significantly ($p<0.001$).
Furthermore, the total turning angle of the gestures articulated during the large scale study was significantly ($p=0.009$) higher (118.8°) than during the game study (12.3°). Relating to the orientation errors, there was the trend that the game gestures were articulated less accurate than the control study gestures. The gesture orientation error (indicative angle) differed by 10° and the error in orientation of the line segments differed significantly. We assume that the error in the indicative angle of the control study gestures can be also caused by the difference in their shape.

Gesture Segments Features and Accuracy.

The comparison of the gesture features and errors of the single line segments showed that the first and the second line segments were articulated less precisely during the large scale study than during the control study. This is justified by a 20° higher orientation error for the first line segments, 20° higher error for the second line segments, and 13° higher error for the third line segment. Additionally, the shape errors of the first line segment and the second line segment were significant greater compared to the control study's shape errors. Also the bending errors of the first and the second line segments were higher during the game study. Surprisingly, the last line segments were articulated more accurately during the large scale study with respect to the average bending error and the average shape error. This can also be the effect of a sharper articulated second corner. In contrast, the orientation error of all line segments was significantly higher for the gestures of the game study. Figure 6.1 shows the differences of the articulation accuracy for both studies.

Focusing on the corner segments, the users articulated the first and the second corners less accurate during the large scale study than during the control study.
Also the average angle error between the first and the second line segments was 13° higher for the gestures articulated during the game study. The angle error between the second and the third line segment was 9° during the game study.

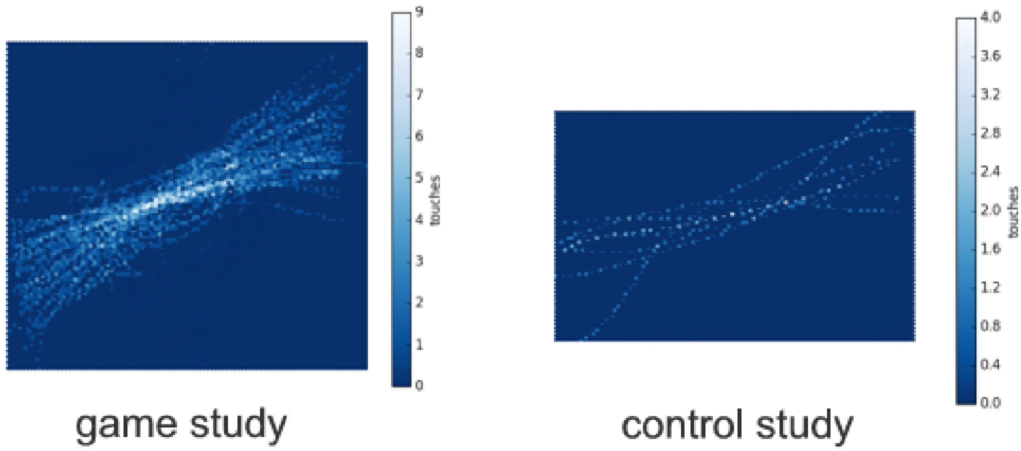


Figure 6.1.: Heat map comparison of the composed line gestures articulated during the large scale study and the control study. The intended orientation of the gesture is 188°.

6.2. Discussion

In this Section, the results of the study comparison is discussed with respect to the Investigation *Hypothesis 6: The accuracy of the articulated gestures is affected by external influences (distraction)*.

As expected, there was a significant difference in the users' accuracy between both studies. The participants tended to articulate the single line gestures and the composed line gestures less accurate during the game study than in the control study. We assume that the lower accuracy of the gestures performed during the large scale is the effect of the users' distraction. The lower accuracy is shown by an increased shape error, bending error, and orientation error for both gesture types. Further we assume that the varying errors between the studies are also an effect of the use of different implements (index finger or thumb) or the use of the left or the right hand.
Conclusion: The hypothesis 6 is verified by the comparison of both studies.

7. Conclusion

In this work we analyzed users' touch accuracy by running a large scale study using a mobile game and a control study. In particular, we focused on single line gestures and composed gestures consisting of straight lines which varied in their orientation (rotation) and the angles within the gestures. The articulated gestures were analyzed with respect to their accuracy in bending, shape, orientation (rotation), and angle. Furthermore, we analyzed how the variation of the orientation and the angles within the gestures affected the users' articulation accuracy. Both studies were designed in a manner which included the users' perception of the gestures and their reproduction.

There was the need for two studies because we aimed to investigate how accurately the users are able to reproduce the gestures without distraction compared to their touch accuracy when being in their natural environment.
First, the controlled lab study was designed for providing a high internal validity. Each participant had to reproduce 1036 gestures which were falling from the top to the bottom of the screen. To provide the internal validity, the participants had to perform in sitting position and only the input by the right thumb was allowed. Nine participants took part in the control study and 10631 gestures were recorded for analysis.
The principle of the falling gestures was also applied in the large scale game study. The users were rewarded for accurately articulated gestures by an animation of various dancing characters and an increased game score. Additionally, the users were given a visual and haptic feedback if they articulated a gesture inaccurate. The game included of 300 levels which consisted of a varying amount and complexity gestures to be performed and also changing visual backgrounds. The aim of the game study was to provide a higher external validity compared to the control study in order to be able to analyze the effect of distraction on the accuracy of articulating gestures.
The game was published for mobile Android devices on an app store. After three weeks, more than 900 smart phone users have downloaded the game and they played 1800 levels. After filtering the gestures which were articulated during a low frame rate and wrongly articulated gestures, there were 21000 gestures left to be analyzed.

The analysis of the control study showed that the participants made an average error of 9.7° in the gesture orientation and a bending of 0.97° when they performed single line gestures.

The accuracy of the composed line gestures depended significantly on their orientation and the angle within the gestures. The 180° rotated gestures (horizontal mid-line segment with right to left movement) showed to be the most accurately articulated while the gestures with a diagonal mid-line segment near 45° and 225° were articulated significantly less accurate with respect to their angles accuracy, bending, and shape error. The analysis of varying angles within the gesture showed that sharp angles were articulated more accurately with respect to the corners shape errors (CSD) but more accurate in the angle and the length ratio error compared to obtuse angles. Further, the first line segments and corners were articulated more precisely than the last ones.
The analysis of the game study showed a similar user behavior for single line gestures depending on the lines orientation. However, the users' average error in the orientation was significantly higher compared to the gestures articulated during the control study. Also, the composed line gestures tended to be articulated less accurately during the game study. However, because of the few recorded composed line gestures an analysis of the accuracy depending on their orientation and angle was not possible.

7.1. Future Work

While this work focused on analyzing the accuracy of articulated unistroke gestures consisting of straight lines, there will be the need for analyzing further kinds of gestures. This includes gestures consisting of only arcs of different bending and also the combination of arcs and straight lines. Because of the easy extensibility of the study apps, these gestures can be added to apparatus of the game study and the control study. Analyzing these gestures aims to identify how additional gesture features, like the bending factor of the composition of different segment types, affect the users' articulation accuracy. Also, running the study with another game concept should prove that the results of this work are independent from the users' satisfaction and individual appeal of the game.

The results of this work and the described further studies can then be applied in a predictive accuracy model for gesture sets. Reviewing the previous gesture models (see Subsection 2.1.2 and 2.2.1) showed that there were investigations in predicting the gestures visual similarities, production time, and complexity but not the predicted accuracy by the user so far.

Finally, changing the game principle in a way that the users have to memorize and guess gestures considers all main factors that are included in the gesture articulation process. Like previous work noted, there is the need for a bridge between gestures which are easy to articulate and easy to remember because these factors are conflicting mostly (cf. Zhai and Kristensson [ZK12]). Extending the previous described model by the factors learnability and guessability (cf. Wobbrock et al. [Wob+05]) enables gesture designers to test designed gesture sets for their predicted in guessability, memorability, and articulation accuracy. Furthermore this model can be applied in a gesture design tool which can optimize gestures with respect to these features which can give an option for selecting optimized gesture sets. Thus a future gesture designer may be able to create fast, accurate, easy to remember, or easy to guess gestures depending on the use cases of the application.

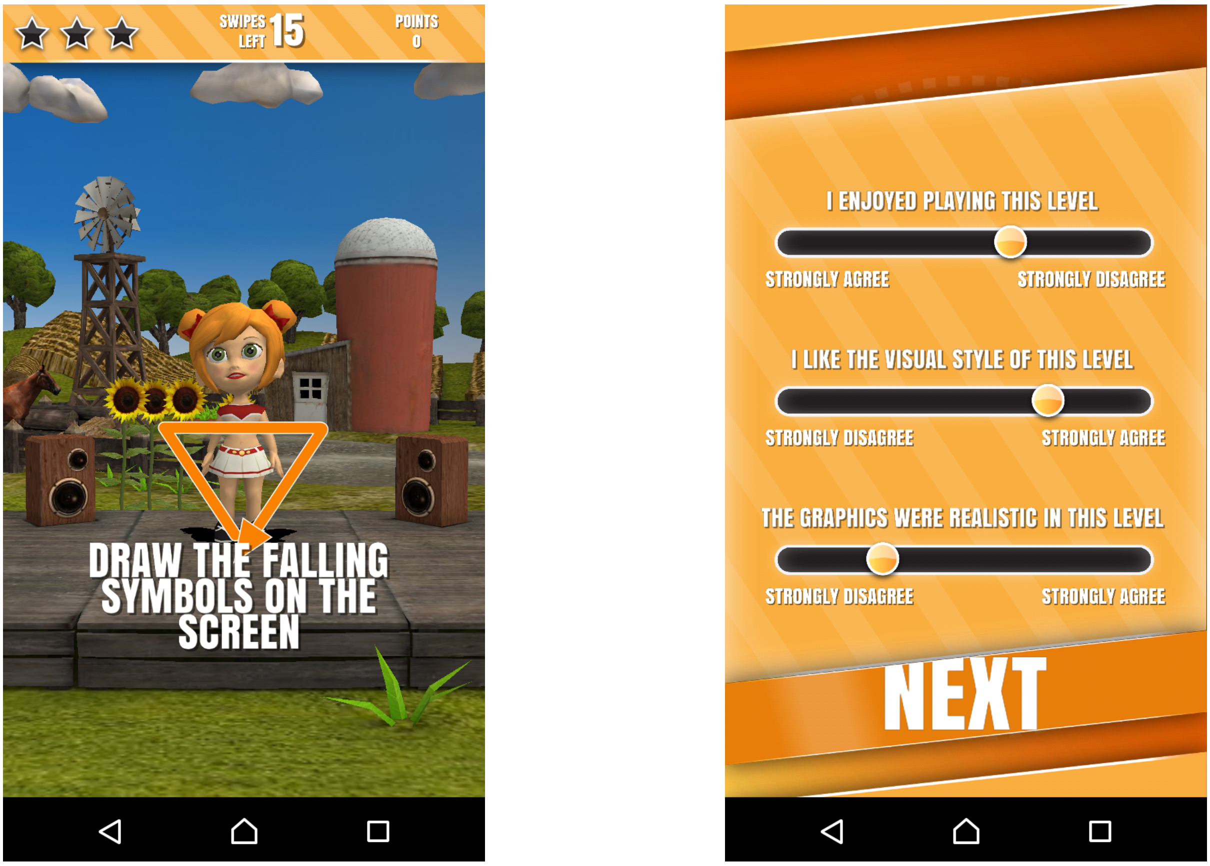
A. User Interface

Game Interface

(a) The german version of the start menu screen. This screen appears after the app is launched.

(b) The level map screen. The blue dots are levels which have to be unlocked by the user.

Figure A.1.: The start menu screen and the level map screen.

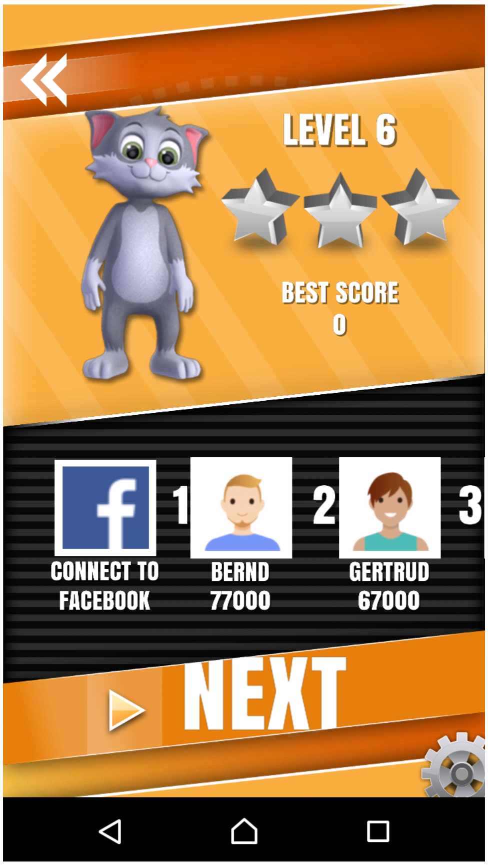


(a) The tutorial screen. This level has to be done by the user if he plays the game for the first time.

(b) The questionnaire which appears if a level was successfully finished.

Figure A.2.: The tutorial screen and the questionnaire screen.

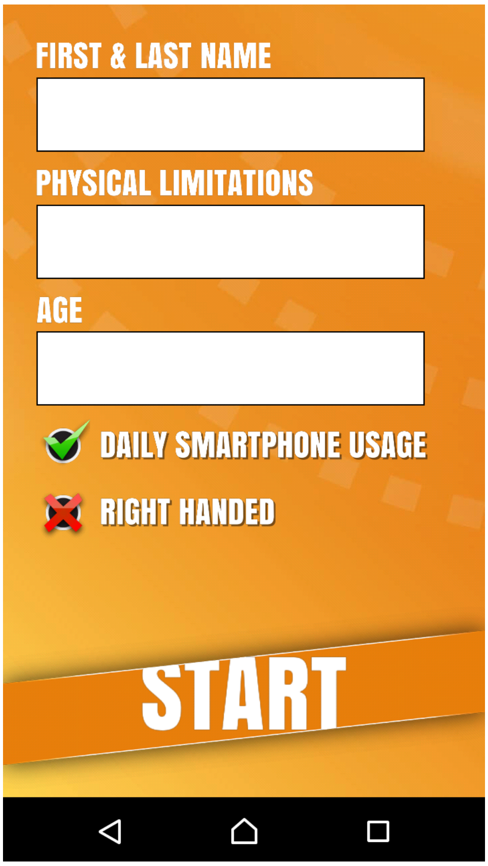
(a) The robot character in the dessert level.



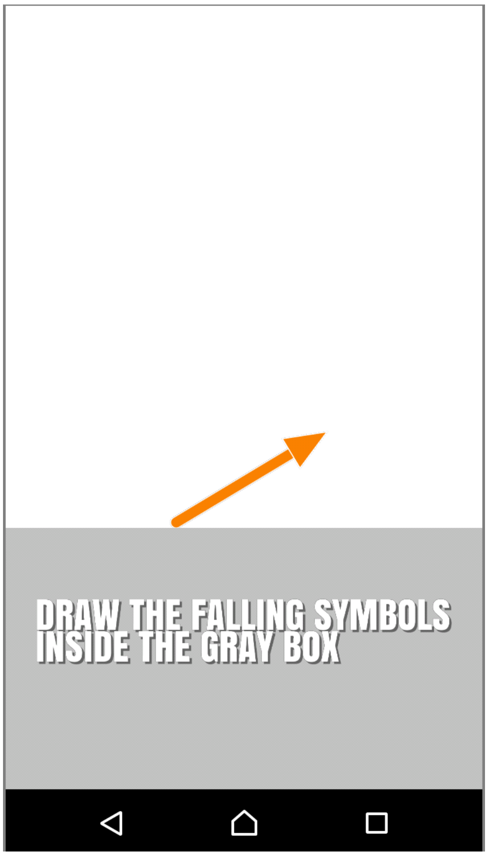
(b) The level introduction screen. This screen is shown before a level starts.

Figure A.3.: The robot character and the level introduction screen.

Control Study Interface



(a) The questionnaire which is shown before the control study starts.



(b) The study tutorial screen. This level explains the control study.

Figure A.4.: The control study questionnaire and the tutorial screen.

Bibliography

[AVW13] L. Anthony, R.-D. Vatavu, J. O. Wobbrock. "Understanding the Consistency of Users' Pen and Finger Stroke Gesture Articulation." In: GI '13 (2013), pp. 87–94 (cit. on pp. 18, 21, 25).

[AZ08] T. H. Andersen, S. Zhai. ""Writing with Music": Exploring the Use of Auditory Feedback in Gesture Interfaces." In: *ACM Trans. Appl. Percept.* 7.3 (June 2008), 17:1–17:24. ISSN: 1544-3558. DOI: 10.1145/1773965.1773968 (cit. on p. 21).

[AZ09] C. Appert, S. Zhai. "Using Strokes as Command Shortcuts : Cognitive Benefits and Toolkit Support." In: *Proceedings of the 27th international conference on Human factors in computing systems (CHI'09)* 2009 (2009), pp. 2289–2298. DOI: 10.1145/1518701.1519052 (cit. on pp. 20, 25, 68).

[CZ07] X. Cao, S. Zhai. "Modeling Human Performance of Pen Stroke Gestures." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)* (2007), pp. 1495–1504. DOI: 10.1145/1240624.1240850 (cit. on p. 18).

[HRH07] G. A. ten Holt, M. J. Reinders, E. Hendriks. "Multi-dimensional dynamic time warping for gesture recognition." In: 300 (2007) (cit. on p. 26).

[Har06] S. G. Hart. "NASA-task load index (NASA-TLX); 20 years later." In: 50.9 (2006), pp. 904–908. DOI: 10.1177/154193120605000909 (cit. on p. 45).

[Hen11] N. Henze. "Release Your App on Sunday Eve : Finding the Best Time to Deploy Apps." In: *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)* (2011), pp. 581–586. DOI: 10.1145/2037373.2037464 (cit. on p. 28).

[Iso01] P. Isokoski. "Model for Unistroke Writing Time." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '01. Seattle, Washington, USA: ACM, 2001, pp. 357–364. ISBN: 1-58113-327-8. DOI: 10.1145/365024.365299 (cit. on pp. 18, 19).

[KD11] P. O. Kristensson, L. C. Denby. “Continuous recognition and visualization of pen strokes and touch-screen gestures.” In: *Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling - SBIM ’11* 1 (2011), p. 95. DOI: 10.1145/2021164.2021181 (cit. on pp. 26, 68).

[LFL14] H. Lü, J. A. Fogarty, Y. Li. “Gesture Script: Recognizing Gestures and Their Structure Using Rendering Scripts and Interactively Trained Parts.” In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems* (2014), pp. 1685–1694. DOI: 10.1145/2556288.2557263 (cit. on p. 27).

[LLR01] A. C. Long, J. A. Landay, L. A. Rowe. “"Those Look Similar!" Issues in Automating Gesture Design Advice.” In: *Proceedings of the 2001 Workshop on Perceptive User Interfaces*. PUI ’01. Orlando, Florida, USA: ACM, 2001, pp. 1–5. DOI: 10.1145/971478.971510 (cit. on p. 25).

[LLR99] A. C. Long, J. a. Landay, L. a. Rowe. “Implications For a Gesture Design Tool.” In: *ACM, Computer-Human Interface, CHI ’99: Proceedings of the SIGCHI conference on Human factors in computing systems* (1999), pp. 40–47. DOI: 10.1145/302979.302985 (cit. on pp. 18, 23).

[Li10] Y. Li. “Protractor: a fast and accurate gesture recognizer.” In: *Proceedings of the 28th international conference on Human factors in computing systems* (2010), pp. 2169–2172. DOI: 10.1145/1753326.1753654 (cit. on p. 26).

[Lon+99] a. C. Long, J. a. Landay, L. a. Rowe, J. Michiels. “Visual Similarity of Pen Gestures.” In: *Conference on Human Factors in Computing Systems, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* 2.1 (1999), pp. 360–367. DOI: 10.1145/332040.332458 (cit. on pp. 19, 38).

[MS10] A. Myronenko, X. Song. “Point Set Registration: Coherent Point Drift.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.12 (2010), pp. 2262–2275. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2010.46 (cit. on p. 41).

[Nac+13] M. A. Nacenta, Y. Kamber, Y. Qiang, P. O. Kristensson. “Memorability of Pre-designed and User-defined Gesture Sets.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 1099–1108. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2466142 (cit. on p. 24).

[OF13] U. Oh, L. Findlater. “The Challenges and Potential of End-user Gesture Customization.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 1129–1138. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2466145 (cit. on p. 24).

[PH11] B. Poppinga, N. Henze. "My App is an Experiment : Experience from User Studies in Mobile App Stores." In: *International Journal of Mobile Human Computer Interaction* Volume 3 I (2011), Pages 71–91. ISSN: 1942-390X. DOI: 10.4018/jmhci.2011100105 (cit. on pp. 28, 66).

[Rub91] D. Rubine. "Specifying Gestures by Example." In: *Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '91. New York, NY, USA: ACM, 1991, pp. 329–337. ISBN: 0-89791-436-8. DOI: 10.1145/122718.122753 (cit. on pp. 25, 37, 38).

[Rub92] D. H. Rubine. "The Automatic Recognition of Gestures." In: (1992). UMI Order No. GAX92-16029 (cit. on pp. 17, 19, 26).

[SKN07] B. Signer, U. Kurmann, M. Norrie. "iGesture: A General Gesture Recognition Framework." In: *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02*. ICDAR '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 954–958. ISBN: 0-7695-2822-8 (cit. on p. 25).

[Swi05] S. Swigart. *Easily Write Custom Gesture Recognizers for Your Tablet PC Applications*. https://msdn.microsoft.com/en-us/library/aa480673.aspx. Accessed: 2016-04-11. 2005 (cit. on p. 27).

[TRZ15] H. Tu, X. Ren, S. Zhai. "Differences and Similarities between Finger and Pen Stroke Gestures on Stationary and Mobile devices." In: *ACM Transactions on Computer-Human Interaction* 22.5 (2015), pp. 1–39. ISSN: 10730516. DOI: 10.1145/2797138 (cit. on pp. 20, 40, 41, 63, 64).

[VAW12] R.-D. Vatavu, L. Anthony, J. O. Wobbrock. "Gestures As Point Clouds: A $P Recognizer for User Interface Prototypes." In: *Proceedings of the 14th ACM International Conference on Multimodal Interaction*. ICMI '12. Santa Monica, California, USA: ACM, 2012, pp. 273–280. ISBN: 978-1-4503-1467-1. DOI: 10.1145/2388676.2388732 (cit. on p. 27).

[VAW13] R.-D. Vatavu, L. Anthony, J. O. Wobbrock. "Relative Accuracy Measures for Stroke Gestures." In: *Proceedings of the 15th ACM on International Conference on Multimodal Interaction*. ICMI '13. Sydney, Australia: ACM, 2013, pp. 279–286. ISBN: 978-1-4503-2129-7. DOI: 10.1145/2522848.2522875 (cit. on pp. 18, 25, 38, 39).

[VAW14] R.-D. Vatavu, L. Anthony, J. O. Wobbrock. "Gesture Heatmaps: Understanding Gesture Performance with Colorful Visualizations." In: *Proceedings of the 16th International Conference on Multimodal Interaction*. ICMI '14. Istanbul, Turkey: ACM, 2014, pp. 172–179. ISBN: 978-1-4503-2885-2. DOI: 10.1145/2663204.2663256 (cit. on p. 25).

[VW15] R.-d. Vatavu, J. O. Wobbrock. “Formalizing Agreement Analysis for Elicitation Studies.” In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI ’15* (2015), pp. 1325–1334. DOI: 10.1145/2702123.2702223 (cit. on pp. 24, 25).

[Vat+11] R. D. Vatavu, D. Vogel, G. Casiez, L. Grisoni. “Estimating the perceived difficulty of pen gestures.” In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 6947 LNCS.SEPTEMBER (2011), pp. 89–106. ISSN: 03029743. DOI: 10.1007/978-3-642-23771-3{_}9 (cit. on pp. 19, 22, 64, 68).

[WMW09] J. O. Wobbrock, M. R. Morris, A. D. Wilson. “User-defined gestures for surface computing.” In: *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09* (2009), p. 1083. ISSN: 1605582468. DOI: 10.1145/1518701.1518866 (cit. on p. 24).

[WWL07] J. O. Wobbrock, A. D. Wilson, Y. Li. “Gestures without libraries, toolkits or training: a 1 recognizer for user interface prototypes.” In: *Proceedings of the 20th annual ACM symposium on User interface software and technology UIST 07* 85.2 (2007), p. 159. ISSN: 01689452. DOI: 10.1145/1294211.1294238 (cit. on pp. 26, 38, 68).

[Wob+05] J. O. Wobbrock, H. H. Aung, B. Rothrock, B. A. Myers. “Maximizing the Guessability of Symbolic Input.” In: *CHI ’05 Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’05. Portland, OR, USA: ACM, 2005, pp. 1869–1872. ISBN: 1-59593-002-7. DOI: 10.1145/1056808.1057043 (cit. on pp. 24, 82).

[ZK03] S. Zhai, P.-O. Kristensson. “Shorthand Writing on Stylus Keyboard.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’03. Ft. Lauderdale, Florida, USA: ACM, 2003, pp. 97–104. ISBN: 1-58113-630-7. DOI: 10.1145/642611.642630 (cit. on p. 20).

[ZK12] B. S. Zhai, P. O. Kristensson. “Foundational Issues in Touch-Surface Stroke Gesture Design — An Integrative Review and Xiang Cao.” In: 5.2 (2012), pp. 97–205. DOI: 10.1561/1100000012 (cit. on pp. 17, 22–24, 64, 82).

All links were last followed on 31 May, 2016.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature