UNIVERSITÉ DE LILLE 1

ÉCOLE DOCTORALE SPI

Sciences Pour l'Ingénieur

THÈSE

pour obtenir le titre de

Docteur en Sciences

de l'Université Lille 1

Spécialité : INFORMATIQUE

Soutenue par

Yosra REKIK

Comprendre, Modéliser et Concevoir l'Interaction Gestuelle Tactile

préparée au centre Inria Lille Nord-Europe
au sein de l'équipe MINT

soutenue le – 10 Décembre 2014

Numéro d'ordre : 41575 — Année : 2014

Jury :

Président :	Gilles GRIMAUD	-	Professeur, Université Lille 1
Rapporteurs :	Joaquim JORGE	-	Professeur, Université Technique de Lisbonne
	Laurence NIGAY	-	Professeur, Université Joseph Fourier Grenoble 1
Examinateurs :	Martin HACHET	-	Chargé de recherche - HDR, Inria Bordeaux
	Eric LECOLINET	-	Maître de Conférences, Télécom ParisTech
Directeurs :	Laurent GRISONI	-	Professeur, Université Lille 1
	Nicolas ROUSSEL	-	Directeur de Recherche, Inria Lille Nord-Europe
Invité :	Radu-Daniel VATAVU	-	Professeur, Université Stefan cel Mare de Suceava, Romanie

UNIVERSITY OF LILLE 1
DOCTORAL SCHOOL SPI
Sciences Pour l'Ingénieur

PHD THESIS

to obtain the title of

PhD of Science

of the University of LILLE 1
Specialty : COMPUTER SCIENCE

Defended by

Yosra REKIK

Understanding, Modeling and Designing Multi-Touch Gestures Interaction

prepared at INRIA Lille Nord-Europe,
MINT Team

defended on December 10, 2014

Order Number : 41575 — Year : 2014

Jury :

President :	Gilles GRIMAUD	-	Professor, University of Lille1
Reviewers :	Joaquim JORGE	-	Professor, Technical University of Lisbon
	Laurence NIGAY	-	Professor, University Joseph Fourier Grenoble 1
Examinators :	Martin HACHET	-	Research Scientist - HDR, Inria Bordeaux
	Eric LECOLINET	-	Associate Professor, Telecom ParisTech
Advisors :	Laurent GRISONI	-	Professor, University of Lille1
	Nicolas ROUSSEL	-	Senior Researcher, Inria Lille Nord-Europe
Invited :	Radu-Daniel VATAVU	-	Professor, University Stefan cel Mare of Suceava, Romania

Contents

List of Figures

List of Tables

Abstract

This thesis presents our research investigations on multi-touch gesture variability. We first study multi-touch gesture variability from a user perspective, and then we investigate applicable tools and techniques for multi-touch interaction.

Towards understanding multi-touch variability, we set-up a pair of user studies. From the first study, we outline a taxonomy of multi-touch gestures in which we present the different aspects of a single unified dynamic mechanism ruling users in the achievement of a multi-touch gesture. In particular, we introduce the concept of atomic movement that reflects users' perception of fingers movement during gesture articulations. From the second study, we provide a more comprehensive analysis on multi-touch gesture variability. We differentiate between the major and minor sources of variation during multi-touch gesture articulation and outline eight representative gesture classes. We analysis the link between gesture shape and gesture articulations.

Moreover, we address the question of whether these different sources of variations induce different degree of articulation difficulty or if they are equivalent from a user-centric perspective. We there-by conduct the first investigation on the user-perceived difficulty of multi-touch gesture articulations. We report correlation results between users' subjective assessments of difficulty and objective gesture descriptors to enable a better understanding of the mechanisms involved in the perception of articulation difficulty. Through an in-depth analysis, we reveal new findings about how people synchronize their fingers and hands during gesture articulation by studying gesture structure, geometry and kinematics descriptors. We use our large body of results and observations to compile a set of guidelines for multi-touch gesture design by considering the ergonomics of multi-touch input through the prism of the user-perceived difficulty of gesture articulation.

After studying multi-touch gestures from a purely user-centric perspective, we provide tools and techniques that can be integrated in multi-touch interaction systems. We first propose a new preprocessing step, Match-Up, specific to multi-touch gestures (a first in the gesture literature) that structures finger movements consistently into clusters of similar strokes, which we add to the practitioners' toolkit of gesture processing techniques. We then apply Match-Up to recognize multi-touch input under unconstrained articulation (Match-Up & Conquer), for which we show an improvement in recognition accuracy of up to 10%. Finally, we introduce the concept of *rigid movement* and investigate its potential usage in order to strength interaction and offer users more flexibility in articulating gestures. In particular, we show how it can enable to free users from the use of a prefixed number of fingers, as well as from a predefined trace, when articulating a gesture.

Keywords: Multi-touch gestures, user studies, gesture taxonomy, gesture variability, gesture analysis, gesture structure, gesture geometry, gesture kinematics, gesture articulation difficulty, gesture recognition, structuring finger movements, gestural interaction techniques.

Résumé

Cette thèse présente nos travaux de recherche sur la variabilité du geste tactile multi-doigts. Nous étudions d'abord la variabilité du geste multi-doigts du point de vue utilisateur, ensuite nous décrivons un ensemble d'outils et de techniques d'interaction multi-doigts.

Dans le but de comprendre la variabilité du geste tactile multi-doigts, nous avons mis en place deux études utilisateur. À partir de la première étude, nous présentons une taxonomie des gestes multi-doigts dans laquelle nous présentons de façon unifiée les différents aspects qui amènent les utilisateurs à la réalisation d'un geste en particulier. Dans ce contexte, nous introduisons le concept de mouvement atomique permettant de refléter comment l'utilisateur perçoit les déplacements de ses doigts durant la production d'un geste. De la seconde étude, nous proposons une analyse fine et approfondie de la variabilité des gestes multi-doigts. Nous distinguons les sources majeures et mineures de variations et présentons huit classes représentatives de la variabilité des utilisateurs. Nous analysons notamment le lien entre la forme du geste et la façon avec laquelle il est produit.

Nous abordons, ensuite, la question de savoir si les différentes sources de variations induisent différents degrés de difficulté ou si elles sont équivalentes pour l'utilisateur. Pour celà, nous décrivons une étude nouvelle sur la perception de la difficulté des gestes multi-doigts. Nous présentons les résultats de corrélation entre l'évaluation subjectives des utilisateurs et les descripteurs du geste, permettant ainsi une meilleure compréhension des mécanismes impliqués dans la perception de la difficulté chez les utilisateurs. Grâce à une analyse approfondie portant sur la structure, la géométrie et la cinématique du geste multi-doigts, nous donnons des résultats sur la façon dont les gens synchronisent leurs doigts et mains pendant la production d'un geste. Nous utilisons, ensuite, notre vaste ensemble de résultats et d'observations pour définir un ensemble de lignes directrices en adoptant le prisme de la perception de la difficulté perçue par les utilisateurs comme un facteur important des gestes multi-doigts lors de la conception de geste multi-doigts.

Après avoir étudié les gestes multi-doigts d'un point de vue utilisateur, nous fournissons des outils et des techniques prenant en compte la variabilité de l'utilisateur et qui peuvent être intégrés dans des systèmes d'interaction tactile. Nous proposons d'abord une nouvelle étape de prétraitement, Match-Up, spécifique à des gestes multi-doigts qui structure (pour la première fois dans la littérature) les mouvements des doigts. Nous appliquons ensuite Match-Up dans le cadre de la reconnaissance (Match-Up & Conquer), et nous obtenons une amélioration des taux de reconnaissance de 10%. Enfin, nous introduisons le concept du mouvements rigides et nous étudions son potentiel pour rendre l'interaction plus flexible. En particulier, nous montrons comment il peut libérer l'interaction d'utiliser un nombre prédéterminé de doigts, ainsi que d'une trace pré-établie.

Mots-clés: geste tactile multi-doigts, études utilisateur, taxonomie de geste, variabilité du geste, analyse du geste, structure du geste, géométrie du geste, cinématique du geste, difficulté de production d'un geste, reconnaissance du geste, structuration de mouvement de doigts, interaction tactile gestuelle.

Acknowledgments

I would like to first thank my advisors, Laurent Grisoni and Nicolas Roussel, for their faithful support and sharp insight. It has been a real pleasure working with them.

I would like to thank my reviewers Prof. Joaquim Jorge and Prof. Laurence Nigay for their time and energy on reviewing my thesis. I also thanks Prof. Gilles Grimaud, Dr. Martin Hachet and Dr. Eric Lecolinet for agreeing to examine my thesis defense. A special thanks goes to Prof. Radu Daniel Vatavu, I enjoyed working with you. I would like to thank the whole MINT team who is such a wonderful group of people.

My gratitude goes to my family – Rekik and Derbel. Thank your for your encouragement and generosity. Though you are far away, I know that you have never stopped caring or believing in me. Especially, I thank my father Amor and my mother Souad, my brother Wassim and my two sisters Dorra and Naira, as well as, my parents in-law Moncef and Jamila, my brothers and sisters in-law Mohamed, Nizar, Houda, Aida, Myriam and Nizar.

Last, but not least, I would like to thank and express my utmost appreciation and love for the person who shares my life and my heart: my husband Bilel Derbel. This dissertation is dedicated to him. He have been my strength, my will and my rock during these four years which seemed like the longest years of my life. Thank you, Bilel, for being always there for me, supporting my stress, finding the right words to make me laugh every day. Thank you for all.

"The important thing about gestures is that there are not fixed. They are free and reveal the idiosyncratic imagery of thought."

MCNeil David – Professor at the Department of Psychology, University of Chicago (1992)

1 Introduction

Gestures are "a separate symbolic vehicle with their own history and finding their own outlet in space, movement, and form" [McNeill 1992]. In this context, how and where the gesture is articulated is important and can highly attend humans in a meaningful and specific dialogue. For instance, when drawing a symbol using a pen and a paper, different movements done by hands to control pen's trace as well as the way the pen is maintained by the hand provide a subtile different stroke. When using paintbrush and canvas, the resulted strokes can be controlled in a subtile manner, and specific hand movements can also be more expressive than when using a simple pen. When observing an artist drawing, we can see that the movements of his hands are crucial to adequately express his feelings when drawing a desired stroke. When being at the beach, one can draw directly on the sand to write his name or draw some forms. In this case, we can use a rod as well as our hands, fingers, palms as input and the sand as output. We can observe in these examples different levels of movements in the drawing task which contributes in the construction of different forms of meanings and dialogues. With technology we can observe a similar phenomena: the role of hands' movements increases when passing from a mouse and a keyboard; to a pen/finger with tablets, and to the use of multi-touch freehand gestures on multi-touch surfaces. This enables to enrich the human computer dialogue and it also poses several research questions and challenges. In this dissertation, we are interested in gestural interaction and more specifically on studying multi-touch gestures.

Generally speaking, people exhibit inherent intrinsic variations in their gesture articulations because gestures carry dependency with both the person producing them and the specific context, social or cultural, in which they are being produced. Indeed, in his psycholinguistic study on human discourse and relationship between gesture and thought, McNeill [McNeill 1992] considers that "gestures are the spontaneous creations of individual speakers, unique and personal" and that gestures "reveal the idiosyncratic imagery of thought" (p. 1). The user-dependency aspect of gesture production has been many times reflected by previous work that analyzed users' gesture preferences in conjunction with specific acquisition technology, such as interactive tabletop surfaces [Morris 2010, Wobbrock 2009], accelerated movements [Ruiz 2011], and freehand gestures [Morris 2012, Vatavu 2013e, Vatavu 2012b]. These studies, generally referred to as “gesture elicitation studies”, have shown that some level of consensus exists among users due to similar conceptual models that users naturally seem to construct when thinking about common interactive tasks. Nevertheless, these studies also pointed out many variations in users’ preferences for gesture commands, with probably the most important finding being that users were found to prefer different gesture commands than those proposed by experienced designers [Morris 2010]. In this dissertation, we present the results of our investigations on users’ gestures and their variations for multi-touch surfaces. Our work includes user-centric studies, gesture elicitation and classification, statistical gesture analysis, qualitative and quantitative data analysis, gesture design guidelines, as well as, the design of new tools for multi-touch recognition and interaction techniques.

1.1 Characterizing the Challenges

Multi-touch input offers many degrees of freedom that can be independently controlled during gesture articulation, such as the number of fingers or finger type [Bailly 2012], single-handed or bimanual input [Kin 2009, Wu 2003], variations in the number of strokes forming the gesture [Anthony 2013b], and the use of additional modalities accompanying finger touch input leveraged by sensing pressure [Hennecke 2011] and various parts of fingers anatomy [Harrison 2011]. In addition, multi-touch interfaces become increasingly popular, with applications from personal smart phones and tablet devices [Ruiz 2011] to interactive displays installed in public settings [Hinrichs 2011] and, lately, applications that span between these two interactive spaces [Kray 2010]. With the versatility of users’ gestures and the many degree of freedom offered by multi-touch surfaces, a broad range of gesture-based interaction styles have been designed for these surfaces, which has incontestably contributed toward the success and rise of the underlying technology.

Gestures, or commands issued with touches, are one desirable feature of multi-touch applications. However, users in such systems tend to use different class of gestures, which implies that any application shall potentially integrate ways to handle such variety. Some findings indicate that users’ choice of gestures is influenced by the context in which the current action occurred and not only based on preferences for a given gesture for a particular action (*e.g.*, [Hinrichs 2011]). This suggests that a many-to-one mapping is desirable to strengthen the design of gestural interaction techniques. In the meantime, multi-touch gestures are often thought by application designers for a one-to-one mapping between gestures and commands, which does not necessarily take into account the high variability of users’ gestures – this might also be a design choice that can lead to simplistic interaction choices. For example, in their user-defined gesture study, Wobbrock et al. [Wobbrock 2009] captured extremely well the many degrees of freedom offered by multi-touch gestures and noted that “surface gestures are versatile and highly varied – almost anything one can do with one’s hands could be a potential gesture" (p. 1083). The focus of such a study and many others is on the design and analysis of a one-to-one mapping between gestures and their actions, which inherently leads to the discrimination of some gestures that could possibly fit in a given interaction context. We therefore argue that it is timely to understand how users handle variability in multi-touch gestures under unconstrained conditions.

At the same time, taking into account the variability of users makes prototyping multi-touch recognizers a difficult task because, in many cases, "the programming of these [multi-touch] gestures remains an art" [Lü 2012]. We actually believe that one cause for this recognition challenge lies in our limited understanding of the variability of multi-touch gesture articulation. This issue can affect not only recognition performance, but also users' expression possibilities in current multi-touch interfaces. A better understanding of users' variability could benefit interface design beyond achieving highly-accurate recognition, toward more fluent and expressive interaction which would be able to exploit the explicit signals contained within the variability of the recognized articulation [Caramiaux 2013].

Besides recognition issues when handling gesture variability, eliciting the level of difficulty in possible gesture articulations can highly contribute to the success of multi-touch gesture design. While it is not straightforward to precisely define the notion of gesture difficulty, we argue that it captures many facets of gesture production, such as the ergonomic difficulty to physically articulate the gesture path and the cognitive difficulty required to learn and recall the geometry of the gesture shape and, conceivably, the specifics of its articulation. Thus understanding the effect of each source of variation offered by multi-touch surface in gesture articulations is a crucial issue towards the design of multi-touch gestures.

The previously discussed issues constitute the main research topics that are addressed in this thesis. We can summarize them in the following five points:

1. Understanding how users handle variability in multi-touch gestures under unconstrained articulation conditions
2. Identifying the primary sources of variations for multi-touch input and characterizing gesture articulations falling into the same class of variation
3. Understanding the difficulty of each source of variation on the gesture articulation
4. Designing recognition techniques that handle user variability in a robust and consistent manner
5. Providing a set of tools and guidelines that help the design of more powerful interaction techniques.

1.2 Thesis Statement

The thesis statement of this dissertation is as follows:

> *I argue that analyzing how users handle variability in multi-touch gestures under unconstrained articulation conditions and without regard for recognition or technical concerns is essential for the understanding of multi-touch gestures, the design of successful multi-touch gesture sets, and the strengthening of interaction techniques. To this end, I propose a set of user studies to understand user variability and study its impact on produced gestures, and to characterize users' perceived difficulty of multi-touch gesture articulations. I then propose Match-Up & Conquer, a two step technique for structuring and recognizing unconstrained bimanual multi-touch gestures. I finally propose the concept of rigid-movement and explore how it can allow to design new flexible interaction techniques based on users' freehand movements.*

1.3 Thesis Organization and Contributions Overview

This thesis is organized in two tightly related parts. In the first part, we are mainly concerned with understanding multi-touch gestures and their variations from a pure user-centric perspective. In the second part, we are interested in designing new tools and technique that can deal with this variability to enhance interactions. These two parts and their corresponding chapters are sketched in the following paragraphs. Let us note that in order to allow the reader to better appreciate our different contributions, we shall first provide in **Chapter 2 – "Background and Contribution Positioning"**, a literature overview where we address in a brief but systematic manner the different existing studies that relate directly or indirectly to the work presented all along this dissertation. In particular, we shall give an overview of gesture-based interaction techniques, gesture elicitation studies and related taxonomies, gesture analysis studies, and gesture recognition issues. **Chapter 8 – "Conclusion and Future Work"** concludes our dissertation by resuming the contributions and outlining some areas for future research investigations.

Part I – "Understanding Users' Unconstrained Multi-touch Gestures"

Chapter 3 – "A Multi-level Taxonomy of Users' Multi-touch Gesture Variability" We present the first comprehensive investigation of understanding how users handle variability in multi-touch gestures under unconstrained articulation conditions through a user study. This study allows us to provide a qualitative analysis of user gesture variability and to derive a taxonomy for users' gestures that complements other existing taxonomies. We also study variability within a fine grain quantitative analysis. We finally, discuss implications of our study with respect to interaction design, gesture recognition, and potential gesture-based applications.

Publications . . .

The results of this chapter were published in INTERACT'13 [Rekik 2013]: *Yosra Rekik, Laurent Grisoni and Nicolas Roussel. Towards Many Gestures to One Command: A User Study for Tabletops. In Proceedings of the 14th IFIP TC13 Conference on Human-Computer Interaction, pages 246–263. September 2013, Cap Town, South Africa. 18 pages (Springer).*

Chapter 4 – "On the Proprieties of Multi-touch Gesture Variations" We present a new experiment on user-gesture variability in which we identify the primary sources of variation during multi-touch gesture articulation. We propose a taxonomy of eight distinct super-classes for multi-touch gestures. We attempt to identify the relation between gesture type and gesture articulation variation. We also characterize articulations with geometric and kinematic descriptors and, by employing frequency distribution analysis, we empirically show that the elicited classes have distinct characteristics. For some of the elicited variations, we go into a throughout comprehensive analysis and reveal new findings about how people synchronize their fingers and hands during gesture articulation. We finally accompanied our quantitative results with qualitative data that capture users' mental models as they choose and articulate a gesture articulation.

Publications ...

The results of this chapter include a database of multi-touch gestures composed of 5.155 samples of 22 gesture types collected from 16 participants[a] which is made freely available for the community. We named this dataset Creativity Dataset. Besides, an extended version of this chapter is submitted to a journal.

[a]https://sites.google.com/site/yosrarekikresearch/projects/matchup

Chapter 5 – "On the Perceived Difficulty Of Multi-touch Gestures" We present the first investigation of the user-perceived articulation difficulty of multi-touch gestures, measured as gesture *difficulty ratings* and *rankings*, by examining the effect of finger count, stroke count, and single and bimanual articulation conditions. Then we report correlation results between subjectively-perceived articulation difficulty and objectively-computed gesture descriptors, *e.g.*, path length and production time correlate highest with perceived difficulty in almost all conditions. We introduce a new variant of computing multi-touch gesture descriptors by considering the cumulative effect of all employed fingers, in the form of *actual* path length, *actual* number of strokes, and *actual* total turning angles, which extend the characterization range of multi-touch input. We report novel findings on multi-touch gestures articulated under different conditions enabled by the use of structure, geometric and kinematic descriptors, *e.g.* bimanual articulations result in gesture shapes that are horizontally stretched. We present a set of guidelines for multi-touch gesture set design that correlate multi-touch ergonomics, user-perceived difficulty, multi-touch recognizers and potential gesture to function mappings.

Publications ...

The results of this chapter were published in ICMI'14 [Rekik 2014b]: *Yosra Rekik, Radu-Daniel Vatavu and Laurent Grisoni. Understanding Users' Perceived Difficulty of Multi-Touch Gesture Articulation. In Proceedings of the 16th International Conference on Advanced Visual Interfaces, to appear, November 2014, Istanbul, Turkey. 8 pages (ACM).* Besides, our set of 7.200 samples of 30 gesture types collected from 18 participants annotated with RATING and RANKING data.[a] is made freely available for the community. We named this dataset Supervised Dataset.

[a]https://sites.google.com/site/yosrarekikresearch/projects/gesturedifficulty

Part II – "Tools and Techniques for Unconstrained Multi-touch Interaction":

Chapter 6 – "Match-Up & Conquer: Structuring and Recognizing Multi-touch Input" We present a new preprocessing step, Match-Up, specific to multi-touch gestures (a first in the gesture literature) that structures finger movements consistently into clusters of similar strokes, which is added to the practitioners' toolkit of gesture processing techniques, next to scale normalization, resampling, and rotation to indicative angle [Anthony 2010, Anthony 2012b, Li 2010, Vatavu 2012c, Wobbrock 2007]. We present an application of Match-Up to recognize multi-touch input under unconstrained articulation (Match-Up & Conquer), for which we show an improvement in recognition accuracy of up to 10% over an existing technique. We present pseudocode for assisting practitioners in implementing Match-Up into their gesture interface prototypes.

Publications ...

The results of this chapter were published in AVI'14 [Rekik 2014a]:
Yosra Rekik, Radu-Daniel Vatavu and Laurent Grisoni. Match-up & Conquer: A Two-step Technique for Recognizing Unconstrained Bimanual and Multi-finger Touch Input. In Proceedings of the 12th International Conference on Advanced Visual Interfaces, pages 201–208, May 2014, Como, Italy. 8 pages (ACM). Besides, we make available the Match-up & Conquer c++ library[a] distributed under the LGPL version2 license agreement.

[a] https://sites.google.com/site/yosrarekikresearch/projects/matchup

Chapter 7 – "Rigid Movement Based Interaction" We propose the concept of *rigid movement* to increase the flexibility of gestural interaction and to enlarge the panel of possible unconstrained gestures. We show how these movements can be determined in real time and extracted suing solely the contact information usually provided by multi-touch surface without any additional sensing or technological facilities. We explain how rigid-movements can contribute to free the interaction from the number of fingers when designing a gesture and to move towards multi-movement gestures, independent or coordinated. Finally, we prototype some interaction techniques that takes advantage of this increased flexibility and discuss their potential benefits.

Publications ...

The results of this chapter were partially published in IHM'12 [Rekik 2012]:
Yosra Rekik, Nicolas Roussel and Laurent Grisoni. Mouvements Pseudo-rigides Pour Des Interactions Multi-doigts Plus Flexibles. In Proceedings of the 24th Conference on Ergonomie Et Interaction Homme-machine, Ergo'IHM'12, pages 241–244, October 2012, Biaritz, France. 4 pages (ACM).

"We're in the middle of a period that I refer to as a period of "combinatorial innovation." So if you look historically, you'll find periods in history where there would be the availability of a different component parts that innovators could combine or recombine to create new inventions."

Hal Varian, Chief Economist at Google (2009)

2 Background and Contribution Positioning

Since the beginning, multi-touch systems have been conceived [Mehta Nimish 1982] to free interaction from the constraint of a single point of input and to allow users to apply any number of touches. Multi-touch interaction is by essence based on multi-touch gestures. In this respect, gestures play a crucial role in the design of multi-touch applications, and highly contribute in making interaction accessible and reachable to an increasingly large audience of people. Accordingly, the fact that the human body (fingers, hands, palm, etc) actively participate in the interaction, constitutes one important factor that contributes in the success and development interactive surfaces. Hence, there is evidence to understand what can be gestures and to better grasp their properties and their specificities.

From a very general human perspective, gestures are deeply incorporated in our everyday life as they constitute something that convey information and allows us to communicate, to express a meaning, a feeling, an intention, etc. Generally speaking, a gesture is revealed by the physical movements or postures of whole or parts of one's body including fingers, hands, arms, face, etc. The vocabulary of gestures that people use is not unique but versatile. In his psycholinguistic study on human discourse and relationship between gesture and thought, McNeill considers that "gestures are the spontaneous creations of individual speakers, unique and personal" and that gestures "reveal the idiosyncratic imagery of thought" [McNeill 1992, p. 1]. In an attempt to organize the large spectrum of human gestures, several taxonomies and classification criteria have been proposed by psycholinguistic and sociolinguists. Efron [Efron 1941] is from the first researchers that conducted a methodological study for describing human gestures. He classifies gestures into two broad categories depending on whether the gesture had meaning independent of speech or in conjunction with speech. This results in five categories on which later taxonomies were built. These categories were: *physiographics*, *kinetographics*, *ideographics*, *deictics*, and *batons*. The first two are considered as *iconics* in the work of McNeill [McNeill 1992] while the last three are also identified. The classifications proposed by Efron and McNeil were mainly based on human discourse which make them of limited applicability to surface gestures. In the same spirit, Adam Kendon [Kendon 1988] distinguished 5 different kinds of gestures along a continuum going from 'gesticulation' up to 'signs'. Kendon's classification of human gestures is as follows. 'Gesticulation' refers to "idiosyncratic spontaneous movements of the hands and arms accompanying speech". 'Speech framed gestures' represent a gesticulation that is incorporated into speech to complete the sentence structure. 'Emblems' refer to conventionalized signs that may or not be accompanied by speech (*e.g.*, thumbs up for "OK"). 'Pantomime' gestures are produced without speech and have a narrative function by telling a story. 'Sign' gestures represent lexical words in a sign language and have their own linguistic structures, including grammatical patterns, stores of words, morphological patterns, etc. When moving along Kendon's gesture continuum, we notice that the role of speech in accompanying gestures decreases, and the relative intrinsic role of gestures increases. This propriety makes Kendon's Continuum interesting to surface gestures even though it is not readily applicable for this type of surface.

Actually, the limited applicability of the previous classifications and taxonomies in the context of surface gestures stems from one major point which is the input characteristics of interactive surfaces and particularly multi-touch devices. In fact, multi-touch input offers many degrees of freedom that can be identified during gesture articulation. Basically, multi-touch devices support gestures defined as a function of the touch positions provided by the system. Some multi-touch devices are able to detect other attributes such as finger pressure, touch shape, etc. Besides, as multi-touch devices are able to detect multiple fingers in the same time, they are good candidate to support bimanual gestures. This, in addition to the different properties, features and possibilities offered by the newly emerging multi-touch devices, have inspired researchers to create a large panel of multi-touch gestures. Besides, in order to enhance the so-designed gestures and to strengthen interaction, many research axis are being actively investigated by the community.

In the rest of this chapter, we first review some gesture-based interaction techniques specific to interactive surfaces and emphasizes the multi-touch related aspects. This shall allow us to give an overview of different issues that are related to the work presented in this thesis. We there-by provide a discussion of different paradigms being specifically investigated with respect to multi-touch and gesture-based interaction. For the sake of coherence, and in order to make the contributions of this thesis clear with respect to existing work, we shall highlight our contributions in a regular fashion using styled boxes.

2.1 An Overview of Gesture-Based Techniques

In this section, we survey a representative set of existing gestures for basic tasks such as object manipulation, as well as stroke-based gestures, touch-attribute based gestures and bimanual gestures. Our goal is to highlight different interaction with their underlying multi-touch gestures and to give a brief overview on their usage.

2.1.1 Gestures for Object Manipulation

A rich set of gestures and a specific gesture vocabulary have been defined by designers to allow users to interact with multi-touch surfaces. Object manipulation and spatial grouping are among the most common tasks that motivated the design of a variety of gestures. In the following, we give a brief overview of such gestures.

Gestures based on translation, rotation and scaling. The most standard multi-touch gestures appear in the context of simple object transformation, namely, rotation, translation and scaling. Hancock et al. [Hancock 2006] surveyed five different mechanisms implying different gestures to manipulate objects displayed on a direct-touch digital tabletop. In the first mechanism, an explicit specification of the position and orientation are defined. For example, Wu and Balakrishnan [Wu 2003] defined a rotation gesture as a two-point gesture made with the thumb and the index fingers, with which a user can control the object orientation by manipulating the rotation angle. In the second mechanism, the user is provided with specific areas within which he can control the rotation or the translation in an independent manner [Shen 2004, Streitz 1999, Tandler 2001]. The third mechanism consists in an automatic specification where translation and rotation are combined in one motion [Forlines 2005, Shen 2004, Vernier 2002]. In the fourth mechanism, rotation and translation are used through the integration of a physically-based model [Beaudouin-Lafon 2001, Kruger 2005, Mitchell 2003]. Finally, the last mechanism is characterized by a two-point rotation and translation where the translation is calculated as the distance between the first contact point T_1 and its new position $T_1^{'}$ and the rotation represents the angle between T_1, $T_1^{'}$ and $T_2^{'}$ (the new position of the second touch point T_2). In this case, a scale factor can be further calculated as $T_1^{'}T_2^{'}/T_1T_2$ [Fitzmaurice 1999].

Nacenta et al., [Nacenta 2009] investigated four multi-finger manipulation techniques using translation, rotation and scaling in order to study their separability and to outline the best subset of manipulation techniques: Handles, Magnitude Filtering, and Gesture Matching (Frame-to-Frame and First-Touch). In the Handles technique, one handle is provided in order to enable a user to directly select the desired manipulation (by picking the corresponding handle). In the Magnitude Filtering technique, every transformation (rotation, scale, translation) induced by a multi-touch gesture is filtered. Then, the performed rotation, scale and translation will be effective only if its value is bigger than a fixed threshold. In the Gesture Matching technique, the simplest subset of transformations that explain reasonably the actual motion implied by the gesture is selected and applied.

Gestures for cooperative work. Multi-touch gestures have also been defined in several collaborative techniques. Rekimoto [Rekimoto 2002] created "SmartSkin" an infrastructure for freehand manipulation on surfaces where multiple users can interact in the same time. A set of physical gestures has been defined including panning, scaling, rotating and lifting objects. Wu and Balakrishnan [Wu 2003] proposed RoomPlanner, a prototype room furniture layout application. RoomPlanner allow collaborative interaction and support gestures for rotation, menu access, object collection, and private viewing. Shen et al., [Shen 2004] proposed "DiamonSpin" a toolkit for supporting multi-user multi-touch interaction. Multi-user interactions are allowed through multi-thread event streams, multiple active objects and multiple concurrent menus. Morris et al., [Morris 2006] introduced cooperative gesturing, a multi-user interaction technique for co-located single display groupware systems. Wu et al. [Wu 2006] defined three design principles (gesture registration, relaxation and reuse) to design a gesture vocabulary for adding and erasing annotations, cutting, copying and pasting documents and piling and browsing items.

Gesture for spatial grouping and selection. Examples of multi-touch gestures are also available for dealing with multiple objects. For instance, Wu et al. [Wu 2006] proposed "Pile-n-Browse" gesture to allow pilling and browsing a group of items. The gesture consists in placing two hands on the surface, so that a filled circle can appear and show which items will be selected. The set of items in the so-defined circle can be selected and the user can either activate their piling by scooping both hands in or browse them through by moving hands apart. Tse et al. [Tse 2008] explored the design of multi-touch multi-user multi-modal group selection techniques. Several techniques have been then designed and tested including: (1) hand-bracketing (similar to [Wu 2006]), (2) lasso grouping meaning using a single finger to draw a lasso around the desired object, (3) searching notes by speech and (4) using multi-modal selection and grouping by first selecting one or more objects by speech and then say "group selected". Selected groups can be further manipulated. For example, a selected group can be moved around by using a five-fingered grabbing gesture and rearranged by saying "Arrange group" or sorted in alphabetical order by saying "sort alphabetically". Groups can be also panned using a 5-finger gesture or zoomed with a 2-finger gesture in an empty area. Wilson et al. [Wilson 2008] brought physics into multi-touch interaction. They proposed to select a group of objects by placing multiple fingers on desired objects or by 'pushing' with full hand shapes to form a pile. Cao et al., [Cao 2008a] explored how virtual contact forces from contact regions and motion can be inferred in multi-touch interaction to enable interaction similar to users' everyday experience when interacting with real objects. Users can use both hands to quickly sweep and collect multiple objects and scatter them into a pile. The pile of objects can then be further acted upon through different virtual force frequency, *e.g.*, a large virtual force can be used to drag a whole stack of objects. North et al. [North 2009] proposed to select a group of objects by using different strategies: while individual object are selected and moved by using one or two fingers, a group of objects can be selected by using three or more fingers. In this case, the convex hull formed by multiple fingers touching the surface represents a selection area. Höchtl et al., [Höchtl 2012] proposed two spatial grouping techniques on interactive surfaces: (1)"Bin Technique" allows to group items by moving them into a container object. (2) The "Blub Technique" enable to group items by moving them in close proximity to each other. Almeida Madeira Clemente et al., [de Almeida Madeira Clemente 2014] proposed the "Scoop Net" technique to dynamically select and move a group of graphical objects. One or multiple objects are selected with multiple fingers and moved to the same location. The touch points are then connected and form a virtual net if they are closed to each other. The net persists and can be winded to collect additional objects or narrowed to skip objects that are not desired.

2.1.2 Stroke Based Gestures

Stroke based gesture consider touch positions over time to define the pattern shape of a gesture. Researchers have developed stroke recognitions systems [Anthony 2010, Anthony 2012b, Vatavu 2012c, Wobbrock 2007] that deliver robust recognition accuracy in a short processing time. Thus, stroke based gestures have been widely adopted to interact with multi touch devices. For example, Kurtenbach [Kurtenbach 1993] allows users to select a menu item by drawing a directional stroke. Zaho and Balakrishnan [Zhao 2004] extended this technique by proposing the multi-stroke marking menus which allow users to efficiently traverse a hierarchy of submenus by drawing a sequence of stroke-gestures. Appert et al., [Appert 2009] used stroke gestures as shortcuts for touch screen-based devices. For example, by performing successively two short linear strokes, the users can define their choice within the menubar and the selected menu. The first stroke selects the menu and the second one selects the item in the menu. Actually, each stroke has a predefined orientation with eight possible choices. Kin et al., [Kin 2011] proposed two-handed multi-stroke marking menu variants in which users either draw strokes with both hands simultaneously or alternate strokes between hands. Roy et al. [Roy 2013] proposed Augmented Letters, a gesture shortcut that combines uni-stroke letters with marking menus. The goal is to simplify command memorization and to reduce the cognitive load by using the initials of command names as gesture shortcuts. To handle conflicts among commands that share the same initial, the stroke is augmented with a tail that can be oriented in up to eight directions. For mobile phone, Roudaut et al. [Roudaut 2009] proposed MicroRolls a gesture vocabulary that discriminates between thumb gestures for mobile devices by detecting specific patterns in the trajectory of thumb touch events.

2.1.3 Touch Attribute Based Gestures

One important propriety of multi-touch devices is that they are able to detect multiple touch points in the same time. Bailly et al., [Bailly 2012] have exploited this main propriety to propose finger-count technique in which they define gestures by simply counting the number of fingers of each hand in contact with the surface. Furthermore, researchers have investigated the benefits of multitouch input and mapped touch attributes to individual operations. Cao et al. [Cao 2008b] presented ShapeTouch, an exploration of interactions that leverages the contact shape to manipulate objects on the interactive surface similar to those in real world. Wilson et al. [Wilson 2008] used the contact contours to emulate physical reactions between touch input and digital objects. Davidson et al [Davidson 2008] demonstrated pressure-based depth sorting technique using a pressure sensitive surface which extends standard two-dimensional manipulation techniques. Benko et al [Benko 2006] use the contact size to simulate pressure input on the tabletop. Using stereo vision, Malik et al. [Malik 2004] determined user's finger orientation and the distance of a fingertip from the surface to manipulating images. Wang and Ren [Wang 2009b] used a Frustrated Total Internal Reflection (FTIR) based multi-touch surface to empirically investigate finger contact properties such as size, shape, width, length and orientation. Harrisson et al. [Harrison 2011] proposed TapSense, an enhancement to touch interaction that allows the surface to discriminate between the various parts of the fingers' anatomy including the tip, pad, nail and knuckle. Wang et al. [Wang 2009a] used the anatomy of the human hand in order to identify which touches belong to the same hand. Wagner et al. [Wagner 2014] proposed a posture vocabulary named multi-finger chords that describe a positioning of particular fingers on a screen relative to the hand and allow to access items in a menu.

2.1.4 Bimanual Gestures

Using two hands is common in the physical world. Multi-touch devices favor this interaction style by allowing users to make use of their both hands. Bimanual interaction can be asymmetric or symmetric. Asymmetric interaction has been introduced by Guiard in his seminal work on the Kinematic Chain Model [Guiard 1987]. Asymmetric interaction consists in attributing asymmetric roles to the hands. The dominant hand (here we consider that it is the right hand) and the non-dominant hand (left hand) cooperate and act together to reach a goal. Guiard defined three main principles that researchers should follow to design efficient two-handed asymmetric interaction: (1) "Right-to-Left Spatial Reference in Manual Motion" *i.e.*, "the motion of the right hand finds its spatial references in the results of motion of the left hand", (2) "Left-Right Contrast in the Spatial-Temporal Scale of Motion" *i.e.*, "on spatial axis, right hand realizes shorter and more accurate movements ... on temporal axis, right hand realizes movements with a greater frequency", and (3) "the left hand's contribution to current action starts earlier than that of the right hand" *i.e.*, "the left hand's contribution to current action starts earlier than that of the right hand". Based on these 3 principles, different bimanual asymmetric gestures and techniques have been designed [Bailly 2012, Malik 2004, Wu 2003, Buxton 1986, Balakrishnan 1999, Hinckley 1997, Kabbash 1994, Odell 2004]. On the other side, Symmetric interaction consists in assigning hands the same actions. For instance, Kin et al., [Kin 2011] presented two-handed symmetric multi-stroke marking menu. Symmetric roles are assigned to hands by drawing bilaterally symmetric strokes in the same time. For others systems (not necessary multi-touch based), symmetric techniques have been proposed including controlling two mouses simultaneously [Latulipe 2005], tracking a pair of targets [Balakrishnan 2000, Owen 2005] and editing rectangle and navigation [Casalta 1999].

> **In this thesis** ...
>
> *All along this dissertation, we also deal with a variety of multi-touch gestures in an attempt to better understand their nature and help designing interaction tools and techniques to handle them in a consistent and robust manner. At the end of this thesis, we also investigate the natural ability of users to move fingers on multi-touch surface in order to define new gestures. In fact, we contribute in Chapter 7 an approach for estimating the movements of multiple fingers independently of the way they are moving and propose the "Rigid movement" concept in an attempt to increase flexibility in designing multi-touch gestures. We use this concept to dynamically group fingers and to capture the global nature of the movement performed by users within a gesture. We then contribute a multi-touch technique for dynamic grouping and selection of objects.*

2.1.5 Discussion

With all the work described in the previous paragraphs, we can see that there exist a high diversity of interaction techniques with specific multitouch gestures. All gestures presented in this section have been thought empirically and designed by system designers and then-after presented to users for further evaluation. Despite skillful design, gestures can suffer some limitations that impact interaction. For instance, there is relatively little literature on what is the best mapping between the possible gestures to the desired commands or actions. This makes it difficult to choose multi-touch gestures accurately and to grasp the multitude of possibilities that one can consider. Fortunately, we can find a set of considerations that are believed to be important when designing multi-touch gesture based interaction. For example, gestures should be efficient to perform [Morris 2010, Wobbrock 2009, Nielsen 2004b], gestures should be not ambiguous [Bailly 2012], gestures should be coherent meaning that a set of gestures should "make sense together" [Wigdor 2011] and gestures should fit with their associated commands [Morris 2010, Wobbrock 2009, Nielsen 2004b]. Furthermore, several research axis have addressed specific issues in order to enhance multi-touch gestural interaction and to help the understanding, the design and the setting of multi-touch gestures. We sketch them in the following points:

1. Gestures discoverability, learning and memorization. This issue deals essentially with assisting users during gesture articulation by teaching them the set of gestures with sometimes an extra-learning step to guide novice users towards the expert mode or by guiding users with visual clues to perform gestures.

2. Gesture participatory design. This issue deals essentially with the process of including users till the design step in order to end up with user-defined gesture sets.

3. Gesture analysis. This issue deals essentially with proposing new predictive models to evaluate gesture complexity and difficulty or by analyzing gestures properties and outlining a set of principles and guidelines that can help designer when defining gestures.

4. Gesture recognition. This issue deals with the robust recognition of gestures despite the unavoidable errors, imprecision, freedom, etc, that users could have when articulating a gesture.

In the following, we give an overview of different work dealing with these issues. We note that the last three point are directly related to the different contributions presented in this thesis. The first point is not directly related to our contributions although it finds several connections with the other points. We basically include it in our literature overview since it is a recurrent issue when dealing with multi-touch gestures which might have some influences on our choices and developments.

2.2 Gestures Teaching and Memorization Methods

With the design of the different novel gestures, different teaching methods have been proposed for learning a gesture set to users. In the following, we first review traditional teaching methods namely cheat sheets. Then, we discuss systems that combine feedforward and feedback approaches. Finally, we briefly discuss active exploration of gestures and progressive learning.

2.2.1 Cheat Sheets

Cheat sheets provide users with a complete overview of a gesture set and its mapping to commands. Diagrams and pictograms [Brandl 2008, FingerWorks 2001, Elias 2007] have been used to make the learning easier. For instance, Brandl et al. [Brandl 2008] presented a prototype graphical editing application that includes a new method of teaching bimanual gestures. Elias et al. [Elias 2007] provided users with a dictionary of multi-touch gestures to facilitate leaning and memorization. Kurtenbach [Kurtenbach 1994] provided users with the animation of the articulation of a gesture with a text that describes the special features of the gesture. On Apple Mac OS X users are provided with short videos showing gesture articulation, the number of fingers to use and their mapping. Bragdon et al. [Bragdon 2010] proposed a novel approach for online learning of multi-touch gestures by using small games with physical simulations.

2.2.2 Feedforward and Feedback Systems

Several systems have been developed to help users discover, learn and articulate gestures by combining two concepts: Feedback and Feedforward. Feedback provides users with information about the current state of their gesture articulation. This feedback consists in the visualization of the system interpretation of user's current gesture. Feedforward provides users with information about the gesture total shape and proprieties (*e.g.*, hand pose) during the gesture articulation. For instance, the feedforward can provide users with the possible available completion paths necessary to finish the articulation of the gesture. With interaction systems adopting these concepts, novices users can safely explore the different gesture articulations and their mapping to commands. In the following we give a short overview of such systems.

For pen computers, Kristensson et al. [Kristensson 2007] presented "Command Strokes" a command entry technique. Command Strokes represent pen-gesture traces that define textual representations of commands on a graphical keyboard. To help users in discovering gestures, Command Strokes has been extended with a continuous recognition of gestures during articulation. This technique also provides users with a visual feedback that show the most likely commands that they are articulating. Bau and Mackay [Bau 2008] proposed "OctoPocus", a dynamic guide for helping users to learn, execute and remember single-stroke single touch gesture-based commands. OctoPocus provides novice users with continuous feedforward and feedback during gesture articulation. For instance, the system can provide novice users with a set of possible gestures' paths with the goal of helping them to recall the mapping between gestures and commands.

For multi-touch systems, Freeman et al. [Freeman 2009] proposed "ShadowGuides" a system for learning multi-touch and whole hand gestures. In line with OctoPocus, ShadowGuides provides users with continuous feedforward and feedback during gesture articulation. ShadowGuides interpret the current hand pose (feedback) and informs users about the possible hand poses and paths to complete the gesture (feedforward). Ghomi et al. [Ghomi 2013] proposed "Arpège" a dynamic guide for learning multi-touch chord gestures vocabularies. Arpège provides users with the position of fingers that have been identified (feedback) and indicates to users the current possible chords and how to make them (feedforward). Feedforward and feedback systems have been also designed for mid-air gestures interfaces such as LightGuide [Sodhi 2012]. LightGuide guides users to articulate a gesture by providing them with a visual feedback hints directly onto the user's hand during the gesture articulation.

2.2.3 Active Exploration and Memorization of Gestures

Active exploration of a set of gestures and their mappings to commands has been used to enhance the transition from novice performance level to expert performance level. Roughly speaking, active exploration consists in pushing the user to repeatedly perform gestures until their corresponding commands are memorized and learned [Appert 2009, Cockburn 2007]. In this context, more efforts can be required for users in the learning stage in order to enhance their practice of gestures and to guide them towards an expert mode. To give an illustration, Kin et al. [Kin 2011] trained users to use multi-stroke marking menu by giving them a stimuli in the form of arrows that indicated the strokes they should draw. Each participant is given an iPod touch to use for training over five days. Each day, users are invited to spend 45 minutes to perform the gesture set in order to improve memorization and learning. In finger count [Bailly 2012], users can navigate in menus just by adding or removing fingers. Users can use the non dominant hand to select menu and the dominant hand to select a desired item. Since the number of fingers is visualized next to each menu and each item, the user can be guided to find the right menu and select the desired item. In the learning stage, users are invited to use their hands sequentially, and as they move to an expert mode, they are invited to use both hands simultaneously.

2.3 Gesture Elicitation Studies and Taxonomies

Although the previously discussed techniques are effective, they all assume that the created gesture set on which a user can be trained was accurately designed. Nonetheless, coherence between a gesture set and the associated commands might be partially or totally inexistent. The choices made by designers can be different from what users would have effectively made, and users can prefer different gesture commands than those proposed by experienced designers [Morris 2010]. Furthermore, multi-touch gesture can by essence take a variety of forms depending on the considered technologies, on its designers and on the concerned users. The work of Wobbrock et., [Wobbrock 2009] captured extremely well the many degrees of freedom in multi-touch gestures while noting that: "surface gestures are versatile and highly varied—almost anything one can do with one's hands could be a potential gesture". To deal with the versatility of surface gestures, a widely adopted approach consists in involving users in the design of gesture sets. This has led to the emergence of several gesture elicitation studies as well as multi-touch gestures taxonomies with the aim of a better understanding, classification and design of multi-touch gestures. This is described in more details in the following.

2.3.1 Gesture Elicitation Studies

Several studies can be found in an attempt to grasp and unify the rich vocabulary of gestural multi-touch interaction and pushing forward the process of developing surface computing technologies, *e.g.*, [Morris 2006, Long 1999, Karam 2005]. Given the versatility of free-hand multi-touch gestures and the high variety of users behaviors in producing them [McNeill 1992], user-centric approaches that inform the design of multi-touch gestures have been at the heart of many research studies, particularly user-elicitation studies. User-elicitation studies are a specific type of participatory design methodology that involves users in the design of gesture-sets. Rather than bounding users to an arbitrary set of gestures defined by system designers (*e.g.*, [Malik 2005, Rekimoto 2002, Tse 2006]), Wobbrock et al. [Wobbrock 2009] adopted a guessability methodology [Wobbrock 2005] to build up a user-defined gesture set for classical control actions and object manipulations. Finding its fundamentals in [Nielsen 2004b, Schuler 1993, Epps 2006], this approach consists in presenting a referent (the effect of an action) to users and then asking them to invent the corresponding gesture. The users are also asked to perform the gesture using one hand, then using two hands. The gesture which was consistently performed by the largest number of users is then retained to be representative of the corresponding action. The set of gestures defined by this work are physically and conceptually more simple than those proposed by researchers. A follow-up study made by Morris et al. [Morris 2010] showed that users preferred gestures designed by end-users over those designed by 2-3 researchers. Gestures defined by a single designer were the least preferred. Recently, Rust et al. [Rust 2014] have extended wobbrock study by asking children to create for each action a gesture in an attempt to understand whether children and adults define similar gestures.

Gesture elicitation studies have been employed in several other interaction contexts. For instance, Cohé et al. [Cohé 2012] have employed this methodology to examine how users perform gestures for 3D manipulation including rotation, scale and translation of a 3D cube on a 2D touch screens. Buchanan et al. [Buchanan 2013] focused on 3D manipulations of physical objects on multi-touch displays. Ruiz et al. [Ruiz 2011] presented a study to elicit user motion gestures on a smartphone device. Kray et al. [Kray 2010] presented investigations on eliciting gestures from users in which they span between mobile phone, interactive tabletop and large displays. Kurdyukova et al. [Kurdyukova 2012] evaluated how users transfer data between two iPads, iPad and tabletop and iPad and public display. Seyed et al. [Seyed 2012] designed a user study to elicit gestures from potential users of multi-display environments (MDEs) incorporating multi-touch tabletops, tablet and wall displays. Valdes et al. [Valdes 2014] explored what type of gestures users create for multi-touch and tangible interaction with active tokens.

Vatavu [Vatavu 2012b] and Jansen [Jansen 2012] adapted the same user-centric paradigm in the context of interactive television. In addition, Jansen [Jansen 2012] considered teaching users a set of user-elicited gestures for interactive television: in the first way named imitation, users observed a human arm performing the gesture. Kühnel [Kühnel 2011] presented gesture-based user interface to a smart-home system. They have also investigated how well users could learn user-elicited gestures. Thus, all referent, that is the effect of a gesture, have been rated by the authors according to their complexity. Their findings indicated that users are able to distinguish intuitive mappings between referents and gestures and that intuitive mappings were more remembered.

Within all those studies, valuable discussions are reported about the different characteristics of user-defined gestures. For instance, it is observed in [Wobbrock 2009] that users rarely care about the number of fingers they use, some of them use hand pose gestures and others act both above the table or above and on the table, etc. In addition, for a target action, it was found that several different gestures could be defined by users and that users are more likely to expose very little agreement when choosing a mapping between an action and a possible gesture. Since one concern of these studies is on the design and analysis of a one-to-one mapping between gestures and their actions, researchers have chosen to select for each action only the gesture that have been articulated by the largest number by users and have ignored the rest of users-proposed gestures.

Orthogonal to these work, Henze et [Henze 2010] proposed to derive and compare multiple gesture sets rather than a single one. Their findings indicate that this is a beneficial approach to reduce the risk to exclude promising candidates for gestures. In a field study investigating the variety of gesture performed by people, Hinrichs et. [Hinrichs 2011] found that users choice of gestures was influenced by the context in which the current action occurred and not only based on preferences for a given gesture for a particular action. They suggest that a many-to-one mapping is also desirable to strengthen the design of gestural interaction techniques. While being specific to pen gestures, [Long 2000] studied perceived gesture similarities, which can be viewed as a dual approach explaining the differences between gestures classes [Long 1999]. Recently, Oh et al. [Oh 2013] described investigations on the feasibility of user-customizable gesture commands. Customization refers to allowing every user to customize the gesture that fits him/her best. Their findings showed that users focused on familiar gestures and were influenced by the misconceptions and the performance of gesture recognizers.

2.3.2 Multi-touch Gestures Taxonomies

As sketched in the introduction of this chapter, one can find valuable taxonomies of human gestures [Kendon 1988, Efron 1941, McNeill 1992] which are unfortunately not directly applicable for surface gestures. Inspired by such work, Karam and schraefel [Karam 2005] created a more extensive taxonomy especially tailored for HCI. They classified gestures into 5 classes: deictic, gesticulation, manipulation, semaphore, and sign language. Deictic gestures identify or locate an object. The term gesticulation encompasses "hand movements within the context of the user's speech". Manipulation gestures occur when a "tight relationship between the actual movements of the gesturing hand/arm" applies to an entity.

Four different styles of manipulative gestures have been discussed: (1) gesturing in two degrees of freedom for two-dimensional interactions. For this style, the entity that will be manipulated is displayed on a 2D screen such as a cursor or a window, (2) gesturing in multiple degrees of freedom for two-dimensional interactions consisting in manipulating 2D object through a 3D manipulation such as sweeping a group of object on a table [Wu 2003], (3) gesturing with tangible objects for three-dimensional interactions. In this manipulation style, gestures are used to intact with physical objects that represent digital object and (4) gestures for real-world physical object interactions where gesture are used to allow users controlling physical objects such as robots. Semaphore gestures require a stylized dictionary and Sign Language (SL) is used for conversational style interfaces, requiring individual signs with grammatical structures.

Wobbrock et al. [Wobbrock 2009] are among the first to establish a unified taxonomy for surface gestures based on user's behavior. This taxonomy can actually be viewed as *the* reference for surface gestures. In their work, Wobbrock et al gave a coarse-grain classification along four dimensions: form, nature, binding, and flow. These dimensions are discussed in more details in the following.

The *Form* dimension captures *how* gestures are performed by users. The scope of this dimension is within one hand. In the case where the gesture is articulated by two hands, the form dimension is applied separately to each hand. In this dimension, Wobbrock et al. distinguished between (1) gestures where hand have a static pose *i.e.*, hand pose is held in one location and the case where the hand have a dynamic pose *i.e.*, hand pose changes in one location and (2) gestures that contain a path and those without. The form dimension distinguished also between the case where a single point is detected and the case where more than one point is detected per hand. The form dimension is expanded into 6 classes: (1) static pose *i.e.*, hand did not change position, (2) dynamic pose *i.e.*, hand change location, (3) static pose and path *i.e.*, hand do not move but generate a path, (4) dynamic pose and path *i.e.*, hand move and generate path, (5) one point touch, *i.e.*, static pose with one finger and (6) one point path *i.e.*, static pose and one finger path.

The *Nature* dimension captures the users *semantic* interpretation of gestures. In Wobbrock's taxonomy, the Nature dimension distinguishes symbolic, physical, metaphorical and abstract gestures. Symbolic gestures are gestures that visually depicts a symbol. For example drawing a caret ("^") to perform insert. Physical gestures act physically on objects. Metaphorical gestures indicate a metaphor such as swiping as if to turn a book page. For this kind of gesture nature, the gesture alone is not enough to reveal its metaphorical nature; the answer lies in the user's mental model. Finally, abstract gestures have an arbitrary mapping without symbolic, physical, or metaphorical connection to their referents. For example triple-tapping an object to delete it.

The *Binding* dimension determines whether or not the gesture requires particular information either about the object it affects or produces, or about the application environment. In Woobrock's taxonomy, the binding dimension distinguishes between 4 classes: (1) Object-Centric gestures require information about the object location, and hence, location is defined with respect to the object. For example, selecting a specific location on the computer screen by touching or pointing to it would be Object-Centric. (2) World-Dependent gestures are those which need information about the world such as tapping in the top-right corner of the display or dragging an object off-screen. (3) World-Independent gestures, in contrast, require no information, neither about the object they affect nor about world (the application environment), and generally can occur anywhere on the surface. For example, drawing a check mark with the index finger to perform "Accept" is also World-Independent if it can be performed anywhere on the screen. Finally, mixed dependencies occur for gestures that are world- independent in one respect but world-dependent or object-centric in another. This sometimes occurs for 2-hand gestures, where one hand acts on an object and the other hand acts anywhere.

The *Flow* dimension captures two classes of gestures: discrete and continuous. A gesture's flow is discrete if response occurs after the user acts. The gesture flow is continuous if response occurs while the user acts. These two classes have been previously defined by Wu et al., [Wu 2006].

Wobbrock's taxonomy describes the appearance but also the consequence of the gesture. This taxonomy has been extended or adapted for others gesture based systems including multi-touch and tangible interfaces [Valdes 2014], 3D manipulation on multi-touch screens [Cohé 2012, Buchanan 2013], personal devices such as motion gestures for smartphones [Ruiz 2011], mid-air hand gestures [Aigner 2012], gestures for Augmented Reality [Piumsomboon 2013].

For surface gestures and tabletops, the Form dimension has interested others researchers as it captures how the gesture is performed by users. For instance, Wu et al. [Wu 2006] described the process of gesture performance as a finite state machine, with start position (registration), a dynamic phase (continuation), and end position (termination), similar in concept to that described in Charade [Baudel 1993]. Freeman et al [Freeman 2009] expanded the Form dimension along three dimensions: registration pose, continuation pose, and movement. The registration pose dimension captures what part of the hand is in touch with the surface in the initial touch. Freeman et al distinguished between gestures initially performed with fingertips and those performed with other parts of the fingers or hand (*e.g.*, palm). Accordingly to this, 4 classes emerged from the registration pose dimension: (1) initial touch with single finger, (2) initial touch with multi-finger (*e.g.*, 2 fingers touch initially the surface), (3) initial touch with single shape and (4) initial touch with multi-shape. While the single shape class corresponds to initial touch with a single hand shape such as a palm, the multi-shape class refers to touching the surface with multiple hand shapes which typically corresponds to bimanual input. The continuation pose indicates if the hand pose changes or not after the registration. Thus, two classes emerged from this dimension: (1) static meaning that hand pose remains the same after registration and without relative movement, (2) dynamic meaning that hand pose changes after registration (*e.g.*, new fingers come in contact with (or leave) the surface, or the shape of the hand in contact changes). Finally, the movement dimension refers to whether the user's entire hand follows a path along the surface or not. Two classes emerged from this dimension: (1) no path refers to the hand staying in place and (2) path refers to the hand moving along a surface path. In the case where the hand remains stationary, but the fingers move relative to each other, the gesture is classified as having no path. However, as the posture of the hand changes, but not its location the gesture is classified with a dynamic continuation pose. The movement level corresponds only to the movement of the hand while the continuation pose corresponds to both the hand pose and fingers movement. Actually, the primary intention of the work of Freeman et al [Freeman 2009] was to describe ShadowGuides a gesture learning tool and to evaluate its performances on a prefixed representative gesture set spanning different combinations of the so-defined dimensions. Although the taxonomy of Freeman et al presents a sound picture of the large variety of multi-touch gestures, it strictly dissociates the Form of a gesture from its Nature. With this respect, it finds some limitations in informing the different choices that users can make to express the same semantic when performing a gesture.

In this thesis . . .

In Chapters 3 and 4 we design new user studies to inform about the versatility of user defined gestures. We contribute a fine-grain analysis informing the preferences of users and the versatility of their choices when performing a gesture, with an attempt to leverage previous findings and to provide new insights into the mental perception of gestures in users' mind. In particular, we contribute in Chapter 3 a taxonomy harnessing multi-touch gestural variations occurring at mental, physical, and movement levels, as a result of a user-centric study. We enlighten the features that deserve deeper modeling efforts for future system-oriented gesture formalizations, thus shortening the gap between the users and the system.

2.4 Gesture Analysis Studies and Predictive Models

Complementary to user-centric studies and gesture elicitation, gesture analysis and modeling is another approach that informs about the characteristics of gestures. In this section, we give an overview of gesture analysis studies and predictive models that have been investigated so far.

2.4.1 Gesture Analysis Studies

To harness the versatility of users and its impact when articulating gestures, a key aspect consists in a better understanding of the characteristics and the proprieties of gestures. In particular, the path of stroke gestures has been analyzed extensively by using different geometric and kinematic features. An interesting set of 13 geometric and algebraic features can be found in Rubine's work [Rubine 1991a]. Recently, Blagojevic et al. [Blagojevic 2010] have defined a comprehensive set of 114 features to represent the articulation path of stroke gesture. However, while these features give a fine grain details to characterize the articulation performance, they have basically serve for recognition purposes [Rubine 1991a, Blagojevic 2010, Vatavu 2012a, Willems 2009] rather then for a better understanding how users articulate gestures.

Anthony et al. [Anthony 2013b] examined the consistency of users' pen and finger gesture articulation. The analysis were made both between and within users. They have captured execution variation in gesture by employing two kind of articulation features: (1) geometric features including number of strokes, path length, area of the bounding box, cosines of both starting angle and ending angle, line similarity, global orientation, total tuning angle, sharpness and curviness and (2) kinematic features including production time and average speed. Their findings indicated that there is a high degree of consistency within users. However, a lower consistency has been found between users. Besides, they found that the more users are familiar and comfortable with a gesture (*e.g.*, letters and numbers are commonly written) the more the variation in articulating the gesture is lower. This result is valid for both within users and between users. In addition, they found that increased geometric complexity of shapes leads to people producing gestures less consistently where a log-linear relationship was found between the number of strokes and consistency. With respect to gesture kinematics, their findings show that the more users articulate their gestures faster, the more the agreement among users increases. Finally, they found that smaller gestures have higher agreement which can be explain by the fact that less variations are possible when performing smaller motions.

Vatavu et al. [Vatavu 2013c] defined 12 relative accuracy measures for single and multi-stroke gestures. These measures aims at capturing what happens during gesture articulations compared to their ideal, called "references". Reference gestures are determined by computing the average gesture from a set, and then finding which articulated gesture is closest to that average. The geometric accuracy captures how well users are able to reproduce a gesture, given its geometric shape alone. 6 measures are defined: (1) shape error refers to the average spatial deviation from a reference gesture, (2) shape variability reports the total spatial deviation from a reference gesture, (3) length error measures an amount of "stretch" relative to a reference gesture, (4) size error computes the amount of space consumed relative to a reference gesture, (5) bending error measures the average "turn" relative to a reference gesture and (6) bending variability computes the total "turn" relative to a reference gesture. Kinematic accuracy captures differences in the time domain and, therefore, informs how fluent or smooth the gesture path is. To this end, 4 measures are defined: (1) time error refers to the average temporal deviation from a reference gesture, (2) time variability reports the total temporal deviation from a reference gesture, (3) speed error is an average deviation in speed compared to a reference gesture and (4) speed variability represents the total deviation in speed compared to a reference gesture. Finally, the articulation accuracy computes how consistent users are in producing the individual strokes of gestures and is defined using two measures: (1) stroke count error reports the difference in number of strokes compared to a reference gesture and (2) stroke ordering error indicates how similar the stroke ordering comparing to a reference gesture.

Kane et al. [Kane 2011] argued that touch screens should be usable by all users with all abilities, and thus, designers should have a better understanding of how people actually use touch screens in order to guarantee a more accessible interaction. For this purpose, they set up two user studies in which they examined the difference between blind people and sighted people when articulating stroke gestures. In the first study, participants are asked to invent gestures that could be used to conduct standard computing tasks on a touch screen-based tablet PC. Their findings indicate that blind participants' gestures contained significantly more strokes than those produced by slighted people. They also found that blind people were more likely to choose gestures that used the edge or corner and they were also significantly more likely to invent multi-touch gestures with at least two simultaneous contact points than sighted people. In addition, while a higher number of symbolic gestures was invented by sighted people, a higher number of abstract and metaphorical gestures was invented by blind people. In the second user study, participants are asked to articulated specific gestures. Their findings indicate that blind people articulate significantly larger gestures with more variation in size when performing the same gesture many times than sighted people. Gestures articulated by blind people were also found to be slightly wider than those articulated by sighted people. In addition, blind people took approximately twice as long to articulate the same gesture. Kane et al. have also analyzed specific gesture features including (1) location accuracy where they found that all participants were able to perform gestures at any location on the screen, (2) form closure where they found that gestures performed by blind people were more likely to have start and end points that did not coincide and (3) line steadiness where they found that lines created by blind people tended to be less steady than lines created by sighted people.

Tu et al., [Tu 2012] conducted an empirical study to understand the impact of devices on stroke gestures. They rely on a user study where participants were asked to produce a set of stroke gestures with varying degrees of complexity and in different target sizes. Their goal is to understand the differences and the similarities between finger-articulated gestures and pen articulated gestures. Their findings indicate that gestures articulated with fingers are different from gestures articulated with pen in terms of size and average speed. For instance, gestures articulated with finger tend to be lager and faster than those articulated with pen. The shape geometry of gestures drawn with pen are also different from gestures drawn with a finger. Since the finger may more severely obscure the start point when getting close to it, gestures produced with a finger have larger aperture between the start point and the end point than gestures produced with a pen. They also defined two measures "Corner Shape Distance (CSD)" (*i.e.*, the mean distance between the corresponding corners in the drawn gesture and the target gesture) and "Intersecting Points Deviation" (IPD) (*i.e.*, the mean distance between the intersecting points in the drawn gesture and the target gesture). A significant effect was found for CSD and IPD between finger and pen gestures. Their findings indicated also that there were similarities between pen articulated gestures and finger articulated gestures including articulation time, indicative angle difference, axial symmetry and proportional shape distance. With respect to shape features, gestures articulated with a pen have a more accurate axial symmetry and are more exact than gestures articulated with a finger in the case of complex gestures. In contrast, finger gestures are found more accurate than pen gestures for simple gestures.

Anthony et al., [Anthony 2013a] examined the need for visual feedback during gesture interaction on mobile touchscreen devices. They have set-up a user study in which participants (including children, teens, and adults) are asked to draw different gestures types under two conditions: with and without visual feedback. From collected data, they extracted a set of features to represent the gesture articulation including number of strokes, number of points, gesture length, gesture height, gesture width, gesture area, gesture duration, gesture pressure, and gesture speed. Their findings indicate that there is a significant difference between gestures articulated with and without visual feedback between users of different ages: the more younger children, the more there were variation in the gesture articulation. Overall, users were more careful when articulating gestures in presence of visual feedback: gestures were shorter, more compact and contained more strokes. Finally, although users of all ages preferred to see visual feedback, adults were more willing to accept the lack of feedback.

Kin et al., [Kin 2011] examined the speed and accuracy of one and two-handed multi-stroke marking menus. They compared the time performance of two-handed simultaneous gestures (*i.e.*, two hands are used in the same time to draw different strokes), two handed ordered gestures (*i.e.*, hands are alternating one after the other to draw the different strokes), single left hand gestures and single right hand gestures. Their findings indicate that two handed simultaneous gestures outperformed the single dominant handed technique, and that the two handed ordered technique is not significantly faster than the one handed technique. Wobbrock et al., [Wobbrock 2007] reported the effect of gesture articulation speed on the recognition rate under the speed conditions: slow, medium and fast. They found that speed has a significant effect on users articulation errors. They also found that different recognizers (DTW, Rubine and 1$) are affected similarly.

In this thesis . . .

In Chapter 4 and in Chapter 5, we specifically address multi-touch gestures which is to contrast to the work presented in this section which has been done for stroke pen gestures or one finger gestures. We contribute a set of descriptors to characterize multi-touch gesture articulation in terms of (1) gesture structure, (2) geometry and visual appearance, and (3) kinematics. In Chapter 4, we mostly contribute in a comprehensive analysis to characterize different gesture classes in terms of the so-defined descriptors. In chapter 5, we provide a detailed study to understand the effect of number of fingers, number of strokes and number of hands on gesture articulation characteristics.

2.4.2 Predictive Models

Among the main issues that can occur when creating a gesture set is that a gesture can be difficult to remember, and mis-recognition can make users frustrated during interaction [Long 1998]. To limit these problems, researchers have proposed a set of tools and models that help the creation and the evaluation of designed gestures, and eventually to improve them early in the design process.

Long et al., [Long 2000] have proposed a model for predicting the visual perceived similarity of two gestures. To generate their model, Long et al have selected a subset of geometric and dynamic features. Their model is able to reasonably predict similarity of two gestures. Predicting visual similarity is useful as it can inform about how easy the learning and the memorization of the gesture by users can be. For instance, for similar operations designers could assign visually similar gestures such as scroll up and scroll down, and conversely for abstract operations designers could assign visually dissimilar gestures such as cut and paste. Furthermore, such predictive models can be used to improve recognition rate by avoiding ambiguous gestures. Long et al. relied on a user study and were also able to extract other findings such as gestures articulated with different scales are perceived as dissimilar. This issue has been investigated by Vatavu et al., [Vatavu 2013d] to define a predictive model to estimate the user perceived scale of stroke gesture. The motivation of such a model is that the gesture scale can be exploited to improve the mapping between gestures and their functions. It can also help users by reducing the need of learning and recognizers by reducing the need to discriminate between unnecessary symbols.

Isokoski, [Isokoski 2001] proposed a model for estimating the writing time of an unistroke alphabet. The model is based on geometric complexity and was designed for expert users. In the same spit, Cao et al., [Cao 2007] proposed "Curves, Lines and Corners" (CLC) a model to predict the actual production time of a single stroke gesture. For single stroke pen gestures, Vatavu et al., [Vatavu 2011b] proposed a model for estimating user's perceived difficulty. Estimating the execution difficulty of articulating a gesture is very useful as the execution difficulty can be incorporated at different levels including the ease with which a gesture may be learned, remembered, and articulated. Vatavu et al. asked participants to articulate different single stroke gestures for several times. Participants were then asked to rank and to rate gestures according to their difficulty. The perceived execution difficulty was then estimated by first calculating a set of geometric descriptors characterizing the gesture articulation and then correlating those measures with the rating and ranking scores collected form users' responses. Using Bayes' classification rules, they derived a model to estimate the class of execution difficulty of a given gesture which has been validated by setting up a second experiment similar to the first one but with new gesture types and new participants.

In this thesis . . .

In this dissertation we specifically address multi-touch gestures which is to contrast to the work presented in this section which has been done for stroke pen gestures. In particular, we provide in chapter 5 a detailed study to understand users' perceived difficulty of multi-touch gesture articulation under the many degrees of freedom afforded by multi-touch input including the number of fingers touching the surface, the number of strokes that structure the gesture shape, and single-handed and bimanual input. We do not set up a predictive model for multi-touch gestures, but we outline a set of guidelines that can help designers when defining a gesture set.

2.5 Gesture Recognition Techniques

Studies investigating how users articulate and define gestures have many often shown differences in gesture articulation among users, which justifies the need to capture users' gesture variations as part of the recognizer. Given the rapid emergence and development of touchscreen devices, there is evidence to dispose of accurate multi-touch recognizers that are robust to user variations and, at the same time, easy to deploy for user interface prototypes. To give an idea on the wide literature on gesture recognition techniques, this section is organized as follows. We start by reviewing well-established recognizers that have been used in the past for handwriting recognition, followed by the simple but accurate $-family recognizers for single touch and pen gestures, and finally we mention about less popular recognition frameworks.

2.5.1 Towards Recognizing User Interface Gestures

Various approaches for gesture recognition have been proposed in the state-of-the-art literature including Hidden Markov Models (HMMs) [Sezgin 2005], Dynamic Time Warping (DTW) [Myers 1981, Keskin 2011], neural networks [Bozinovic 1982] and statistical classifiers [Rubine 1991b]. All these recognizers have been mainly designed to recognize on-line and off-line handwriting. As our concerns in this dissection is on recognizing gesture for user interface, a brief discussion of these techniques and their main limitations for user interfaces are provided (for further details, the readers is referred to [Plamondon 2000, Tappert 1990]).

The previous mentioned techniques deliver robust and high accuracy rate for gesture recognition. However, most of them use sophisticated algorithmic techniques. For instance, HMMs, neural networks, and statistical classifiers should be trained with numerous examples. Thus, these techniques are not practical for UI prototypes in which application end-users define their own gestures. Besides, these techniques are hard to program and debug which can represent an additional challenge for UI designers and prototypers. For example, in the case of Rubine's popular classifier [Rubine 1991b] which is relatively easier to program than HMMs, one has the calculate of 13 gaussian distributed features. Similarly, dynamic programming methods are expensive to compute [Zhai 2003]. Some work have been done to improve the speed of these methods [Salvador 2007]. However, these improvements are beyond the scope of most user interface designers. Finally, ad-hoc or specific improvements can not be easily adapted and do not allow end-users to define new gestures. With respect to Rubine's classifier [Rubine 1991b] an extension to multi-touch gesture recognition has been proposed in [Rubine 1992]. However, as the ordering of touch trajectories is crucial for delivering correct recognition, gesture types that can lead to ambiguity were avoided. Furthermore, by this definition, the classifier does not support configurable speed, rotation, scale, and position invariance.

2.5.2 $-family Recognizers

For symbolic gestures, the popular $-family recognizers [Anthony 2010, Anthony 2012b, Vatavu 2012c, Wobbrock 2007] allows to deliver robust recognition accuracy with a set of simple techniques. $-family are easy to be incorporated into user interface prototypes. We review the different recognizers of this family in the following paragraphs.

$1 and $1-Protractor recognizers. $1 [Wobbrock 2007] and Protractor [Li 2010] are designed to recognize 2D uni-stroke gestures and support configurable rotation, scale, and position invariance. The name $1 was chosen in order to emphasize the fact that the prototyping and the integration of this recognizer is very cheap. To recognize a gesture, $1 (and all $-family) is based on template matching. It compares an articulated candidate uni-stroke C to a set of stored templates T. The template T_i closest to C is the recognition result. The "closeness" criteria is determined by the average Euclidean distance between the points in C and T_i. To increase flexibility and accuracy, multiple templates T_i can be stored for the same gesture type. Besides, as candidate and template gestures are all uni-strokes, a mis-recognized candidate can immediately be added as a new template which allow users to teach $1 at runtime.

An important step to successfully recognize a gesture is to properly define the points to consider in the template and candidate gestures. For this purpose, $1 uses four preprocessing steps on both templates and candidates. First, gestures are resampled into a fixed number of points such that all resulted points are spread equidistantly along the stroke's path. Second, the stroke path is rotated such that its "indicative angle", defined by the centroid to the first point, is at 0°. Third, the stroke path is scaled non-uniformly to match a reference square. Finally, the stroke path is translated so that its centroid is at the origin. These four steps normalize all stroke paths such that each point in a candidate gesture corresponds to exactly one point in a template gesture. The recognition result corresponds to the template T_i with the least point-to-point distance from the candidate. Protractor [Li 2010] is an extension of $1 that improves recognition speed. Although $1 is accurate and simple, it has some limitations: (1) it recognizes only one-stroke gesture and ignores gestures comprising multiple strokes, (2) it recognizes only 2D-gestures meaning that 1D gestures like lines cannot be recognized (3) it uses full rotation invariance which implies that symbols differing only by orientation (*e.g.*, A vs. ∀) can not be distinguished.

$N and $N-Protractor recognizers. $N and $N-Protractor [Anthony 2010, Anthony 2012b] cover the 3 main limitations of $1. The 'N' in the name of this recognizer refers to its ability to handle multi-stroke gestures. The main idea of $N is in fact to view a multi-stroke gesture as a uni-stroke one by simply connecting the endpoints of individual strokes. However, the stroke order and direction may differ among users when articulating the same gesture type. Thus, to retain user independence, $N needs to generate and store all the possible "uni-stroke permutations" for a given multi-stroke gesture. Hence, for a gesture that contain N strokes, 2^N combinations should be treated. To limit the negative impact of this step, $N applies process the permutations of multi-strokes only once for template gestures. Then, at runtime, for each candidate multi-stroke gesture, the articulated strokes are simply connected in the order drawn to form a uni-stroke which is compared to all template uni-stroke permutations (using Euclidean distance as in $1). $N-Protractor [Anthony 2012b] is an extension of $N that improves recognition speed.

Although $N is accurate and simple, the order of strokes and sampled points is still an important issue which affects considerably recognition and negatively impacts memory usage, execution time and system performance. This limitation is covered by $P.

$P recognizer. $P [Vatavu 2012c] is the most recent recognizer of the $-family. It avoids the storage complexity of $N by representing gestures as clouds of points, which also makes it independent of how users actually articulate gestures. The 'P' in the name of this recognizer refers the 'cloud point' paradigm. In fact, the $P recognizer departs from it predecessors, *e.g.*, $1, $N [Wobbrock 2007, Anthony 2010], by avoiding the need to permute and store gestures by stroke order and direction. The set of strokes composing a gesture is simply converted into a unique set of points called *cloud*, regardless to the number of strokes and to the timeline of points in each stroke. By viewing a gesture as a cloud of timeless points, $P attempts to compute a matching distance between a candidate gesture and each training gesture template. Having a candidate gesture C to recognize and a training gesture template T, a matching function $\mathscr{M} : C \longrightarrow T$ refers to the assignment of every point in C to a unique point in T. The matching distance of C and T with respect to $\mathscr{M}$ is then defined by the formula:

$$\sum_{p \in C} \sqrt{(p.x - \mathscr{M}(p).x)^2 + (p.y - \mathscr{M}(p).y)^2}$$

where p denotes a point of C, $\mathscr{M}(p)$ its matching point in T. With this definition at hand, $P first proceeds by resampling gestures using the same number of points than $N. Then, the main task of $P is to find for each training template T, the matching function $\mathscr{M}$ providing the best minimum matching distance. For this purpose, $P uses a greedy heuristic which is proved to be substantially faster and easier to implement than the exact method for the same problem while providing essentially the same accuracy in terms of matching distance. The output of $P is then the template having the smallest matching distance computed by the greedy heuristic.

$-family recognizer for multi-touch gestures. All $-family recognizers were mainly designed and validated for single-touch gestures, although some attempts to adapt them for multi-touch gestures exist. Jiang et al. [Jiang 2012] extended $1 to recognize multi-finger single stroke gestures by aggregating touch point trajectory into a single stroke. This idea inherits both the strength but also the weaknesses of the original $1 recognizer. More importantly, the reduction developed in [Jiang 2012] to transform all touch points in a single stroke is incompatible with the fact that multiple strokes can interleave in time. Oh et al. [Oh 2013] explored the potential recognition rates that can be obtained for the custom gesture sets they were targeting. However, no detailed description of how $N was applied and the reported accuracy was found to be relatively low (at most 88%).

2.5.3 Other Tools and Frameworks

A set of user interface toolkits have been proposed, *e.g.*, SATIN [Hong 2000], Artkit [Henry 1990], Amulet [Myers 1997] in order to support the incorporation of gesture recognizers in user interfaces. Garnet [Myers 1990] has been designed to aid the design and the implementation of highly interactive graphical direct manipulation user interfaces. Agate [Landay 1993] has extended Garnet User Interface Development Environment while using the Rubine classifier [Rubine 1991b].

Several programming libraries have also provided APIs to support gesture recognition on specifics platforms. For instance, the Siger library for Microsoft's Tablet PC [Swigart 2005] allows developers to define customer gestures for their applications on tablet PC. The Siger recognizer converts the points of a stroke into a sequence of directions and then match them using regular expressions and heuristics. This library is designed for pen gesture.

Turning to a more formal perspective, some remarkable researches have being conducted on the reliable recognition of multi-touch gestures. GeForMT [Kammer 2010] provides a formal abstraction of multi-touch gestures using context-free grammar. Gesture Coder [Lü 2012] recognizes multitouch gestures via state machines. Proton [Kin 2012b, Kin 2012a] describes multi-touch gestures as regular expressions modeling a whole sequence of touch events. GestIT [Spano 2012] is a proof of concept library implementing a meta-model based on compositional operators and Petri Nets to describe multi-touch gestures. "Concepture" [Donmez 2012] is a framework based on regular language grammars for authoring and recognition of sketched gestures with repetitive patterns. All these technical tools, frameworks and languages provide system sound specifications allowing to express complex multi-touch gestures. Nevertheless, it is not obvious how they can apply to capture in a comprehensive and faithful manner the behavior and variability of non technical users in producing gestures. For instance, the number of fingers and strokes as well as their space-time combination are predefined by the designer.

In this thesis ...

In Chapter 6, we contribute a technique for recognizing multi-touch gestures under unconstrained user articulation behavior. Our technique employs a stroke clustering procedure that consistently structures the key strokes of a multi-touch gesture. The result in the form of a multi-touch point cloud is fed into the $P recognizer, leading to improved recognition accuracy of multi-touch gestures in a manner that is independent of gesture articulation.

2.6 Chapter Summary

In this chapter, we highlighted different research issues that deal with the design, the understanding and the recognition of multi-touch gestures while sketching and positioning our contributions with respect to existing work.

In the rest of this thesis, we shall describe in more details the different results that we are able to find during our research investigations. In particular, we organize the rest of this dissertation into two parts which are directly motivated by the different aspects discussed in this chapter. In fact, we showed that adopting a user-centric approach and studying user-defined gestures can be beneficial to multi-touch interaction in several aspects. Hence, in the first part of this thesis, we shall systematically adopt a user-centric approach in order to enhance our understanding of multi-touch surfaces. We shall also provide a throughout investigation on the properties of users-defined gestures and study their perceived difficulty. In the second part of this thesis, we turn to a more system perspective where we provide a set of tools aiming at enhancing and strengthening interaction. We first show how to extend existing recognizers with a robust and simple technique that allows unconstraint gesture articulation. We then show how users freehand movements can be consistently classified by the an interactive system in order to broaden and to enlarge the design space of multi-touch gestures while being as much near as possible to users' abilities.

Part I

Understanding Users' Unconstrained Multi-touch Gestures

"Surface gestures are versatile and highly varied—almost anything one can do with one's hands could be a potential gesture."

Jacob Wobbrock, Meredith Ringel Morris and Andrew D. Wilson (2009)

3 A Multi-Level Taxonomy Of Users' Multi-touch Gesture Variability

Multi-touch gestures are often thought by application designers and do not necessarily take into account the high variability of users' gestures. The approach taken in this dissertation is to first establish a fine grain analysis on user gesture variability for multi-touch surfaces, in order to properly design tools and techniques that effectively handle user variability and improve user interaction experience. Towards making this approach effective, we setup the first user study to understand user gesture variability on multi-touch surfaces. The main research issues that we are addressing in this chapter can be formulated as follow:

- What type of gestures users produce when they are asked to create gestures for a general, open-ended use context?
- How users perceive and articulate a multi-touch gesture?
- How do people articulate symbolic gestures using their hands on tabletops, and how do they express variability for the same gesture type?
- What are users thoughts and priorities in conceiving the different articulated gestures for the same gesture type?

In the following, we describe the user study we set up in order to answer the previous questions. Our user study shall allow us to introduce a new taxonomy of users' gestures and also to provide a set of quantitative results informing about some characteristics of users' produced gestures. The chapter concludes with a detailed discussion on the implications of our results for gesture design, surface technology and user interfaces.

3.1 User Study

User-centered design is a cornerstone of human-computer interaction. For instance, eliciting user behaviors have been proved extremely useful to help the design of new, strong and flexible interaction tools (*e.g.*, [Schuler 1993, Wobbrock 2009, Cohé 2012, Guilford 1967]. Users are not designers; therefore, care must be taken to elicit user behavior profitable for design. As sketched in the previous introductory questions, our main goal is to broaden the range of possible responses we can get from users and to gain insights into users' variability when issuing a multi-touch command. We want to elicit the different ways users perceive and issue a multi-touch gesture with the ultimate goal of determining the rules leading to a better definition of what shall be a gestural language for multi-touch interaction.

For that purpose, we conducted a user study composed on two tasks. The first matter of interest was to make a step toward many-to-one mappings between user gestures and commands, by understanding user gestures variability for multi-touch systems. Most studies done in the state of the art are task oriented, and allow to exhibit best matches between gesture type, and elementary tasks or commands. Some results show little agreement among users in mapping between gestures and their effect [Wobbrock 2009]. In order to provide application designers with knowledge that will help designing good many-gestures-to-one-command mappings, we need another experimental approach, which is to exhibit, for a specific type of gesture, all the possible gestural representations that users may achieve. Instead of studying relation between gestures and tasks, we propose to study relation between gesture and underlying symbolic pattern.

We then ask participants to appeal to their imagination to perform different gestures, at the aim of grasping and analyzing the variability and dynamic of user behavior. As a result, our proposed study is divided in two tasks: a first task in which we familiarize participants with the interactive surface, and more importantly, we observe and analyze their interaction styles, within an uncontrolled experimental procedure where user can both choose gesture type and gesture that represents it. This task is intended to construct a taxonomy that is used as a basis for the remainder of the study. In the second task, we achieve quantitative analysis of how users articulate symbolic gesture types, using explicit instructions (name of the gesture type, number of variations) and asking participants to explore the different articulations to achieve specific gesture type.

Our user study contributes a fine-grain analysis informing the preferences of users and the versatility of their choices when performing a gesture, with an attempt to leverage previous findings and to provide new insights into the mental perception of gestures in users' mind. This section describes our approach to understanding users' variability when issuing a multi-touch gesture in unconstrained conditions.

3.1.1 Participants

A call for participation has been made using mailing lists and an ad on the advertising lobby screens available at our lab and its institutional partners. The call concerned people who were not user-interface designers and not computer scientists on human computer interaction. In final, a total of 30 volunteer participants responded in the positive, among them 14 were female. Participants' ages varied between 20 and 57 years (mean 28.4 years). All participants were right-handed. Participant occupations included secretary, chemists, biologists, electronic and mechanics experts, researcher in networks and telecommunications and graduate students. Participants nationalities include different European, African and Asian countries. In terms of self-reported familiarity with touchscreen devices, the expertise of participants were found to significantly differ by the type of interactive surface ($\chi^2(3)$=62.27, p<.001). In Table. 3.1 summarizing users familiarity, we can see that none of our participants has previously used neither a Tablet Pc nor Tabletops, thus making them completely novice to the interactive surface used in our experiments.

	Smartphone	**Tablet**	**Tablet PC**	**Tabletop**
Never	8	18	30	30
Occasional	9	9	0	0
Regular	13	3	0	0

Table 3.1: Distribution of usage of touchscreen devices among our participants.

3.1.2 Apparatus

The study took place in our lab, where we had set up a Microsoft Surface prototype measuring 24”$*$18”. Only the experimenter and the subject were present during the study. In order to prevent any screen content from influencing the gestures participants were performing, we provided no visual feedback of gesture input. We do not also provide participants with gesture recognition feedback. To record performed gestures, we installed a video camera in face of the subject and the experimenter was located between them (Figure 3.1). Then at each session, at the beginning the participant is asked to sit in front of the interactive surface and to adjust the chair. Then participants’ hands were videotaped and the experimenter observed user behavior and took detailed notes. The think aloud protocol was used throughout all tasks. Care was taken throughout the study to reduce bias. For instance, the experimenter did not demonstrate or provide example gestures while communicating instructions (*e.g.*, single finger vs. multi-finger, single hand vs two-hands etc.). Before beginning the first task, participants completed a background questionnaire to collect demographic information and previous touchscreen experience.

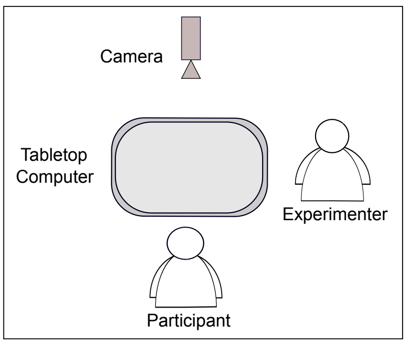


Figure 3.1: Top view of study setup: the video camera was facing the participant and was recording each session. The experimenter was located between them.

3.1.3 Procedure

Our study consisted of the following two tasks

Task 1: Open-ended Gestures Creation. The interactive surface is presented to the participant and he/she is explained that the surface accepts multiple fingers. The participant is then asked to perform any multi-touch gesture that comes to his mind and meaningful to him/her. No further comment or request is made that may suggest degrees of freedom, such as number of hands to use, number of fingers, etc. For each performed gesture, users are asked to describe it in a think-aloud protocol. Participants have 3 minutes to represent all the gestures they can think of; they are also free to stop before 3 minutes in case they consider the task to be over or after if they still have gestures to propose.

Task 2: Articulating Gestures for Specific Symbolic Gestures. The second task provided a more realistic context of creating gestures for 8 specific symbolic gesture types. For each gesture type, the participant is asked to propose four different gesture articulation to draw it. We do *not* show the participants any image of the required gesture types, but only give them an oral instruction consisting of the name of the gesture type to perform.

To reduce bias [Oh 2013], none of the tasks of the user study reported in this chapter were conducted with a prototype where recognition was used to respond to the user. In all cases, the system allowed multi-touch input in unconstrained articulations conditions, but instructional feedback was not provided to prevent any visual content from influencing participants into how they articulate gestures.

3.1.4 Gesture Set

We employed the following 8 gesture types in our second task: circle, square, triangle, vertical line, horizontal line, corner, V and Caret (see Figure 3.2). The gesture set is based on those found in other interactive systems [Anthony 2012a, Anthony 2010, Vatavu 2011b, Wobbrock 2007]. Gestures were selected to be general and simple enough so that participants could reproduce them without a visual representation and thus encourage unconstrained articulation behavior.

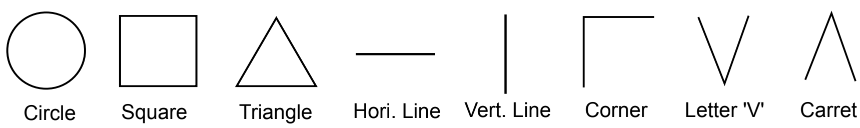


Figure 3.2: The set of 8 gestures used in our experiment.

3.1.5 Design

Our study was a within-subjects laboratory study consisting of two tasks. Both tasks are conducted consecutively for each participant with a small pause in between. Participants were not constrained by any timing issues when performing their gestures. The user study took around 20 minutes to complete. For both tasks, participants were *not* provided feedback, traces, or any system specific elements that could influence their gestures while touching the interactive surface *e.g.*, [Wobbrock 2009, Jiang 2012]. While many participants were asking about the goal of the experiment before even knowing about the instruction, we courteously restricted our answers to providing the procedure of the first task so that participants feel comfortable.

3.2 Task 1 Results: Open-ended Gestures Creation

In the following, we report results from the first task. Our results include qualitative observations, a gesture taxonomy and the concept of atomic movement.

3.2.1 Gestures Collected Characteristics

In the first task of our study, we collected 618 user-made gestures. The number of gestures per participant ranged from 8 to a maximum of 46 gestures. Although collected gestures were having broad properties, similar features were observed among different participants. Without surprise participants produced different forms using interchangeably one or more fingers and one or two hands, to draw different kinds of symbolic gesture types (*e.g.*, line, circle, square, triangle, etc), alphanumeric characters (i.e, letters and numbers), shaped (*e.g.*, tree, heart, flower, star, bird). 6 participants, being regular users of iPads or Smart Phones, additionally performed gestures mimicking standard control actions such as double tape, or rotational, translational, scaling patterns and selecting a group of object.

From our collected data, we were also able to extract several observations about the physical engagement of participants. In particular, 26 (*resp.*, 27) over the 30 participants have used at least once a single (reps. two) hand(s). 24 participants moved simultaneously both hands in symmetric poses. 4 participants alternated from one hand to another in a sequential style. 6 participants used one hand to perform a gesture while their second hand was hold in a stationary pose as to draw a static reference guiding the other hand. This observation holds for gestures performed with a single hand using the thumb and the index. 1 participant used to move hands in the air and touching the surface with her fingers from time to time. 1 participant used exclusively static hand posture on the surface. Except for these two cases, the relative movement of participants fingers was the rule guiding the achievement of gestures. Neither the number of fingers nor their type seemed to us as a conscious parameter that participants were intentionally thinking about. We did also notice no particular preference on the start and the end positions of performed movements. Participants mostly used their right hands when moving from left to right, and inversely they used their left hands when moving from right to left. However, we did not notice other apparent rules applying to the direction of movements nor to the size of their trajectories.

3.2.2 Atomic Movement Concept

The very recurrent observation in participants' behavior is that they grouped their fingers into unitary blocks moving in a consistent manner, while being completely free from the microscopic timeless notion of touch as may be handled by the system. We found that number of contact fingers does not impact the accomplishment of their movements, as long as involved fingers are close to each others. The interesting observation is that the notion of proximity is relative to the gesture type performed and also to user-proper referential and seems to be hardly definable in absolute and universal manner from a system point-of-view. Users referential can in fact be substantially scaled up or down from the performance of one gesture to another one. However, it tends to stay constant and consistent over time and through possibly multiple movements composing the same single gesture. For example, one participant is observed using two hands simultaneously with multiple fingers by hand to draw a circle such as each hand is drawing a half of circle. The same participant is observed to use his both hands simultaneously moving from up to down of the surface and miming that he is translating all images that are in-between his hands. For these two examples, the relative distance between fingers composing the same produced movement is relatively different: in the first one, it represents the distance between fingers of the same hand, but in the second example, it represents the distance between the two hands and can covers all the surface width.

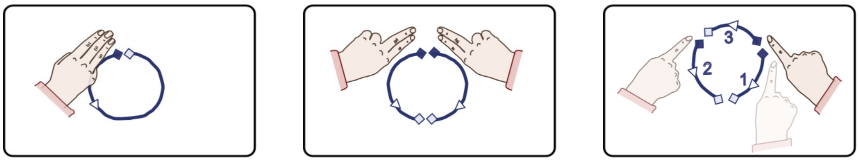
From our observations, we introduce the notion of "*atomic movement*" which reflects users' perception of the undividable role that a group of fingers is playing when performing a gesture. From our observations, users atomic movements are mostly in reference with the imaginary trail of a group of fingers which position is evolving in closely related movements. An atomic movement can have an internal state that can change depending on hands shape, fingers arity, velocity, direction, etc. However, state changes do not alter the role an atomic movement is playing in users' mind and its primary intention. Atomic movements are often mapped to global strokes in symbolic gestures, but they also capture more abstract movements implied globally by a whole set of fingers. In the particular case of users performing a symbolic gesture, users do not mind about the trail of each individual finger; instead they seem to view the atomic movement done by a whole group of fingers as an atomic or a global stroke without consideration to the individual stroke implied by every single finger. In the particular case of users performing more abstract multi-touch gestures, fingers' atomic movement seems to express a global meaning that users are attempting to convey and which seems to be related to what the movements are intended for. In all cases, the stroke or the trace of individual fingers considered separately are not an important issue from the user's atomic movement perspective, which is to contrast with a purely system perspective when dealing with multi-touch input. From our observations, we distinguish between two classes in participants movements depending on whether (i) the trail corresponding to fingers is stationary or (ii) it implies an embodied motion. As a practical examples, variable number of fingers, from one or two hands, moving together following the same path or being held stationary to delimit or point a region in the interactive surface, are among the most frequently observed atomic movements.

A set of atomic movements can be performed in the same time or in sequential:

- *Parallelism and Symmetry.* We observed that some participants articulated a gesture by performing synchronous parallel atomic movements. For example using two hands in the same time to draw a heart such that each hand is drawing a half of the heart. Those synchronous parallel articulations are generally accompanied with a certain symmetry. For instance, participants are observed combined the movements of their fingers simultaneously, in either a symmetric or an asymmetric style for some kind of gesture types. For the symmetry style, two kind of geometric symmetry were observed: *Axial* symmetry, which is the most recurrent type, occurs when the gesture involves two trails which are mutually mirrored according to an imaginary plane axis. This kind of symmetry applies recursively when more than two movements are used to construct the gesture, *e.g.*, to construct a square as the byproduct of two compound parallel symmetric movements, each movement being mapped in one different hand and subdivided into two atomic parallel movements performed symmetrically by the thumb and the index and/or the middle. *Central* symmetry is similar to the axial one with the exception that trails are mapped to movements going in opposite directions, *e.g.*, a circle with one finger moving up and one finger moving down, each one producing a half of a circle. On the other side, asymmetry in gestures occurs when participants were holding some fingers stationary upon the surface as a reference and simultaneously moving some others. For instance, users are observed to produce circled patterns by touching a region of the surface with one hand and simultaneously moving fingers from the other hand all around. From a physicality perspective, bimanual parallel movements are mostly attended with the use of the same fingers combination on each hand, while the use of one hand mostly engages the use of the index and the thumb. From these observations, we found that symmetry in users gestures can be described by the parallelism expressed by atomic movements.

- *Sequentiality.* We observed that some participants often operate in a sequential manner by iteratively posing and moving fingers on the surface, then releasing and posing fingers again at a new location binding the set of already performed movements. Users sequential movements imply more than a time pause or direction change. They are performed using one hand, as well as alternating different hands, and mixing parallel atomic movements with elementary atomic movements. In this class of interaction style, movements are mixed and matched both in time and in space according to users specific referential. This referential does not map perfectly with the system. For example, the boundaries of strokes induced by the atomic multi-finger movements are never perfectly matching with one another, though we think that participants intended to do so in their minds.

From this analysis, user gestures can be modeled using atomic movements, possibly combined along with parallelism and sequentiality. Figure 3.3 provides a simple situation illustrating this with three gestures produced by different participants.



(a) Single atomic movement (b) Two Parallel movements (c) Three Sequential movements

Figure 3.3: Users gestures categories

3.2.3 An Embodied Taxonomy of Multi-Touch Gesture

To capture the space in which our participants were conceiving and producing gestures, we propose the multi-level layered taxonomy summarized in Table 3.2. It is worth noticing that the multiple level of our taxonomy do not model separable attributes to be characterized individually. Instead, they represent the different aspects of a single unified dynamic mechanism ruling users in the achievement of a multi-touch gesture.

SEMANTIC-CONCEPT		
Mental meaning, Users' thoughts		
⇳		
PHYSICALITIES		
Enabling Motor Skills (*e.g.*, fingers, hands)		
Posture, Arity (*e.g.*, single, multiple, mixed)		
⇳		
MOVEMENT STRUCTURE		
A set of *Atomic* Movements		
Elementary (E)	Ref (R)	
	Motion (M)	
Compound (C)	Parallel (P)	P $:= \mathrm{P}_1 * \mathrm{P}_2$; $P_1, P_2 \in \{E, P\}$
	Sequential (S)	S $:= S_1 - S_2$; $S_1, S_2 \in \{E, C\}$

Table 3.2: A multi-level model for users' gestures: a taxonomy of Multi-touch gesture

At the high level of our taxonomy, we model the fact that a multi-touch gesture emerges from what user's mind is modeling before even touching the surface. In this respect, an external observer can only try to guess the *semantic concept* hidden in user's gesture, since it might be the case that the gesture it-self is not sufficient to fully reveal user's thought — which is in accordance with previous studies [Thieffry 1981, Huang 1995, Wobbrock 2009]. From a neurological perspective, hands and fingers are controlled and coordinated by human motor system at the aim of achieving a desired task. The *physicality* level thus captures the motor control allowing users to project the semantic level into the interactive surface. Finally, the *movement* level is the consequence of the motor goal expressed by hands and fingers motions in order to infer unitary blocks building the whole gesture.

The movement level is at the core of our model since it constitutes the interface between the user and the interactive surface/system. Consistent with our observations, we propose to structure this level according to two generic classes built in a recursive manner. At the low level of the recursion, we find the class of gestures formed with an elementary atomic movement. An elementary atomic movement can be either of type stationary (Ref) or Motion as discussed previously. The *Compound* class refers to the recursive composition of a set of atomic movements. It is expanded in two classes depending on the lifetime and the synchronicity of composing atomic movements. The *Parallel* classes refers to users making two or more different but synchronous parallel atomic movements. This class engages relative finger motions as well as two-handed symmetric and asymmetric interaction. The *Sequential* class refers to users performing a set of atomic movements, being possibly parallel or elementary, holding and releasing hands or fingers, on and from the surface, in a discrete iterative manner.

3.2.4 Users Variability

Our taxonomy is the result of an in-depth qualitative empirical synthesis of a wide range of gestures. We found that the three levels of our taxonomy contributes leveraging and unifying the high variety of users gestures. In fact, users gestural variations can be elicited as the result of the mental picture and time-space composition of atomic movements, as well as their physical mapping into users fingers and hands. At the semantic level, the global pattern induced by movements is the most apparent attribute that users where instantiating in different manners. However, gestures with similar global patterns can have different properties, *e.g.*, their composing atomic movements can be in different classes. At the physical level, variations in the number of fingers and hands are a natural outcome for most participants. The number of atomic movement is most often related to the number and type of fingers or hands engaged by users. This is especially the case when gesture global pattern contains a kind of symmetry. Finally, users variations can be captured at the movement level by eliciting the different possible time combinations of atomic movements (Motion and Ref) as well as their number which can vary from a gesture to another and from a user to another.

3.2.5 Comparison with Existing Taxonomies

Comparing to previous taxonomies, the semantic concept of our taxonomy relates to the Nature dimension defined by Wobbrock et al [Wobbrock 2009]. In that study, users were shown the effect of a gesture, then they was asked to issue the gesture. Hence, the Nature dimension is tightly related to the action of the gesture. In our first-task study, we did not ask participant to perform any precise action. Thus, the semantic concept level only reveals the meaning of the gesture without mapping it to the type of a particular action. On the other hand, physicality in our taxonomy relates to the Form dimension sketched in [Wobbrock 2009] and expanded by Freeman et al [Freeman 2009]. The registration, continuation and movement dimensions described within the Form dimension there-in did not result from a specific user-centric study, since the intention of Freeman et al work was primary focusing on teaching the user how to perform a gesture. Although, those dimensions provide a sound picture of *how* users may perform a gesture, we find that the physicality and movement levels of our user-centric taxonomy complements and refines in many aspects the empirical work of Freeman et al. For example, Freeman's distinguishes between two types of movements: *path* and *no path*, depending on whether the hand moves along a surface path or not. In our work, we explicitly distinguish between how users perform gestures (Physicality) and the notion of movements. In this respect, the Movements Level introduces a new dimension in users gestures and consistently renders the embodiment of gestural multi-touch interaction. The semantic concept behind users gestures can then be captured within an embodied and coherent flow engaging the cooperation of users fingers and hands to materialize the inter-relation between a set of unitary atomic blocks composing the gesture.

3.3 Task 2 Results: Articulating Gestures for Specific Symbolic Gesture Types

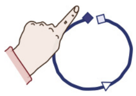
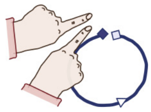
In this section, we discuss taxonometric breakdowns of the variety of gestures proposed by participants. 2 participants were excluded from the provided statistics since they made less than the four gestures required per gesture type in our experiment. 5 participants produced more than four different gestures for some gesture types. We constraint our analysis to only the four first ones. Overall, we have retained 28 x 8 x 4 = 896 gestures that are analyzed in the following.

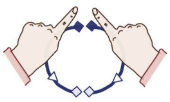
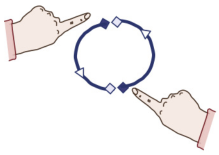
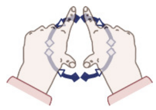
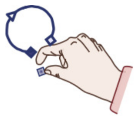
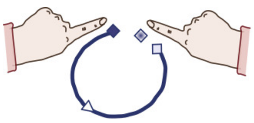
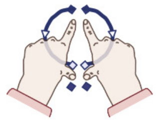
3.3.1 A Comprehensive Overview of Users Gestures

In our second-task experiment, participants were asked to produce symbolic gesture types in four different articulations. We were able to adequately classify all the gestures from this experiment task using our taxonomy. In addition to that, no pair of gestures associated to the same gesture type, and identified as different by one user, falls into the same class according to our taxonomy. As an example, Table 3.3 shows the set of gestures that are articulated by our participants for the circle gesture type. Although Table. 3.3 shows gestures relative to the circle gesture type, the depicted fingers and hand poses, as well as the induced atomic movements fairly holds for the other gesture types. Notice that the compound movement class is further expanded to provide a fine-grain view of users strategies.

3.3.2 Variability Breakdowns: Quantitative Results

In the following, we provide a detailed quantitative analysis of the different strategies adopted by users to draw the experimented symbolic gesture types, at the aim of gaining an accurate and insightful understanding of the priority and preference of these strategies with respect to users.

(E) ELEMENTARY ATOMIC MOVEMENTS	
1H; 1F; 1F M 	2H; 1F; 2F M 

(P) PARALLEL ATOMIC MOVEMENTS			
1H; 2F; 1F M∗M 	2H; 1F; 1F M∗M 	2H; 1F; 1F M∗M 	2H; 2F; 1F M∗M 
2H; 2F; 1F (M∗M)*(M∗M) 	1H; 2F; 1F R∗M 	2H; 1F; 1F R∗M 	2H; 2F; 1F (R∗M)∗(R∗M) 

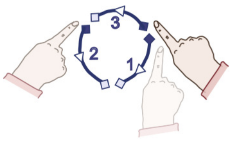
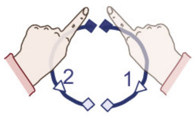
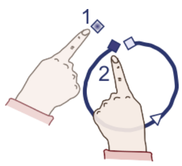
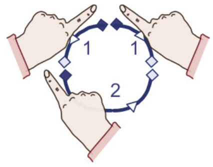
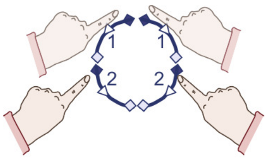
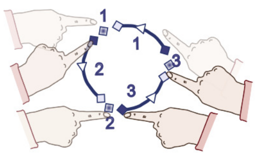
(S) SEQUENTIAL ATOMIC MOVEMENTS			
1H; 1F; 1F M-M-M 	2H; 1F; 1F M-M 	1H; 1F; 1F R-M 	2H-1H; 1F; 1F (M*M)-M 
1H; 2F; 1F (M∗M)-(M∗M) 	2H; 1F; 1F (M∗M)-(M∗M) 	2H; 1F; 1F (R∗M)-(R∗M)-(R∗M) 	

Table 3.3: A representative set of gesture articulations for the circle gesture type. We show respectively the number of hands used; the number of fingers per hand; and the number of fingers per movement (*e.g.*, 2H; 1F; 1F; reads as: two hands, one finger per hand and one finger per movement). The atomic movements (R: Ref or M: Motion) and their time composition is also explicated.

3.3.2.1 Hands and Fingers Inter-Dependency

Figure 3.4 shows the ratio (averaged over all users) of gestures performed with one and two hands for each gesture type and overall. We also incorporate the amount of fingers

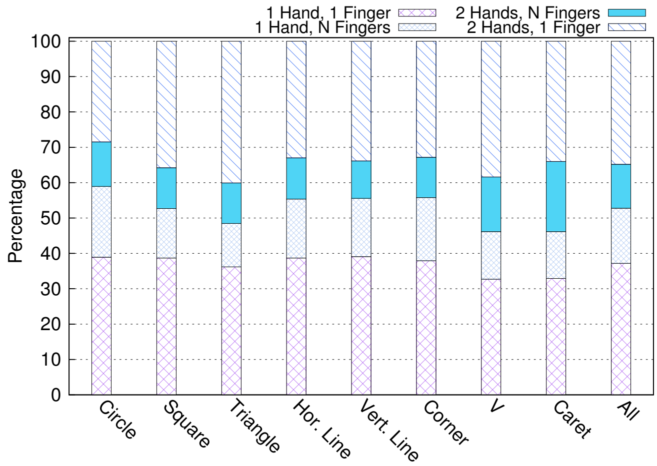


Figure 3.4: Hands and fingers (per hand) ratio

(single or multiple) engaged per each single hand. If the gesture is movement-compound, we count it multi-finger if at least one hand was engaged with more than one finger. We can see that users-gestures is fairly distributed over one hand (52.77%) and two hands (47.22%). Although, participants used more often a single finger per hand (78.17%), a significant ratio of gestures where multiple fingers are used per hand can still be reported (21.83%). A Friedman test revealed that gesture type does not have a significant effect on the ratio of two handed gestures performed by users.

3.3.2.2 Hands and Movement Classes Inter-Dependency

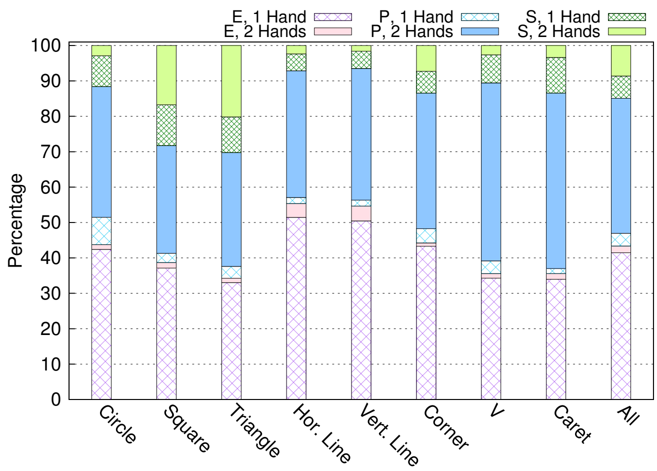


Figure 3.5: Movement and hands ratio (A: Atomic, P: Parallel, S: Sequential).

Figure 3.5 shows gestures ratios classified by movement categories, where we further distinguish between one-handed and two-handed gestures. A Friedman test revealed a significant effect of gesture type on the ratio of movement classes (elementary Atomic:

$\chi^2(7) = 32.55$, $p<.001$; Parallel: $\chi^2(7) = 21.98$, $p<.001$; Sequential: $\chi^2(7) = 50.12$, $p<.001$). In the elementary Atomic class, a post-hoc test using Wilcoxon test showed the significant differences of the couple of gesture types (vertical line, horizontal line) and the other gesture types. We attribute this to the fact that this couple of gesture types do not imply direction change in fingers movements so that elementary atomic movements are the more natural to conceive for users. In the Parallel class, significant differences were found between the couple of gesture types (V, Caret) and the other gesture types. Actually, the ratio of parallel two-handed gestures performed for these two gesture types is higher compared to the other gesture types. This can be explained by the fact that these gesture types can be more easily mapped into users two hands. In the Sequential class, significant differences were found between the couple of gesture types (square, triangle) and the other gesture types. These two gesture types are in fact clearly different from the others by the number of stroke combinations that can be used to perform them. Overall, we can see that users produced elementary atomic and compound parallel gestures in approximately the same proportion (reps. 43.34% and 41.76%), while the compound sequential class is represented in a relatively non-negligible ratio of 14.89%. A Chi-square test with Yates' continuity correction revealed that the percentage of two-hand and one-hand gestures significantly differed by movement class ($\chi^2(2) = 523.34$, $p<.001$). We can in fact remark the high correlation between two-handed gestures (reps. one-handed) and the parallel movement class (*resp.*, elementary).

3.3.2.3 Finger Combination and Movement Classes Inter-dependency

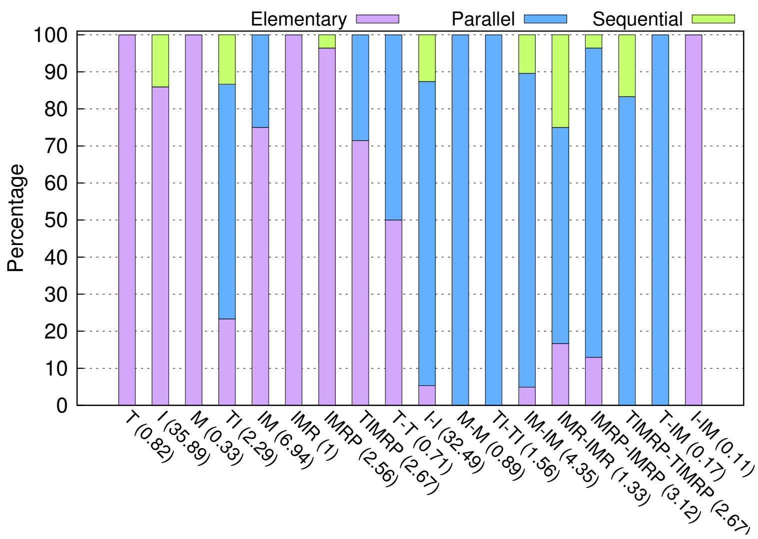


Figure 3.6: Movement class ratio according to hands-fingers combinations. Letters, T, I, M, R and P, in the x-axis labels, denote respectively the thumb, index, middle, ring, and pink. The sign '-' distinguish between left and right hand fingers. Numbers in braces refer to the ratio of the corresponding combination over all users.

Figure 3.6 shows the different combinations of hands fingers and their mapping to the movement classes. Overall we observed 18 possible one-handed and two-handed finger combinations. The parallel movement class is represented in 12 combinations among them only the index-thumb and index-middle is one handed. The other parallel two-handed combinations show a high similarity in the type of fingers used per hand. We can notice the absence of gestures engaging the pink or the ring in an elementary atomic movement. Whenever these two fingers are used, they appear in combination with the middle, index, and/or thumb fingers, by inducing the same multi-finger atomic movement. These combinations reflect the natural (comfortable) motor capabilities of users as well as the affordance of hand movements and their dependencies – which is consistent with previous studies on the mechanical/neurological relationship between fingers and their kinematics, *e.g.*, [Hager-Ross 2000a, Lin 2000, Schieber 2004].

3.3.2.4 Users' Transition-Frequency Automatons

While analyzing our data, we remarked that gestures' properties were not random over the four trials allowed per gesture type. In order to capture users thoughts and priorities in conceiving the different manners of producing a gesture, we find it interesting to model the evolution of gestures properties over time using probabilistic automatons [Segala 1995] mapping gestures properties into states and users variations into transitions. Figure 3.7 shows four such automata in a comprehensive informal manner. Notice that the first three automata are user-centric while the fourth one provides a more system-centric perspective as it will be discussed in the following.

Every initial state of the automata depicted in Figure 3.7 refers to participants starting the experiment. Columns refer to subsequent gestures produced by participants. Every row refers to a gesture propriety. The rows of the first automaton classify gestures according to whether they are one-handed or two-handed. Those in the second automaton classify gestures according to the movement class. The third automaton distinguishes between gestures where *every composing* atomic movement is single-finger and those where at least one atomic movement is multi-finger. Finally, the forth automaton classifies gestures depending on whether exactly one touch is involved throughout the whole gesture, or multiple touches are involved. The main difference with the third automaton is that touches are viewed relative to the system and not to users. The numbers in each cell then is computed as the average ratio over all users of gestures found in the corresponding state. This provides gestures distribution over time and can be interpreted as the empirical probability of user's gesture property mapping the corresponding state. Similarly, transitions depicted by labeled rows show the average ratio of participants moving from a state to another, which can be interpreted as the empirical conditional probability of falling in the subsequent gesture type knowing the type of the present gesture. For example, the initial state of the first automaton reads as .74 of participants perform the first gesture with one hand, or alternatively as users perform a two-handed gesture first with probability .26. Given that a user performs the first gesture with one hand, there is a probability of .45 that the outcome of the second gesture is two-handed. Notice that the cells in each column sum to one, which provides the empirical probability distribution (and thus the average ratio) of corresponding gesture types.

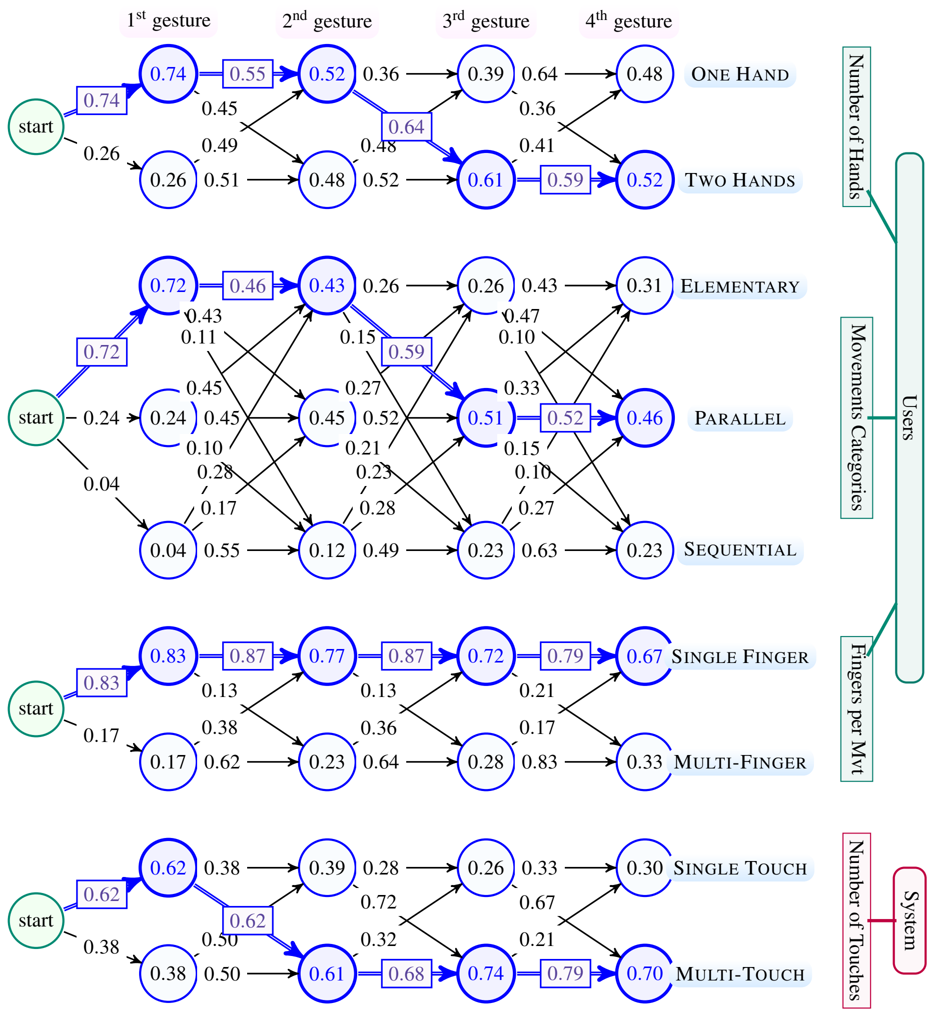


Figure 3.7: Gestures transition-frequency automatons (average over all gesture types and users) according to (from top to bottom): *Number of Hands*, *Movement categories*, *Number of fingers per atomic movement*, and *Number of touches*.

Figure 3.7 provides a time-dependent information about gesture type frequencies and variations. For example, as one can see in bold lines referring to the most likely gesture transitions, users start more likely with one-handed, elementary-atomic-movement, single-fingered, single-touch gestures. Then-after, they are more balanced in their choices consistently switching to two-handed and parallel movements. The empirical probability that users start with two-handed or parallel-movement gestures is relatively significant (.26 and .24) and grows sharply as users advanced in the experiment. This is to contrast with fingers usage since a single finger per movement is most often used all along the produced gestures. We interpret this as bimanual usage and movement variations being the most significant features ruling users' mind in performing the different set of gestures. Although the sequential-movement strategy is unlikely as a starting strategy (.04), it is interesting to remark that users falling in this state are more likely to produce the same type of movements in subsequent gestures. We can interpret this as the sequential mode offering more degree of freedom in producing different gestures by consistently playing with hands movement combination. Finally, we remark that from the system perspective, gestures involving multiple simultaneous touches on the surface are significantly represented all along the experiment mostly because users are either engaging their two hands or performing parallel movements.

3.4 Implications for Gesture-Based Application Design

In this section, we discuss the implications of our results for gesture design, surface technology, and user interfaces. In particular, we address points that seem important in order to design application that authorize several gestures for one command.

3.4.1 Movement Matter more than Posture

Our user study demonstrates that the movement induced by fingers motion matters for participants more than hand posture. Overall our participants, only two participants performed static gestures where fingers or hands were maintained stationary (Ref). Only in this case, the motor skills (blob type, posture, arity) used to structure the gesture are important, while their movements are not. Static gestures (set of Ref atomic movements), where hands/fingers posture is crucial, go beyond available classical multi-touch surfaces and need further sensing and input processing technology. In contrast, gestures where movements on the surface are crucial, are more accurate to the available knowledge and expertise on processing multi-touch input. Therefore it is our opinion that movement-based gestures provide more space to fully take advantage of nowadays multi-touch technology, so that their study and understanding should be the priority in the short term. However, current trends in augmenting surface computing technologies with new sensing facilities are also compatible with our user study. Advances in these directions would allow to enrich multi-touch surface input vocabulary so that gestures embodiment and versatility can be better encapsulated and exploited within gesture interfaces.

3.4.2 Interaction Gesture is a Multi-Level, Multi-View Input

Our study reveals that the variations of users gestures for the same command can be structured and classified by the specific properties of a set of atomic movements. Although the notion of atomic movement and the role it plays in our gesture taxonomy constitutes a low level abstraction of what users them-selves are modeling, it should as well serve for designers as a basic tool in the process of thinking, formalizing and setting up multi-touch gestural interaction techniques. Designing for multi-touch gestures as multi-movement entities embodied in users thoughts would then push a step towards shortening the gap between designers vision and the way multi-touch gestures are perceived and produced by end-users. In particular, an atomic movement is by definition not sensitive to the number of fingers or the number of hands being used, so that it enables to unify and to leverage previous studies recommending to not distinguish gestures by number of fingers, *e.g.*, [Wobbrock 2009].

Thinking about multi-movement multi-touch gestures, one have to keep in mind that the interdependency between the set of atomic movements forming a multi-touch gesture highly depends on users motor control over time and over space. Two main alternatives can be elicited depending on wether one hand or two hands are considered. In both cases, it is more likely that the movements occurs in parallel that is simultaneously in time. In addition, users are more likely to engage a single finger in the performance of one elementary atomic movement, thought this should not be serve as *the* rule.

Besides allowing to provide guidelines for the design of multi-touch gestures, the atomic movement perspective allows to expand in a comprehensive, yet precise, manner the space of possible mappings between a command and users gestures. By investigating the different possible combinations at the movement level, a variety of single and multi-finger, single and two-hand gestures can be supported, which can inherently: (i) improve flexibility, (ii) not penalize users by offering adequate response, and (iii) make sure that the variety of users choices leads to a gratifying interactive experience.

3.4.3 Gesture Recognition Needs to be Deeply Rethought

From the aspect of system feasibility, our study raises new challenges for the generic encoding and the reliable recognition of multi-movement multi-touch gestures. In fact, a formal and rigorous system-computable definition to what is an atomic movement is first needed. From the quality of such a definition depends the design of system embedded programs that determines a faithful representation of users' atomic movements and enable a consistent processing and interpretation of users gestures. One promising research path is to augment existing multi-touch frameworks based on formal grammars (such as proton++ [Kin 2012a] and others [Lü 2012, Kammer 2010, Spano 2012]) with both (i) declarative language elements that capture the notion of touch closeness in an elementary atomic movement, as well as with (ii) new compositional operators that render the time and space relation of atomic movements. The goal would be to automatically encompass users mental model features like: the independence of movements from the number of fingers, the possible variations in the combination of movements etc, within such formal frameworks. For patterned shape gestures, the challenge is more on the extraction of the different strokes implied by users movements. For example, it is not clear how recognizers in the $-family [Vatavu 2012c] can handle the fact that a stroke could be constructed by users using a variable number of fingers. Recently, Jiang and al. [Jiang 2012] proposed an algorithm to extract a single stroke from the different trajectories of multiple fingers on the surface. However, this reduction is incompatible with the fact that multiple strokes can interleave in time, *e.g.*, drawing a circle or a square or triangle or V or caret using two symmetric parallel atomic movements will be recognized as a line. We argue that the state-of-the-art recognizers for multi-touch gestures have to be rethought to support usability and consistently take into account the variety of users gestures in issuing a command. One path can be to take advantage from the consistency of the notion of touch closeness with respect to every user global time-space referential when performing atomic movements.

3.5 Chapter Summary

The study presented in this chapter laid the first investigation step for understanding multi-touch gesture variability and bridging the gap between user and designer. It was found that, in general, a multi-touch gesture is composed of a set of atomic movements, that can be single or multiple and performed with synchronous parallel articulations or asynchronous sequential articulations. Synchronous parallel articulations are related to the use of both hands simultaneously, while asynchronous sequential articulation are more related to the use of one hand. The number of fingers forming this atomic movement can also be single or multiple dependably on the user.

This study also have helped establish that work is needed on the technical challenges of multi-touch recognition in order to provide natural interfaces that support user variability. But before going to this step, we want to better understand multi-touch unconstrained gestures. For instance, we argue that two main challenges should be seriously discussed: First, is there a link between the gesture type and the articulation type. And second, while it is nice to know the set of variants that user handle to articulate a gesture differently, what is the effect of each of this variant on the difficulty of articulating the gesture. In the two next chapters, we treated these two issues in more details and then after two techniques are derived from all these findings to support user variability in a robust style.

4

On the Properties of Multi-touch Gesture Variations

In the previous chapter, we described a general taxonomy allowing us to understand users' gestures from different complementary levels, and to derive implications of users' variability. Our taxonomy is the result of a user-centric study giving new insights into the different possible articulations of unconstraint multi-touch gestures. As with any user study, ours suggests new questions that need to be investigated more deeply. In particular, since our primary goal was to observe and analyze users' variability, we considered a small set of 8 gesture types in our second task and we explicitly fixed the number of variations that users had to produce for every gesture types to 4. While this is sufficiently sound to elicit the user basic behavior when faced to variability, it also gives rise to further questions that can be hardly answered from the data implied by our experimental design. More specifically, we want to address the following interrelated issues:

- How many variations a user would be able to propose when having full freedom in the number of variations? This is important to know since it gives a more faithful idea about users' intuitive representation of variations.
- What are the sources of variations that users consider to articulate differently the same gesture type within unconstrained multi-touch input? What are the major sources of variations and what are the minor sources of variations?
- What are the more representative classes of gesture articulations? The taxonomy presented in the previous chapter allows in fact to derive several articulation classes; but what are the classes that emerge more often when users are given complete freedom in the number of variations? What are the most popular classes and why?
- How gesture shape could effect the gesture articulation?
- So far, we mostly considered a user-centric perspective to characterize gesture variations. However, it might be the case that different gesture articulations have similar

properties when adopting a more system-oriented perspective. Hence, an interesting question to investigate is: Is there major differences between variation classes in terms of gesture geometry and gesture kinematic from a system point of view?

- Finally, what are the preferences expressed by users with respect to different sources of variations? In particular, do they prefer one articulation type among another one and why?

To answer these new questions, we build on our previous user study by conducting a new experiment with essentially four different points: (i) a larger set of gesture types including letters, mathematical symbol, and specific forms is considered, (ii) participants are kept free in the number of variations they want to produce, (iii) participant are asked to give their preferences on performed gestures. Besides allowing us to answer the previously mentioned questions, the results obtained for this new user study constitutes a more comprehensive and in-depth quantitive analysis of the variability of users gestures. Our new results include gesture classification and counting, gesture distribution and gesture properties, subjective responses, and qualitative feedback. In addition, our analysis is conducted in light of our previous taxonomy in order to illustrate and to confirm its predictive power when dealing with new people and new gesture types.

4.1 User Study

In this section, we provide a detailed description of our new user study. Notice that our experiment design is strongly guided by the research questions discussed previously in the introduction.

4.1.1 Participants

Sixteen participants volunteered to take part in our study. Five were female. Participants' ages varied between 22 and 35 years (mean age 27.5, SD=4.1 years, all right-handed). Half of participants were regular users of smart phones and tablet devices, and two were regular users of an interactive tabletop. Table 4.1 shows the self-reported expertise of participants in terms of working with touchscreen devices. Over all our participants, 2 of them did also participated in our first user study presented in Chapter 3. The two user-studies were conducted at 11 months apart.

	Smartphone	**Tablet**	**Tablet PC**	**Tabletop**
Never	6	6	13	13
Occasional	2	5	1	1
Regular	8	5	2	2

Table 4.1: Distribution of usage of touchscreen devices among the sixteen participants in our study.

4.1.2 Apparatus

Gestures were collected on a 32 inch (81.3 cm) multi-touch display 3M^{TM} (model C3266PW) that was set to work at a screen resolution of 1920 $\times$ 1080 pixels. The display supports detection of up to 40 simultaneous touch points. The display was positioned horizontally on a table and connected to a computer running Windows and our custom data collection software application. The interface of the experiment application showed a gesture creation area covering approximately the entire screen in which multi-touch gestures could be articulated. The name of the gesture symbol to articulate was displayed at the top of the screen. Three buttons were available to control each trial of the experiment: start, Save and Next. Before pressing start, participants could experiment multi-touch input with their fingers in the gesture creation area without anything being recorded. Once start was pressed, the Save button was activated enabling participants to save their most recently entered gesture. The application logged all the touch coordinates with associated timestamps and identification numbers. Once a gesture was saved, participants could proceed to the next trial in the experiment.

4.1.3 Procedure

The application asked participants to produce gestures with trials presented in a random order. Only the gesture name was presented to participants. Just for the case in which participants were not familiar with these symbols, paper sheets of each symbol were prepared in advance and presented on demand. However, in such cases, participants were instructed to produce gestures as they would normally do and not copy the picture from the paper sheet. This situation occurred only a few times for the last eight symbols of Figure 4.1. For each gesture type, participants were asked to create as many different articulation variations as they were able to, given the requirement that executions are realistic for practical scenarios, *i.e.*, easy to produce and reproduce later. To do so, for each gesture type, participants first brainstormed multiple distinct gestures and then select a final custom set of gestures articulation that consider good and realistic for that gesture type. We asked five repetitions for each proposed variation of each gesture type in order to dispose of a sufficient amount of training samples to assess the recognition performance of M&C. For each gesture type, participants were given as much time as they wished. To reduce bias [Oh 2013], no recognition feedback was provided to participants at this stage. Also, to prevent any visual content from influencing participants into how they articulate gestures, no visual feedback was provided other than showing light red circles under each finger to acknowledge surface contact. The study took around 1 hour to complete.

After completing the set, participants rated their satisfaction with it on a 7-point Likert scale (1 - negative, 7 - positive). We are also interested in analyzing users preferences regarding the variations in gesture articulation. Thus, participants were finally asked to fill a post-experiment questionnaire consisting of several Likert scale statements regarding their multi-touch input preferences in terms of number of fingers, hands, number of strokes, and preferred order of entering gesture strokes over time, such as in sequence or parallel. Participants are asked to rate their preference regarding the variations in gesture articulation by using four 7-point Likert scales: (1) "Do you prefer articulating gestures with one or multiple fingers?" (1 - single ↔ 7 - multiple (2) "Do you prefer articulating gestures with one or multiple strokes?" (1 = single ↔ 7= multiple); (3) "Do you prefer articulating gestures with parallel or sequential strokes?" (1 = parallel ↔ 7= sequential); and (4) "Do you prefer articulating gestures with one or two hands?" (1 = 1 hand ↔ 7= 2 hands).

The experiment was video recorded and participants encouraged to use the think-aloud protocol. After finishing the experiment, we explicitly ask our participants to describe their strategy in proposing different gesture articulation for the same gesture type. Before the experiment, participants were introduced to the interactive surface and explained that it accepts multiple fingers input. The information displayed on the touch screen was duplicated on a secondary display for the experimenter to observe the gesture articulation behavior of the participants and take notes.

4.1.4 Gesture set

We employed 22 gesture types in our user study (see Figure 4.1). The gesture set is based on those found in other interactive systems [Anthony 2012a, Anthony 2010, Vatavu 2011b, Wobbrock 2007]. Gestures were selected to be general enough so that participants could reproduce them without a visual representation and thus encourage unconstrained articulation behavior. For instance, the gesture set is chosen to be versatile in type including letters, geometric shapes (triangle, square, horizontal line, circle), symbols (five-point star, spiral, heart, zig-zag), and algebra (step-down, asterisk, null, infinite). The gesture set is also selected to be versatile in their structure including gestures that are inherently unistrokes (*e.g.*, S and horizontal line) or multi-strokes (*e.g.*, A, Asterisk) and other gesture structures that are likely to be user-depend (*e.g.*, D, and Five-Point-star). The gesture set contains also gestures with a symmetry (*e.g.*, V and Heart) and others that did not contain a symmetry (*e.g.*, spiral and zig-zag).

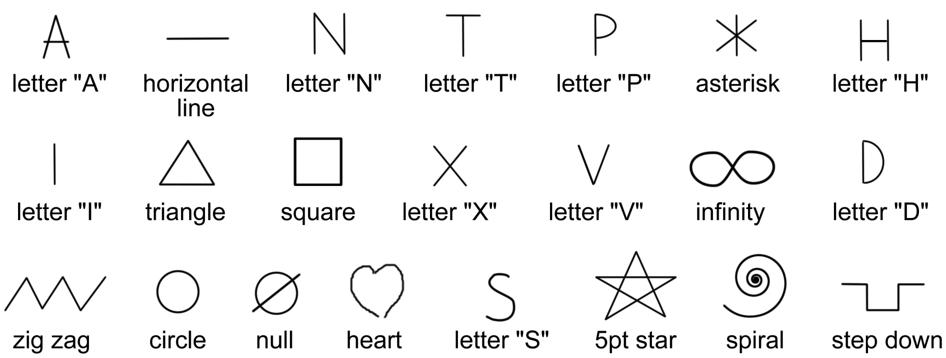


Figure 4.1: The set of 22 gestures used in our experiment.

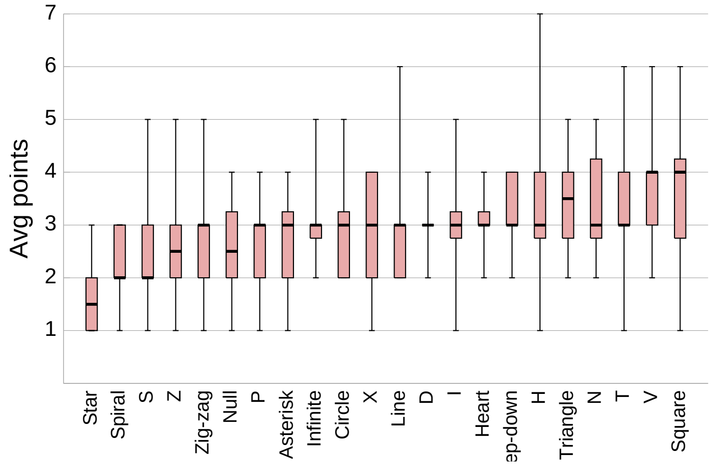


Figure 4.2: Number of gesture variations produced for each gesture type. NOTE: boxes show min, max, median, and first and third quartiles computed with data from all participants.

4.2 Counting and Classifying Variations

To start our analysis, we first count the number of variations that users were able to produce and elicit their different classes.

4.2.1 Number of Variations

Participants in our study were instructed to propose as many articulation variations as possible for each gesture type. As a result, we collected 5,155 (=1031×5) total samples for our set of 22 gesture types. In average, our participants proposed 2.92 variations per gesture type (SD=0.45), a result which we found to be in agreement with the findings of [Oh 2013] for action gestures (mean 3.1, SD=0.8). A Friedman test revealed a significant effect of gesture type on the number of variations ($\chi^2(21) = 84.41$, $p < .001$). The star and spiral gesture types presented the smallest number of variations (1.68 and 2.188 variations in average). The symbol with maximum amount of variations was square (3.56 in average) where participants were observed to easily decompose differently it into several individual strokes that were afterward combined in many ways in time and space. These very first results suggest that the specific geometry of each gesture shape presents users with different affordances of how to articulate that shape. Likely, the mental representation of a gesture variation implies a particular type of articulation which is tightly related to the gesture shape. We can also remark that for all gesture types except star and spiral there are from 4 to 7 articulations variations as a maximum of number of variations. This observation suggests that our choice of 4 variations for each gesture type in the Chapter 3 can be even bigger for some gestures types and some users. In the opposite, the minimum number of variations is between 1 and 2 variations. This result suggests that for some users and for some gesture type, the number of gesture articulation variations can be limited which can be related to the habitude of user but also to the shape geometry of the gesture type.

In the following, we are interested in eliciting the most representative gesture variations and the way they relate to gesture type and shape. In the next section, we classify the different collected variations. Later in this chapter, we shall analyze their distribution and frequency.

4.2.2 Variability of Multi-touch Gesture Input and Classification

Based on our taxonomy in the Chapter 3 and our new observations in the user study presented in this chapter – and additionally informed by previous research on users' preferences [Wobbrock 2009, Morris 2010, Hinrichs 2011] and studies in hand physiology [Jones 2006, Wing 1996], we can identify four major sources of variation and five minor sources of variation for multi-touch gesture input:

4.2.2.1 Major Sources of Variations.

In the following we describe the four major sources of variations that have been observed between and within users.

1. *Number of fingers touching the surface.* The number of fingers that users may decide to employ during gesture articulation represents the primary source of variation allowed by multi-touch sensing technology. Previous work showed that finger count can be exploited to design specific interactions [Bailly 2012], while other studies advised designers to be cautious when adopting finger-count gesture designs that could prove unintuitive because "users rarely care about the number of fingers they employ" and, therefore, "gestures should not be distinguished by number of fingers" [Wobbrock 2009, p. 1091]. In this work, we acknowledge that there is difference between touching the surface with different number of fingers but, in the light of previous work, we adopt a simplifying assumption and consider only two conditions for this source of variation, *i.e.*, ONE or MULTIPLE fingers employed by stroke/hand during gesture articulation. There are several reasons for this simplifying assumption. First, the hand anatomy restricts the independence of finger movements, such as the middle and ring fingers being the least independent [Häger-Ross 2000b]. Second, previous work reported that two classes of gestures should probably be considered by designers in terms of finger count [Wobbrock 2009], *e.g.*, "In general, it seemed that touches with 1-3 fingers were considered a single point, and 5-finger touches or touches with the whole palm were something more." (p. 1090), and "Four fingers should serve as a boundary between a few finger single-point touch and a whole-hand touch." (p. 1091). We therefore adopt the two-class point of view, but we distinctively consider the one-finger-touch condition because of its similarity with pen input with which people have years of writing experience and, consequently, its importance for touch-screen gesture interaction design [Tu 2012].

2. *Number of atomic strokes.* As highlighted in our taxonomy in Chapter 3, a gesture can be composed of a variable number of atomic movements. For symbolic gestures, which is specifically the case in this chapter, different number

of atomic movements can be fairly mapped into different number of atomic or global strokes when comparing to the classical terminology used in pen input. Previous work has found that variations in the number of strokes exist not only between users, but also between consecutive articulations of the same user [Anthony 2013b]. The literature on gesture recognition differentiates between unistroke [Li 2010, Kristensson 2004, Wobbrock 2007] and multi-stroke gesture recognizers [Anthony 2010, Anthony 2012b, Vatavu 2012c]. By following this distinction in the gesture recognition literature, we adopt a similar simplification approach as we did for the number of fingers, and consider two conditions for the number of strokes criterion of gesture variation, *i.e.*, ONE or MULTIPLE strokes. Note that according to our taxonomy single stroke condition corresponds to elementary atomic movement and multiple stroke corresponds to gestures performed with compound atomic movements.

3. *Single-handed and bimanual input.* Our second source of variation is the number of hands employed by users during articulation, which can lead to gesture strokes performed with single-handed input (*i.e.*, all strokes are articulated with one hand) or with bimanual input (*i.e.*, using always two hands over all gesture articulation) or a mix of single-handed and bimanual input (*i.e.*, using one hand at the same part of the gesture and two hands at other parts of the gesture). For example, bimanual multi-touch interaction is faster than single-touch single hand [Kin 2009]. We then adopt three conditions for the number of hands criterion of gesture variation, *i.e.*, ONE, or TWO or MIXED hands.

4. *Synchronicity.* Our last source of variation refers to hands/fingers movements synchronicity leading to gesture strokes performed sequentially asynchronous (*i.e.*, one stroke after the other, such as in drawing a plus sign with one finger) or parallel synchronous strokes (*i.e.*, more than one stroke are articulated at the same and over all the gesture articulation, *e.g.*, using two fingers to draw two sides of a heart at the same time) or a sequence of a mix of single stroke and parallel strokes (*i.e.*, some implied strokes of the gesture are articulated with one stroke at the same time and others are articulated in parallel. *e.g.*, using two fingers at the same time to draw the two diagonal symmetric lines of a triangle and then one finger to draw the horizontal line). For example, not all gestures can be conveniently parallelized in terms of articulation without detrimental effects on the interaction as we have already seen in Chapter 3. We then adopt three conditions for the synchronicity criterion of gesture variation, *i.e.*, sequential SEQ, synchronous SYNC or MIXED strokes.

Combining these four sources of variations theoretically enables the access to up to 36 (=2x3x2x3) variation combinations. However, not all combination are possible to be produced. For example, if the gesture is articulated with only one atomic stroke, then only sequential strokes articulation are possible to produce. Furthermore, if this key stroke is performed by a single finger, then only one hand input can be used (*i.e.*, by combining these two limitations 14 variations can not exist). Similarly, if the strokes are articulated using single finger per stroke with two hands input, only multiple strokes articulated in parallel are possible to do (*i.e.*, 3 other variations can not exist). In addition, if the gesture engaged multi-finger per stroke with single-handed then for gestures drawn with multiple strokes, only sequential strokes can be performed due to hand anatomy, *e.g.*, it is not possible to engage multi-finger parallel strokes from fingers coming from the same hand (*i.e.*, 3 others variations can not exist). These limitations have been confirmed by our observations. Thus, by considering these limitations only 16 possible combinations are still possible. Table 4.2 lists 16 multi-touch gesture classes resulted from logically mixing all conditions from our four sources of variation. Within the 16 listed classes, there are 3 classes that have not been observed during our data collection. The three gesture classes present gestures that contain at least one stroke that has been articulated using a mix of one hand and two hands, *e.g.* drawing a single stroke line using both hands then releasing one hand and continuing the same stroke. The remainder classes are all observed in our data collection at least once.

Results reported in the previous chapter reveled that for symbolic gestures, strokes synchronization is strongly correlated with the number of hands input, *i.e.*, the use of single handed gestures is strongly correlated to the articulation of asynchronous sequential movements and the use of both hands is strongly related to the articulation of synchronous parallel movements. Consequently, a gesture that contains a mixture of sequential and parallel movements is more likely to be performed with a mixture of single hand and two-handed input. This finding is again observed in our current study. Consequently, there are five classes that are rarely used (overall rate $< 1\%$). Thus, to not overload our quantitative analysis with numerical data, we choose to combine the sixteen listed classes into 8 super-classes according to number of fingers per stroke, number of strokes and synchronization (see Table. 4.2. In each super-class, we highlight the most popular classes as well as the 3 non observed classes). In Figure 4.3 we provide an illustration of various articulation patterns, each corresponds to one of the multi-touch gesture super-class, captured from our participants in our own experimental observations when asked participants to produce a square. Please note that these 8 super-classes are especially interesting for symbolic gestures, however, other classes combination can be used to group the 16 classes differently.

4.2.2.2 Minor Sources of Variations.

In the following we describe the five minor sources of variations:

1. *Stroke direction.* Every stroke has two directions *e.g.*, an horizontal line can be drawn from left to right but also from right to left.

2. *Strokes order.* The same gesture can be articulated using many strokes, and for the same strokes users may enter them with different ordering, *e.g.*, square shape may be articulated with many strokes, and these strokes may be entered with different ordering and directionality.

3. *Fingers combination.* The same gesture can be articulated using one or multiple fingers input, but for the same number of fingers, users can employ different fingers combination (*e.g.*, for single finger interaction Benko et al., [Benko 2009] assigned a color for each finger type. For multiple fingers, Marquardt et al., [Marquardt 2011]

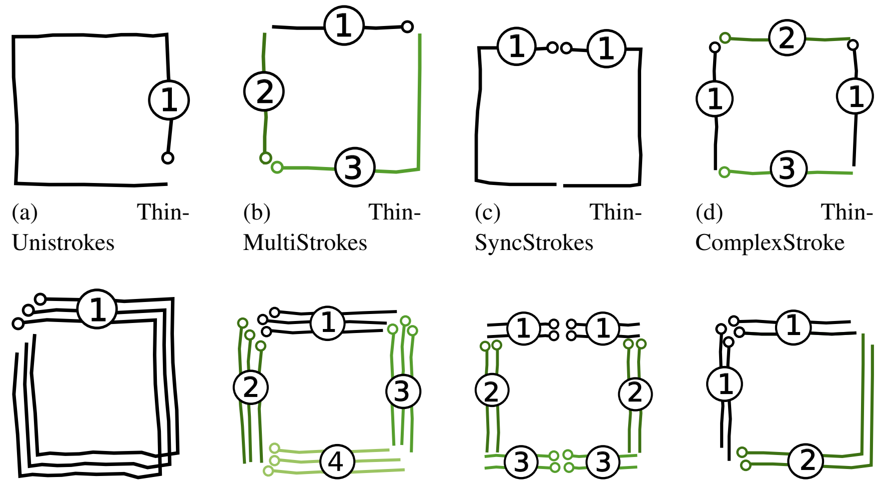
No.	Gesture class	Fingers Count per stroke	Fingers Count per hand	Hands Count	Stroke Count	Synchronicity
THIN-UNISTROKES						
1.	**THIN-SINGLEHADED-UNISTOKRES**	ONE	ONE	ONE	ONE	SEQ
THIN-MULTISTROKES						
2.	**THIN-SINGLEHANDEDMULTISTOKRES**	ONE	ONE	ONE	MULTIPLE	SEQ
THIN-SYNCSTROKES						
3.	THIN-SINGLEHANDED-SYNCSTROKES	ONE	MULTIPLE	ONE	MULTIPLE	SYNC
4.	**THIN-BIMANUAL-SYNCSTROKES**	ONE	ONE	TWO	MULTIPLE	SYNC
5.	THIN-MIXEDHADED-SYNCSTROKES	ONE	ONE	MIXED	MULTIPLE	SYNC
THIN-COMPLEXSTROKES						
6.	THIN-SINGLEHANDED-COMPLEXSTROKES	ONE	MULTIPLE	ONE	MULTIPLE	MIXED
7.	**THIN-MIXEDHANDED-COMPLEXSTROKES**	ONE	ONE	MIXED	MULTIPLE	MIXED
THICK-UNISTROKES						
8.	**THICK-SINGLEHANDED-UNISTOKRES**	MULTIPLE	MULTIPLE	ONE	ONE	SEQ
9.	THICK-BIMANUAL-UNISTOKRES	MULTIPLE	ONE or MULTIPLE	TWO	ONE	SEQ
10.	THICK-MIXEDHANDED-UNISTOKRES	MULTIPLE	MULTIPLE	MIXED	ONE	SEQ
THICK-MULTISTROKES						
11.	**THICK-SINGLEHANDED-MULTISTOKRES**	MULTIPLE	MULTIPLE	ONE	MULTIPLE	SEQ
12.	THICK-BIMANUAL-MULTISTOKRES	MULTIPLE	ONE or MULTIPLE	TWO	MULTIPLE	SEQ
13.	THICK-MIXEDHANDED-MULTISTOKRES	MULTIPLE	MULTIPLE	MIXED	MULTIPLE	SEQ
THICK-SYNCSTROKES						
14.	**THICK-BIMANUAL-SYNCSTOKRES**	MULTIPLE	MULTIPLE	TWO	MULTIPLE	SYNC
THICK-COMPLEXSTROKES						
15.	THICK-BIMANUAL-COMPLEXSTOKRES	MULTIPLE	MULTIPLE	TWO	MULTIPLE	MIXED
16.	**THICK-MIXEDHANDED-COMPLEXSTOKRES**	MULTIPLE	MULTIPLE	MIXED	MULTIPLE	MIXED

NOTE: The gesture classes that are the most observed in collected data are **highlighted**. Lines in gray represent the gesture classes that have not been observed. Please note that the later classes are not observed for symbolic gestures but can be used in others gesture types.

Table 4.2: The sixteen principal classes resulted from four sources of variation: single-handed, bimanual and mixed input, number of fingers, number of strokes and synchronization.

explored different fingers combination). For instance, and theoretically speaking there are 1023 finger combinations that can be explored on multi-touch surfaces. However, as we observed in previous chapter, the use of ring or ping fingers are always accompanied with the use of the index and middle fingers and the use of both hands in the same time is accompanied with the use of the same finger combination by hand. A result that we confirmed in this study with a more largest gesture set.

4. *Dominant and non-dominant hand input.* The gesture can be performed by the dominant hand as well as with the non-dominant hand. For example, the dominant right hand and the non-dominant left hand multi-touch interaction generate the same accuracy, but the dominant hand is faster than the non-dominant hand [Kin 2009].



(a) Thin-Unistrokes
(b) Thin-MultiStrokes
(c) Thin-SyncStrokes
(d) Thin-ComplexStroke
(e) Thick-Unistrokes
(f) Thick-MultiStrokes
(g) Thick-SyncStrokes
(h) Thick-ComplexStrokes

Figure 4.3: Various multi-touch articulation patterns for the square symbol produced with sequential strokes (a-b,e-f), synchronous strokes (c,g) and complex strokes (d-h), different number of fingers per stroke (e-h) and strokes (b-d,f-h). NOTE: Numbers on strokes indicate stroke ordering. The same number on top of multiple strokes indicates that all the strokes were produced at the same time by different fingers.

5. *Alternating hands.* In a multi-stroke gesture where each stroke is articulated with one hand, the user can alternate hands between stroke *e.g.*, using the right hand to draw the horizontal line of the letter T and then the left hand to write the vertical line. For instance, while alternating hands interaction is slower than bimanual simultaneous interaction, alternating hands interaction is as fast as the dominant hand interaction [Kin 2009].

In the remainder of this chapter we study and analyze in more details the proprieties of each of these 8 super-classes.

4.3 Counting and Understanding the Effect of Gesture Pattern on Gesture Variation

In this section, we first show the distribution of the eight super-classes for each gesture type and then we give a comprehensive analysis on how gesture variations relate to gesture patterns.

4.3.1 Counting Variations per Class

In Figure 4.4, we show the distribution of the eight super-classes for each gesture type. Overall, we found that 19.24% of all gesture articulations were performed using single-

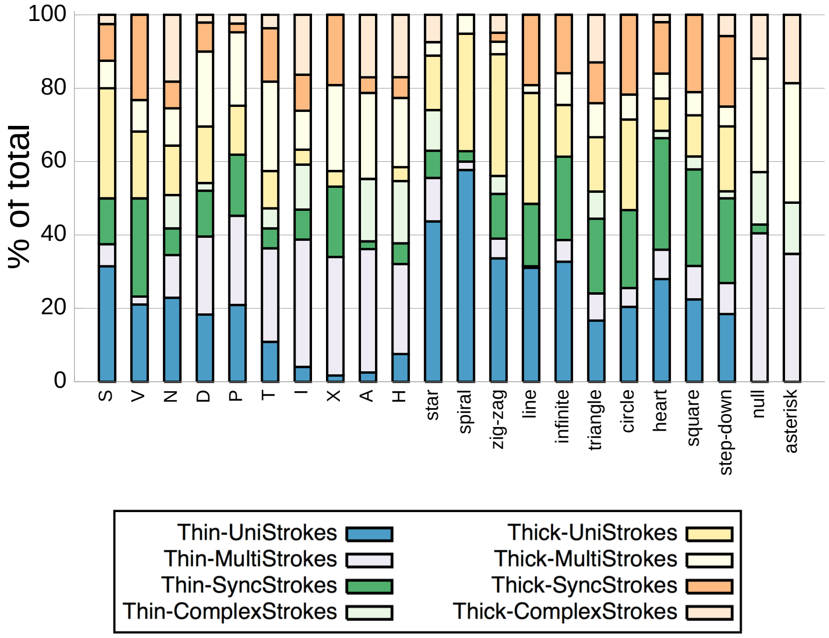


Figure 4.4: Distribution of gesture classes per gesture type.

touch and single stroke (the THIN-UNISTROKES class), 16.14% were single-touch multi-strokes sequential articulation (the THIN-MULTISTROKES class), 14.28% were single touch strokes performed in parallel (the THIN-SYNCSTROKES class), 5.53% were single touch strokes performed with compound articulation mixing synchronous and asynchronous strokes articulations (the THIN-COMPLEXSTROKES class),13.87% were multi-touch single stroke (the THICK-UNISTROKES class), 13.21% were multi-touch strokes performed in sequential one after the other (the THICK-MULTISTROKES class), 11.13% were multi-touch parallel strokes (the THICK-SYNCSTROKES class) and 6.60% were one and multitouch strokes performed with compound articulation (THICK-COMPLEXSTROKES class). Friedman tests reveal that gesture type have a significant effect on gesture count for the eight classes as showed in Table 4.3 (p<.001).

4.3.2 Understanding of the Effect of Gesture Pattern on the Gesture Articulation Variation

To study in more depth the effect of gesture type on gesture count for a given gesture class, we next run post-hoc Wilcoxon signed-runk tests with Bonferroni correction. This allowed us to cluster together the gesture types that presented significant similarity with respect to gesture count. In the following, we shall attempt to better illustrate how gesture types can impact users' variations, starting from the simplest single stroke class to the more sophisticated compound class.

No.	Gesture class	Overall Rate (%)	Friedman $\chi^2(21)$
1.	THIN-UNISTROKES	19.24	121.15**
2.	THIN-MULTISTROKES	16.14	118.84**
3.	THIN-SYNCSTROKES	14.28	87.26**
4.	THIN-COMPLEXSTROKES	5.53	66.66**
5.	THICK-UNISTROKES	13.87	116.21**
6.	THICK-MULTISTROKES	13.21	79.45**
7.	THICK-SYNCSTROKES	11.13	87.48**
8.	THICK-COMPLEXSTROKES	6.60	101.66**

Table 4.3: Overall rating of each gesture class and Friedman tests between gestures type for each gesture class are reported. NOTE: Friedman tests are all reported at $p<.001$ (**) significance level ($N=22$).

4.3.2.1 Clustering Single Stroke Gestures

In the THIN-UNISTROKES class, which corresponds to single touch single stroke articulations, we were essentially able to classify gestures in two distinguishable groups; $\text{group}_{11} = \{$S,V, N, star, zig-zag, line, infinity, triangle, circle, heart, square, setp-down, D and P$\}$ and $\text{group}_{12} = \{$I, T, X, A, null, asterisk$\}$. Inside each group, no significant difference between any pair of gesture types is found. However, a significant difference is found between every pair of gestures being respectively in the two groups ($p < .05$). We also notice that gestures in group_{11} were performed more often in the THIN-UNISTROKES class than gestures in group_{12}. Only the two gestures H and spiral were exceptions that were neither significantly different nor similar with all the gestures of the two groups in terms of gesture count. Nevertheless, the spiral was found to have more similarity with gestures in the first group, while the H was found to have more similarity with the second group.

When examining the structure of gestures forming the two groups, we can notice that gesture in group_{11} corresponds to gestures that can more naturally' be made with unistroke articulation while the group_{12} corresponds more to gestures that are more naturally performed with multi-strokes articulations. This explains the group elicited by our clustering and indicates that users variations is deeply impacted by the pattern shape of a gesture type.

In the THICK-UNISTROKES class, which differs from THIN-UNISTROKES by only the number of touches per stroke, we interestingly found that the groups elicited for the THIN-UNISTROKES class are still valid with the following exceptions. Gestures star, heart and zig-zag (resp., T) became significantly different from some gestures in group_{11} (resp., group_{12}) and not significantly different from some gestures of group_{12} (resp., group_{11}). This means that when changing the articulation of a single stroke gesture from single touch to multi-touch, the number of strokes used to perform the gesture stays essentially the same, and it is likely to be changed for only few gestures, especially those with a sophisticated geometric structure. We in fact argue that the impact of gesture type is not strong when examining the number of fingers used to articulate a gesture (compared to stroke count and synchronicity) as it will be confirmed in the following when examining the other classes of variations.

4.3.2.2 Clustering Sequential Multi-stroke Gestures

In the THIN-MULTISTROKES class, which corresponds to single touch multi-strokes articulations, we again found two groups of gestures that differ significantly by gesture count: $group_{21}$ ={S, V, star, spiral, zig-zag, line, infinity, triangle, circle, heart, square, step-down} and $group_{22}$ ={P, T, I, X, A, null, asterisk }. Inside each group, we found no significant difference between each pair of gesture types except the two gestures triangle and square which are both significantly different from the gesture type line in $group_{21}$. Moreover, gestures N and D are exception in the sense that they presented both similarity and differences with some gestures in every group (N has more similarity with gesture in $group_{21}$ while D has more similarity with $group_{22}$). Compared to the THIN-UNISTROKES class, we can remark that the elicited groups are rather similar with the exception of D, P and N. These exceptions can be explained by the fact that these gestures can be easily perceived and articulated with both single and multi-stroke gestures.

When considering the THICK-MULTISTROKES class, which differs from the THIN-MULTISTROKES by only the number of touches per stroke, we found that $group_{21}$ and $group_{22}$ remain the same with few gestures being no more represented in this class. More specifically, $group_{21}$ becomes composed of gestures {S, star, spiral, zig-zag, line, circle, heart, square} and $group_{22}$ becomes composed of gestures {T, X, A, null and asterisk}. As for the single stroke gestures, this observation confirms that gesture type does not have a strong influence on the number of fingers used to articulate the gesture.

4.3.2.3 Clustering Synchronous Multi-strokes Gestures

In this classes, strokes are articulated simultaneously and the analysis is more tricky. For the THIN-SYNCSTROKES class, we were basically able to elicit five distinguishable groups: $group_{31}$ = {asterisk, infinite}, $group_{32}$ ={null, spiral, A, star}, $group_{33}$ ={P, zig-zag, H, S, T, I, N,D}, $group_{34}$ ={V, heart, square} and $group_{35}$ ={X, line, triangle, circle, step-down}. Inside each group, we found no significant difference between every pair of gesture types. However, the difference between gestures in different groups is more complex to elicit. This is summarized in Table 4.4 giving an overview of how the gestures of each group differ in terms of gesture count. We can see that gestures in both $group_{32}$ and $group_{33}$ do not differ significantly. This is also the case for gesture in $group_{34}$ and $group_{35}$. Actually, when analyzing more carefully the structure of gesture type, we can observe that gestures in $group_{32}$ and $group_{33}$ do not contain a geometric symmetry, while all the gestures composing $group_{34}$ and $group_{35}$ do contain a clear symmetry in their geometric structure. This observation explains the differences of user variations in this class of gestures, for which users are more likely to synchronize both hands in parallel.

In the THICK-SYNCSTROKES class, which differs from the THIN-SYNCSTROKES by only the number of touches per stroke, two groups of gestures emerged from the groups elicited in the THIN-SYNCSTROKES class. The first one is composed by the union of $group_{31}$, $group_{32}$ and $group_{33} \setminus \{T,I\}$. The second one composed by the union of $group_{34}$ and $group_{35} \cup \{T,I\}$. A significant difference was found between gestures in these two groups, whereas no significant difference between gestures in the same group. Once again,

	$group_{31}$	$group_{32}$	$group_{33}$	$group_{34}$	$group_{35}$
$group_{31}$	+				
$group_{32}$	+	+			
$group_{33}$	*	+	+		
$group_{34}$	**	**	*	+	
$group_{35}$	**	*	+	+	+

Table 4.4: Post-hoc Wilcoxon signed-runk test for the THIN-SYNCSTROKES class. '+' means no significant difference was found between each pair of gesture types from composed by one gesture from the $group_i$ and one from the $group_j$. '**' (resp., '*') means that a significant difference was found between all pairs (resp., almost all pairs with few exceptions that we do not detail for clarity purposes) of gestures in the corresponding groups ($p < .05$).

this confirms that the number of fingers is not of primary importance for users' articulations.

4.3.2.4 Clustering Complex Gestures

In the THIN-COMPLEXSTROKES class, two main groups emerged: $group_{41}$ = {Spiral, S,V, line, P, infinite, circle, X} and $group_{42}$ = {star, T, triangle, A, H, I, N, null, asterisk} with the gestures in $group_{42}$ being performed more often in the THIN-COMPLEXSTROKES class. Again, we found a significant difference between gestures in the two groups, whereas we found no significant difference within the two groups. Gestures D, heart, square, zigzag and step-down were exceptions in the sense that none of them is significantly different from gestures of $groups_{41}$, but all of them are significantly different from gestures A and H in $group_{42}$. Thus, we can reasonably say that they are more likely to be classified in the $group_{41}$.

Interestingly, we can remark that $group_{42}$ corresponds to gestures that contain a subpart that present a straightforward geometric symmetry. Hence, this group contains the gestures that are more likely to be performed with an articulation variation that fall into the COMPLEXSTROKES class than in the SYNCSTROKES class. For instance, for letter A, users are able to draw the two diagonal lines in parallel articulation using two hands and then the horizontal line in sequential style using only one hand. In the opposite, $group_{41}$ contains gestures with either a clear symmetry or gesture with no symmetry even in some part of its structure. For these gesture types, users have a tendency to either produce a gesture using fully synchronous strokes (using two hands) or fully asynchronous strokes (using one hand) only. These observations show that users are able to decompose the geometric structure of sophisticated gestures in order to map it accordingly to a desired articulation.

Finally, and similarly to the other classes, the groups elicited in the THIN-COMPLEXSTROKES class remain the same when examining the THICK-COMPLEXSTROKES class with the exception of gestures star and T.

4.4 Analysis of Variation Classes Between and Within Classes

Gesture Type	Thin-UniStrokes	Thin-MultiStrokes	Thin-SyncStrokes	Thin-ComplexStrokes	Thick-UniStrokes	Thick-MultiStrokes	Thick-SyncStrokes	Thick-ComplexStrokes
Star	$56_{(22)}$	25	12	19	12	12	–	12
Spiral	$75_{(67)}$	06	06	–	$56_{(33)}$	06	–	–
S	$56_{(44)}$	12	$25_{(25)}$	–	$75_{(08)}$	12	$19_{(33)}$	06
A	12	$69_{(27)}$	06	$38_{(33)}$	–	$44_{(43)}$	12	$44_{(14)}$
Zig-zag	$56_{(56)}$	12	$25_{(25)}$	12	$69_{(18)}$	12	06	12
Null	–	$62_{(60)}$	06	$25_{(50)}$	–	$50_{(38)}$	–	$25_{(25)}$
P	$44_{(29)}$	$56_{(11)}$	$31_{(20)}$	–	$25_{(25)}$	$31_{(40)}$	06	06
Asterisk	–	$56_{(56)}$	–	$31_{(20)}$	–	$50_{(62)}$	–	$44_{(14)}$
Infinite	$62_{(50)}$	12	$50_{(25)}$	–	31	31	$31_{(40)}$	–
Circle	$50_{(12)}$	19	$44_{(43)}$	–	$56_{(33)}$	19	$50_{(25)}$	–
X	06	$62_{(40)}$	$31_{(60)}$	–	06	$50_{(38)}$	$44_{(29)}$	–
Line	$69_{(36)}$	–	$38_{(33)}$	–	$62_{(40)}$	06	$38_{(50)}$	–
D	$44_{(29)}$	$50_{(25)}$	$25_{(25)}$	06	$44_{(14)}$	$31_{(60)}$	$19_{(33)}$	06
I	12	$75_{(33)}$	25	$25_{(25)}$	12	$25_{(25)}$	31	$38_{(33)}$
Heart	$75_{(17)}$	$19_{(33)}$	$69_{(36)}$	06	$19_{(33)}$	19	44	06
step-down	$50_{(25)}$	$19_{(33)}$	$44_{(57)}$	06	38	25	$44_{(43)}$	$12_{(50)}$
H	25	$56_{(33)}$	$12_{(50)}$	$50_{(12)}$	12	$38_{(50)}$	19	56
Triangle	$50_{(12)}$	25	$44_{(43)}$	25	50	$25_{(25)}$	38	$38_{(17)}$
N	$56_{(33)}$	$38_{(17)}$	$19_{(33)}$	31	$44_{(14)}$	31	$19_{(33)}$	$44_{(43)}$
T	$25_{(50)}$	$62_{(30)}$	19	19	$31_{(20)}$	$62_{(20)}$	$44_{(14)}$	12
V	$56_{(33)}$	06	$56_{(67)}$	–	$56_{(11)}$	$25_{(25)}$	$50_{(50)}$	–
Square	$62_{(20)}$	$25_{(25)}$	$50_{(50)}$	12	44	19	$44_{(57)}$	–

Table 4.5: For each gesture type and for each articulation class, we report the rate of participants that have performed a gesture variation at least once in the corresponding class. Whenever there exist participants that performed more than one variation in the same class, we further report their rates in the number within the braces.

When counting and classifying variations, we remarked that users differ significantly by the ratio of gesture class. This was confirmed by a Friedman test conducted over our 16 participants ($\chi^2(16) > 150$, $p < .001$). This observation suggests that it is likely that some classes are more popular than the others when looking at a representative sample of users and independently of gesture type.

In order to faithfully render how many often users conceive a variation in a given class, we summarize in Table 4.5 the ratio of participants that have employed a given class to produce at least one variation per gesture type. We further report the ratio of participants that have produced more than one variation in the same class. This ratio is given within the braces in Table 4.5 and it is with respect to the total number of participants having employed that class at least once. For instance, for gesture S, Table 4.5 reads as follows: 56% (resp., 12%) of participants have drawn the gesture S using a single touch single stroke articulation (resp., single touch multi-stroke sequential). Among those participants, 44% (resp., 0%) have produced the gesture S with more than one variation in that class.

Three major conclusions can be drawn when analyzing the results in Table 4.5. This is discussed and supported with further statistics in the following.

Popularity of Gesture Classes. In Table 4.5, we can see that gesture classes are not represented among users in the same ratio. In fact, we can see that the single stroke and multi-stroke variations are the most represented among users. The parallel and compound classes are overall less represented, but they exhibits relatively high rates among users. We find that the single-touch and the multi-touch variations are rather equally represented.

These results can be explained by the influence of gesture type on gesture variations; but also by users' unconscious habitude in performing some particular classes of variations. In fact, we formulate the hypothesis that users have a tendency to first perform a gesture as they proceed in their daily lives. Then-after, they have a tendency to decompose the gesture in different parts and to perform those part either sequentially or in parallel whenever applicable. As a consequence, single-finger sequential gestures, which resemble pen gestures, makes people fall back on familiar patterns when producing them. This is not the case of parallel articulations, thus making the effect of gesture type more significant.

No. of Gesture Variation	THIN-UNISTROKES	THIN-MULTISTROKES	THIN-SYNCSTROKES	THIN-COMPLEXSTROKES	THICK-UNISTROKES	THICK-MULTISTROKES	THICK-SYNCSTROKES	THICK-COMPLEXSTROKES
First Gesture	**34.26**	**24.20**	2	2.54	15.90	15.39	3.63	2.04
Second Gesture	**16.39**	**14.19**	**14.68**	7.03	12.17	10.21	**14.68**	10.64
Third Gesture	7.13	10.00	**27.84**	7.92	13.65	**12.69**	12.77	7.92

Table 4.6: The distribution of gestures classes in the three first gestures. The two largest rates are **highlighted** for each gesture number.

We argue that this informal hypothesis can be supported by further quantitive results extracted from our experiment. In fact, in order to capture users thoughts in conceiving the different variation of producing a gesture, we extract the evolution of gestures classes over time, that is the ratio of each gesture class when examining only the first gesture performed by a participant, respectively, the second, and the third one. Because we found that overall, users provide 2.9 variations in average per gesture type, we report the distribution of classes for only the first three gestures. The obtained results are sown in Table 4.6. For each gesture number, we highlighted the two largest representatives classes.

When observing the distribution of classes over the three gesture, we found that user start more likely with sequential synchronization. Then-after they are more balanced in their choices between sequential and parallel synchronization then gestures with parallel synchronization are more represented. However, the rate that user start with an articulation that contains parallel synchronization is relatively important (11%) and grows sharply as users advanced in the experiment. These results support our hypothesis that users are more likely to start with familiar articulations which typically resemble to pen gestures and then to perform gestures with two-handed and parallel strokes synchronization.

Variations within the same class. The second interesting observation from Table 4.5 is that there exist different variations produced by some participants that fall within the same class. First, in each super-class there are hidden classes: the number of hands used to articulate the strokes can vary within each super-class. For gestures that contains multiple strokes (*i.e.*, MULTISTROKES, SYNCSTROKES and COMPLEXSTROKES), for the same class, users can vary the number of stroke(s) composing a gesture to propose a new articulation. For example drawing the triangle with 2 sequential strokes than with 3 sequential strokes. Similarly, when considering multi-finger gesture classes, users can vary the number of fingers composing a key stroke to propose a new articulation that will be classified in the same class, (*e.g.*, drawing the triangle with two-finger strokes, then drawing the triangle with three-finger strokes.). In addition, as we have mentioned before, there are 4 other minor sources of variations. Actually, when examining the different variations occurring in the same class, we remark that there are two minor sources of variations that accord more than the rest, namely, the stroke direction and the strokes order. For fingers combination, dominant and non-dominant hand and alternating hands, they are used only one time by one participant.

Pairwise affinity of gesture classes At this time of the analysis, one might naturally ask what are the classes of variations that are more likely to be performed by the same user. In other words, our eight classes capture well the variations that can be observed among different users; but, how well can they capture gesture variations for a single user? This question is much more difficult to answer since it is already known that even the agreement of different users on simply one gesture is low [Wobbrock 2009].

To study this question, we show in Table 4.7 the average ratio (over participants and over gesture types) of having two (or more) different variations in the two classes corresponding to each row and each column. Table 4.7 can be interpreted as the empirical probability of observing a user performing two variations in the considered pair of classes for a same gesture type. We highlighted in Table 4.7, the couple of classes that appeared most often together. Overall, we found that the two couple of gesture classes that appear most often together in gesture variations are: 1) THIN-UNISTROKES class with THIN-SYNCSTROKES class (21%) , and 2) THICK-MULTISTROKES class with THICK-SYNCSTROKES (15%). These results show that strokes synchronization (and, also, the number of hands) represents a good feature to handle variability of user.

	THIN-UNISTROKES	THIN-MULTISTROKES	THIN-SYNCSTROKES	THIN-COMPLEXSTROKES	THICK-UNISTROKES	THICK-MULTISTROKES	THICK-SYNCSTROKES	THICK-COMPLEXSTROKES
THIN-UNISTROKES	**14**							
THIN-MULTISTROKES	**13**	**10**						
THIN-SYNCSTROKES	**21**	**10**	**.11**					
THIN-COMPLEXSTROKES	5	**10**	2	2				
THICK-UNISTROKES	**10**	4	**7**	1	**5**			
THICK-MULTISTROKES	3	4	3	2	**11**	**8**		
THICK-SYNCSTROKES	**5**	3	4	2	**14**	**13**	**7**	
THICK-COMPLEXSTROKES	1	3	1	2	**5**	**10**	**5**	3

Table 4.7: The average ratio (over participants and over gesture types) of having two (or more) different variations in the two gesture classes corresponding to each row and each column. We **highlighted** the couple of gesture classes that appeared most often together

We also can remark that the pairs of class variations that are more likely to be performed by a same user cluster together according to the number of fingers used to perform the gesture. In fact, we remark that the probability that a user perform two different variations by moving from a single touch to a multi-touch gesture is relatively low (< 0.05) except for the two class pairs: THIN-UNISTROKES with THICK-UNISTROKES and THIN-UNISTROKES with THICK-SYNCSTROKES . In contrast, for a fixed number of fingers (either single or multiple), the probability that a user produces different gestures by varying the number of strokes or the synchronicity is relatively high. Notice also that the probability of producing two variations in the same class is also consistent with the results of the previous section. This probability stays relatively significant which indicates that for a single user varying strokes' numbers, hands number and strokes synchronization are the important source of variations. Number of fingers per stroke varies between the gestures of different users but it varies much less within the gestures of the same user. Hence, we can say that the features that are mostly used by a single user to produces different gesture variations are in a decreasing order of importance: synchronicity (parallel, sequential, complex) (and so number of hands input), number of strokes (one, two, etc), the number of touch input (single or multiple) and then the others minors sources of variation such as stroke direction and strokes order.

4.5 Geometric and Kinematic Characteristics of Gesture Classes

In the previous sections, we were able to elicit the properties of variations classes from a purely user-centric perspective. We believe it is also important to understand the features that could characterize those classes from a more multi-touch system perspective. Our motivation is to understand at what extent fundamental differences between two variations in different classes can be detected at the system level. For this purpose, we consider 4 representative gesture descriptors that we believe adequate to characterize multi-touch gesture articulations in terms of (1) gesture geometry and visual appearance, and (2) kinematics [Blagojevic 2010]. We characterize the geometry of a multi-touch gesture by its bounding box size, and the aspect ratio of the width of the bounding box divided by its height; gesture kinematics is characterized by its production time and its average speed. In Table 4.8, we give a detailed summary of all gesture descriptors with accompanying calculation formulas and explanations.

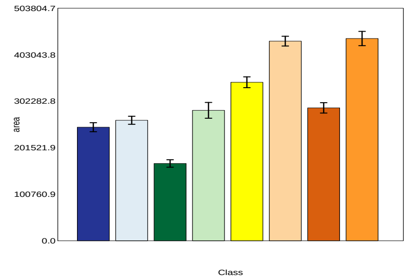
No.	Gesture descriptor	Definition	Units
A. Gesture geometry and visual appearance			
1	Size	Area of the gesture bounding box: $(max(x_i) - min(x_i)) \cdot (max(y_i) - min(y_i))$	pixels^2
2	Aspect ratio	Width of the bounding box divided by its height: $(max(x_i) - min(x_i)) / (max(y_i) - min(y_i))$	-
B. Gesture kinematics			
3	Production time	Total articulation time, $T = t_n - t_1$	ms
4	Average speed	average of the average speed of the touch points composing the gesture; $1/n \times \sum_{i=1}^{n} avg.speed_{touch_i}$; n is the number of touch strokes . The average speed of a touch point is equal to the path length of the touch stroke divided by its production time.	pixels/ms

Table 4.8: Gesture descriptors employed in this study. NOTE: A gesture is represented as a set of two-dimensional points (x_i, y_i) with associated timestamps t_i, $\{p_i = (x_i, y_i, t_i) \mid i = 1..n\}$.

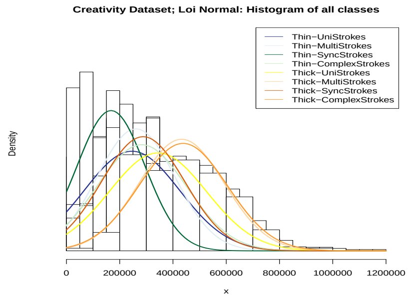
In Figure 4.5 we show the boxplots (with median and quartiles) as well as the frequency distribution obtained for every descriptor and every variation class when collecting data from all participants and all gesture types.

By mean of Kruskal-Wallis tests, we found that the gesture class has a significant effect on any considered gesture descriptor ($\chi^2(7)$=1187.32, p<.001 for gesture area, $\chi^2(7)$=134.43, p<.001 for the ratio, $\chi^2(7)$=771.55, p<.001 for the production time and $\chi^2(7)$=1336.88, p<.001 for the gesture velocity). Moreover, the differences between classes in terms of their descriptors is confirmed when studying the frequency distribution of each gesture class. More specifically, by running Kolmogorov-Smirnov tests [1] between each pair of classes, we were able to compare the pairwise distributions of descriptors in

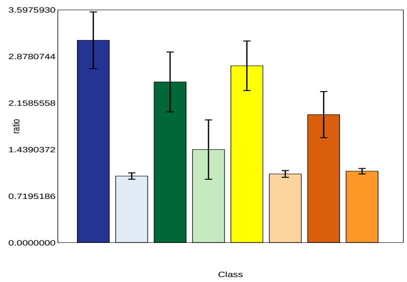
[1]Kolmogorov-Smirnov (K-S) is a non-parametric test of the equality of continuous, one-dimensional probability distributions that can be used to compare a sample with a reference probability distribution (one-sample K-S test), or to compare two samples (two-sample K-S test)



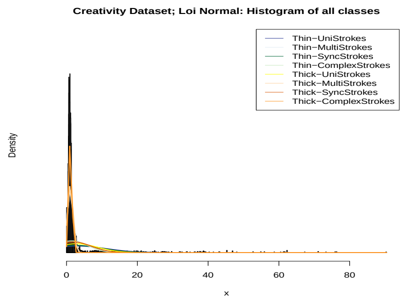
(a) Boxplots of the gesture area.



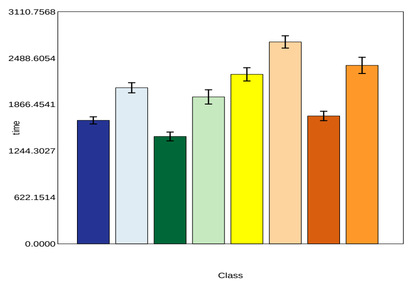
(b) Area frequency histogram with superimposed area normal models.



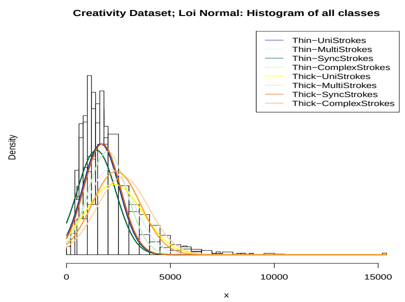
(c) Boxplots of the gesture ratio.



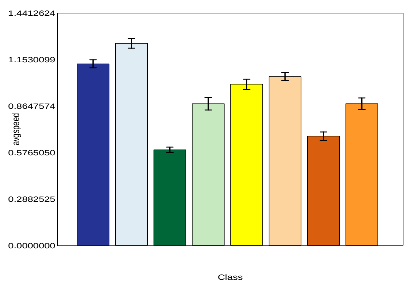
(d) Ratio frequency histogram with superimposed ratio normal models.



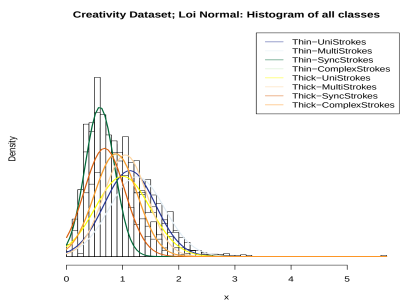
(e) Boxplots of the gesture time



(f) Time frequency histogram with superimposed time normal models.



(g) Boxplots of the gesture speed.



(h) Speed frequency histogram with superimposed speed normal models.

Figure 4.5: Boxplots (left) and frequency distribution (right) of the gesture geometric and kinematic descriptors for each gesture class.

different classes:

- When examining the distribution of the geometric descriptors we found that: (1) for the distribution of gesture area, there is a significant difference between all pairs of classes ($p<.05$) except for the two following pairs: (THIN-MULTISTROKES ; THIN-COMPLEXSTROKES) and (THICK-MULTISTROKES ; THICK-COMPLEXSTROKES); (2) for the distribution of gesture aspect ratio, there is a significant difference between all pairs of classes ($p<.05$) except for the two following pairs: (THIN-MULTISTROKES ; THIN-COMPLEXSTROKES); (THICK-MULTISTROKES ; THICK-COMPLEXSTROKES) and (THIN-MULTISTROKES ; THICK-MULTISTROKES). These results suggest that the area size of gestures articulated using a complex synchronization strokes (and so mixing one and two hands input) have the same distribution than for asynchronous sequential multiple strokes (and so one hand) gestures. This also suggests that the ratio of gestures articulated using a complex strokes synchronization (and so mixing one and two hands input) have the same distribution than for sequential multi-stroke (and so one hand) gestures.

- When examining the distribution of the kinematic descriptors, we also found that for the production of time descriptor a significant difference between the time distribution of all pairs of classes ($p<.05$) except the couple (THIN-MULTISTROKES ; THIN-COMPLEXSTROKES). This result strengthen the fact that articulating the gesture with complex strokes synchronization (and so a mix of one hand and two hands) is similar to articulating it with sequential strokes synchronization (and so only one hand) kinematically speaking. For the gesture velocity, we found a significant difference between the velocity distribution of all pairs of classes ($p<.05$) except the couple (THIN-COMPLEXSTROKES ; THICK-COMPLEXSTROKES) classes. This result suggests that the number of fingers do not affect the velocity of the complex gestures classes.

These results show that the structural and kinematic properties of multi-touch gestures can be deeply impacted due to users' variations. The source of variations that allowed us to elicit our classes, *i.e.*., number of fingers, number of strokes, and the strokes synchronicity over time (and so number of hands), are in fact found to imply different properties. We remark that one can attempt to compare the effect of these source of variations more throughly. For instance, one may ask what source have more effect than the other? Is the effect positively or negatively perceived by users (*e.g.*, if with a gesture variation users are unconsciously faster in performing the gesture, then how this impacts users' feeling about that variation, etc)? Actually, the answer to such questions is a challenging task, and the user study considered in this chapter does not allow us to address them adequately. In fact, not all gesture types presented the same number of variations nor all variations were made by all our participants. The first results obtained within this user-study shall be extended in the next chapter where the reader can find a more specific experiment allowing to deepen our understanding of variations implications both at the user level and at the multi-touch system level.

4.6 Mental Model and Qualitative Feedback

We accompanied our quantitative data with considerable qualitative data that capture users' mental models as they choose and articulate a gesture articulation variation.

4.6.1 Users' Preference

The primary goal of our user study is to understand users' unconstrained multi-touch gesture articulation behaviors and to analyze the features/degrees of freedom that users will consider to propose different variations for the same gesture type within multi-touch input. In this respect, it is important to learn about user preferences when producing a different variations. This was planned before running our experiment in the form of questionary that users had to perform after completing the required task. In fact, we preferred to ask participants about their preferences at the end of the experiment in order to not influence participants during the experiment and to avoid biases in how users freely articulate variations.

First, after completing the set of gestures, participants were asked to rate their satisfaction regarding their multi-touch input performance by using a 7-point Likert scales (1 - negative, 7 - positive). Overall, results showed that participants were satisfied with the set of gestures they proposed (median 6, stdev=.83). Three participants are fully satisfied and only one participant give a score of 4 miming that he could propose other gestures by varying the number of fingers.

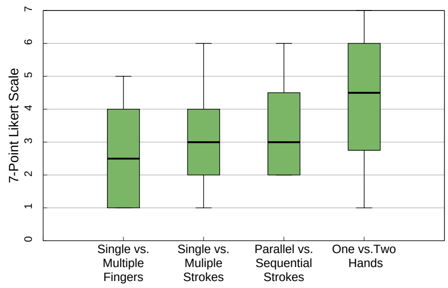


Figure 4.6: Users preferences for articulating multi-touch gestures.

We then asked participants to rate their preference regarding to the number of fingers, number of strokes strokes synchronization and one hand and bimanual input in gesture articulation. As depicted in Figure 4.6, results did not show any specific tendency in user preference. However, we can say that users preference are more likely to be single finger than multiple fingers, single stroke than multiple stokes, parallel movement as opposed to sequential movements and two-handed as opposed to one-handed. Interestingly, although two-handed parallel articulations were more represented in the second gesture performed by user rather than in their first gestures, our participants preferred it to one-handed sequential gestures. This observation suggests that people could develop different preferences with practice for articulating gestures in terms of strokes synchronization.

4.6.2 Qualitative Feedback

Along all our experiments, we also observed carefully the variations in how users articulate multi-touch gestures, and we recorded users' qualitative feedback. We highlight in this section such findings and, where applicable, discuss implications for future gesture input designs.

1. **Preference for multi-finger input.** Thirteen out of sixteen participants used more than one finger per stroke over all performed gestures, *i.e.*, only 2 participants never used multiple fingers. Some participants were enthusiastic to touch the surface with many fingers at once, and witnessed they "feel more free and comfortable when using many fingers", while one participant said he was "more comfortable with multiple fingers, since I feel like their movement is better controlled by my arm". Although multiple fingers were preferred, participants did not care about the exact number of fingers touching the surface. One participant witnessed "one or multiple fingers is the same and has no effect on the stroke nor on gesture expressiveness... I try to see how can I decompose the gesture into multiple strokes and use both hands simultaneously for different strokes". Also, it was often the case for some fingers to disconnect from the surface for a short period of time during gesture articulation (e.g., start drawing with three fingers, continue with two, finish with three fingers again). For such cases, an appropriate visual feedback might prove useful to show users what unintentionally happened during articulation.

2. **One finger is for precise input.** When participants employed one finger only, they explained that they did so to be more accurate. For example, one participant witnessed that "when the symbol is complicated, such as a five-point star or spiral, I prefer using one finger to be accurate". Three participants regularly used one finger to enter gestures. Two witnessed they conceptualized strokes simultaneously articulated by multiple fingers as being different, even though the movement was the same. Participants also made connections between single-finger gestures and pen input in many cases, e.g., "I use my finger like a pen". This finding may have implications for future finger gesture designs, as we already know that finger and pen gestures are similar but also different in many aspects [Tu 2012].

3. **More fingers means more magnitude.** Three participants felt they were drawing thicker strokes when employing more fingers. This finding may have implications on designing interaction techniques that exploit the number of fingers touching the surface beyond finger count menus [Bailly 2012].

4. **Drawing on multi-touch surface resemble to drawing on the sand.** One participant said "drawing gestures on a multi-touch surface is very natural and intuitive way as I feel that when I move my fingers on the surface to draw a symbol is similarly to move my fingers on the sand on the beach to draw a symbol especially for the 'heart' gesture type". She adds that multi-touch surface give her the freedom to use or not both hands, think that she can only do when she draw on the sand but not when using a pen. This feeling makes for her the interaction very suitable and more expressive.

5. **Symbol type influences multi-touch input.** Two participants said they articulated letters just like they wrote them with the pen, one stroke after the other. However, they felt more free and creative for the other symbols. One other participant commented that for the null gesture, she will made it just like she have learn at school: first drawing the circle and then putting a line through it to became a null. Another participant was enthusiastic to touch the surface with both hands at once said "I wish we had been taught to use both hands simultaneously to write letters! It is faster, more precise, and easier". Most participants considered that the number of strokes and their coordination in time depends on gesture type, which we verified as true (see Figure 4.4). One participant said that "if the symbol can be drawn with only one stroke, I prefer to perform it with one stroke only"; two other participants "whenever there is a symmetry in the symbol, I prefer multiple simultaneous strokes"; and another participant "whenever I can decompose the symbol on multiple stokes where I can use my both hands to perform strokes simultaneously, I will do it".

6. **Gesture position, rotation, size and speed can be a source of variation.** One participant said that the position of the gesture on the surface, its size; its rotation and the gesture velocity could be also a source of variation that he can used to propose more gesture articulation variations. However, he did not used for two main reasons: (1) he found that varying the number of hands and their synchronicity over time are more specific and 'intuitive' for multi-touch surface, (2) the velocity can may be difficult to distinguish without any feedback about his velocity from the surface.

4.7 Chapter Summary

The study presented in this chapter laid to a fine-grain understanding on multi-touch gestures variability. We have a more precise idea on how and where user will found the main sources of variation. We have also identified the important gesture classes. For each class, we have a better understanding on its characteristics. This is fundamental to adequately apprehend, identify and delimit the user behavior from a rigorous scientific analysis. This is also crucial in the context of proposing new tools that need the variability of user gestures; but also when exploring the space of new multi-touch interaction possibilities that are based on the natural ability of users to perceive and articulate a gesture in different ways.

However, a crucial aspect which is difficult to analysis from this study is whether these different sources of variations (mainly the number of fingers, number of strokes and strokes synchronization (and so number of hands)) induce an additional degree of articulation difficulty or if they are equivalent. Is this related to the type of variation source or to the gesture type itself? In what case a gesture articulation is more efficient than another? In the next chapter we shall precisely address these questions and focus on the effect of each main source of variation on the gesture articulation difficulty.

"A man of character finds a special attractiveness in difficulty, since it is only by coming to grips with difficulty that he can realize his potentialities."

Général de Gaulle – French President (1890 – 1970)

5 On the Perceived Difficulty of Multi-Touch Gestures

Designing gesture sets can prove a challenging task, because of the many factors involved during gesture articulation and interaction. The current practice and literature of designing gesture interfaces have identified several of these factors, according to which gesture commands should be unambiguously recognized [Long 1999], fast to articulate as shortcuts [Appert 2009], ergonomically easy to execute [Morris 2010, Wobbrock 2009], easy to learn and recall [Appert 2009, Nacenta 2013, Nielsen 2004a], and have a good fit-to-function in the application [Nielsen 2004a, Wobbrock 2009]. Admittedly, and as shown in previous chapters multi-touch gestures present considerably more degrees of freedom which they afford during articulation, such as the possibilities to employ different number of fingers, strokes, and single or bimanual input, which result in variability in multi-touch gesture articulation. Consequently, the design of intuitive multi-touch gesture commands proves a considerably more difficult task, which make designers to ultimately recur to users in participatory design studies [Buchanan 2013, Morris 2010, Nacenta 2013, Wobbrock 2009].

We argue that *articulation difficulty* is an important notion for gesture set design that encompasses multiple factors, such as *ergonomic difficulty* of the physical, motor articulation of the gesture path, but also *cognitive difficulty* required for learning and recalling the geometry of the gesture shape and, conceivably, the specifics of its articulation (e.g., number of fingers to employ, number of strokes, or stroke ordering). Gestures that are perceived by users as difficult to articulate, in any of the acceptations above, may lead to a negative user experience and low adoption of the gesture interface.

Although the notion of articulation difficulty has been mentioned as design criteria in several works [Morris 2010, Nielsen 2004a, Wobbrock 2009], it has only been thoroughly examined for unistrokes [Vatavu 2011b]. However, multi-stroke bimanual multi-finger gestures have been left unexplored so far.

Thus, the research questions addressed by this chapter include:

- Is a gesture articulated with multiple fingers perceived more difficult than in the case where one or less fingers are used?

- How the number of fingers affects the gesture structure, the gesture geometry and its visual appearance and the gesture kinematics?
- Is a gesture articulated with multiple strokes perceived more difficult than in the case where one or less strokes are used?
- How the number of strokes affects the gesture structure, the gesture geometry and its visual appearance and the gesture kinematics?
- Is a gesture articulated with two hands and synchronous parallel articulation perceived more difficult than in the case where one hand and asynchronous sequential articulations are used?
- How the number of hands and the articulation synchronicity affect the gesture structure, the gesture geometry and its visual appearance and the gesture kinematics?

For that purpose, we investigate in this chapter the difficulty of articulating multi-touch gestures, by examining the effect of multi-touch degrees of freedom on the subjective user-perceived difficulty of the gesture articulation task. To this end, we conducted a user study that extends the results of Vatavu et al. [Vatavu 2011b] that were only interested in the perceived difficulty of single-touch unistrokes. We then introduce a new variant of computing multi-touch gesture descriptors and report correlation results between subjectively-perceived articulation difficulty and objectively-computed gesture descriptors.

5.1 User Study

We rely on the methodology of Vatavu et al. [Vatavu 2011b] to collect users' self-reported estimations of perceived articulation difficulty for multi-touch gestures. We apply the methodology for a series of three experimental tasks designed to collect multi-touch gestures articulated under various conditions in terms of (1) number of fingers, (2) number of gesture strokes, and (3) single and bimanual articulation.

5.1.1 Participants

Eighteen participants (four females) volunteered to take part in our series of three experimental tasks involving acquisition of multi-touch gestures. Participants' ages varied between 22 and 35 years (mean 27.4, SD=3.4 years). All participants, except one, were right-handed. All of the participants have majors outside of the human computer interaction field and none of them are a designer. Participants occupations include chemists, biologists, electronic, mechanics, researcher in networks and telecommunications and graduate students. Participants nationalities include different European, African, Canadian and Asian countries. Fourteen participants were regular users of smart phones and tablet devices with multi-touch screens, but none had previous experience with working with interactive tabletops as shown in Table. 5.1. A Friedman test revels that the expertise of participants differ significantly by the type of interactive surface ($\chi^2(3)$=29.34, p<.001).

	Smartphone	Tablet	Tablet PC	Tabletop
Never	3	11	18	18
Occasional	2	2	0	0
Regular	13	5	0	0

Table 5.1: Distribution of usage of touchscreen devices among our participants.

5.1.2 Apparatus

Multi-touch gestures were collected on a 32 inch (813 mm) 3M^{TM} multi-touch display (model C3266PW), measuring 698.4×392.8 mm^2, which was set to work at a screen resolution of 1920×1080 pixels^2. The display supports detection of up to 40 simultaneous touches. The device was positioned horizontally on a table and connected to a computer running Windows and our custom data acquisition software. The interface of the experiment application showed a gesture creation area of 1140×950 pixels^2 (415×345 mm^2), an image of the target gesture that should be articulated with the corresponding instruction. Three buttons were available to control each trial: Start, Save and Clear. The experiment starts when the participant pressed on the Start button. After performing the gesture, two buttons were enabled representing a choice between flagging their input as incorrect or continuing to the next gesture. Participants were instructed to flag a gesture as incorrect if the shape they entered was different from the target gesture, or if some accidental input occurred such as a finger touching the surface unintentionally. This was logged as an input error and the participant was asked to re-execute the gesture. Like Wobbrock et al. [Wobbrock 2009], we wanted our participants to decide whether a gesture was similar to the template, avoiding any confounding effects due to the behavior of a recognizer. As an extra precaution, all participant executions were visually inspected by one experimenter and confirmed that they were correctly entered. The application logged touch coordinates with associated timestamps and identification numbers. Once a gesture was saved, participants could proceed to the next trial in the experiment. Our software randomly presented the trials to participants. Sessions were, also, video recorded and the think-aloud protocol was used throughout all experimental tasks .

5.1.3 Procedure

We controlled the following conditions for multi-touch gesture articulation: (1) number of fingers, (2) number of strokes, and (3) number of hands and their sequential and synchronous use during articulation. In order to keep the overall duration of the data collection procedure manageable for our participants (which was of approximately one hour and 20 minutes per participant), the effect of each condition was tested separately, which resulted in three individual experimental tasks. The order of the three individual tasks was counter-balanced across participants. There are 6 possible orderings: each ordering was experimented with 3 participants. Before beginning the first task, participants completed a background questionnaire to collect demographic information and previous touchscreen experience.

Task #1: Effect of number of fingers. Each trial began by presenting participants with the gesture to articulate and the number of fingers to use: one (1F), two (2F), and three or more fingers (3+F). For the 3+F condition, participants were instructed to use their preferred number of fingers, as long as there were more than three. Participants were instructed to enter gestures at normal speed, using their preferred hand, fingers, and number of gesture strokes. The order of trials was randomized across participants. Each gesture type was articulated with five repetitions, resulting in 10 gestures $\times$ 3 finger conditions $\times$ 5 repetitions = 150 articulations for each participant. The 5 repetitions for each gesture type were not successive. This task took in average 30 minutes to complete.

Task # 2: Effect of number of strokes. Each trial began by presenting participants with the gesture to articulate and the number of strokes to use: one (1S), two (2S), and three or more strokes (3+S). For the 3+S condition, participants were instructed to use their preferred number of strokes, as long as there were more than three. Participants articulated gestures at normal speed and using their hand and their finger of choice, with 10 gestures $\times$ 3 stroke conditions $\times$ 5 repetitions = 150 articulations per participant. The 5 repetitions for each gesture type were not successive. The task took in average 30 minutes to complete.

Task #3: Effect of number of hands and articulation synchronicity. Each trial began by presenting participants with the gesture to articulate and the number of hands to use. In the one hand condition, all the gesture strokes had to be entered one after the other. We also refer to this condition as sequential articulation (SEQ). When two hands had to be used, participants were instructed to articulate strokes in parallel, which we refer to as the synchronous articulation condition (SYNC). Participants were given freedom with respect to the number of strokes they articulated and the number of fingers employed. Each gesture was executed for five times in each condition, resulting in 10 gestures $\times$ 2 hand conditions $\times$ 5 repetitions = 100 articulations per participant. The 5 repetitions for each gesture type were not successive. This task took around 20 minutes to complete.

5.1.4 Post-Experiment Questionnaire

Perceived articulation difficulty was collected after each experimental task using absolute RATING and relative RANKING measurements. RATING was collected in the form of a 5-point Likert scale (Table 5.2), which was presented to participants as a table with five columns, one column for each RATING value. Participants were asked to draw each gesture in the appropriate difficulty column, after having articulated it one more time on the multi-touch surface in order to re-enact the articulation experience and, consequently, perceived difficulty. Participants were allowed to change the ratings of previously rated gestures at any time as they moved along with the rating process until they were confident of the final classification of gesture types into difficulty classes.

After rating each gesture under each condition, participants were asked to provide an ordered list of gestures and conditions in increasing order of perceived difficulty, which

Likert RATING	Explanation provided to participants
1. very easy to execute	I executed these gestures immediately and effortlessly with absolutely no need to pay attention.
2. easy to execute	I executed these gestures easily, almost without paying attention.
3. moderate difficulty	I occasionally paid attention during execution.
4. difficult to execute	I paid special attention with each execution.
5. very difficult to execute	I had to concentrate for each execution. There were times when I did not get the right shape from the first attempt.

Table 5.2: Likert questions employed to elicit absolute articulation difficulty of multi-touch gestures (RATING).

represented the RANKING measurement. The ordered ranking of all gestures according to ascending execution difficulty was completed after the Likert rating to enable participants to use the rating classes to assist with this otherwise difficult task. As before, we asked them to draw the gestures in order to revisit relative differences in difficulty as they completed the ranking. We also asked participants to explain their perception of gesture difficulty: what they found difficult or easy for each gesture execution. Finally, we asked them to identify which shapes they found familiar (they had seen and practiced before) in order to test our choice for familiar and unfamiliar gestures.

5.1.5 Gesture Set

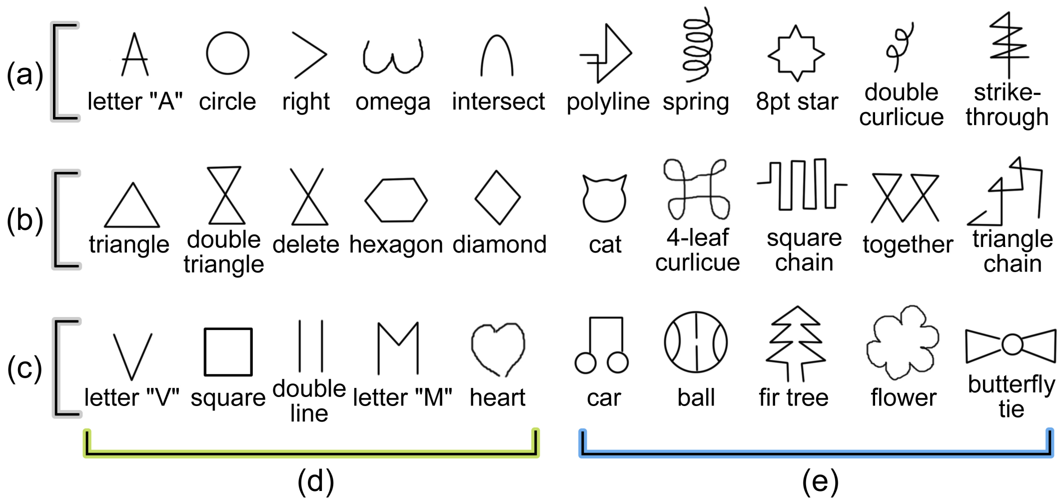


Figure 5.1: The set of 30 gestures used during our experimental tasks : (a) the effect of finger count, (b) stroke count, and (c) synchronicity of the articulation. The left 15 gestures were designed to be familiar (d) and the right 15 unfamiliar to our participants (e).

We employed 30 gesture types in our series of three experimental tasks, with 10 gestures per experimental task(see Figure 5.1). All gestures for the stroke task were carefully chosen so that they could be articulated both as single and multi-strokes. All gestures selected for the number of hands task expose a symmetry axis, as it has been observed previously that only symmetric gestures can be conveniently parallelized during articulation (see Chapter 3). In addition, half of the gesture shapes were selected to appear *familiar* to our participants, where we defined the *familiarity* of a shape as previous, frequent practice articulating that specific shape during everyday handwriting (*e.g.*, familiarity occurs for example for letters and simple geometric shapes). The remaining half of the gesture set was designed to be unfamiliar for our participants, so that we could collect the subjective perception of articulation difficulty on the first encounter with a new geometric shape (e.g., the four curlicue or the stroke-through gestures of Figure 5.1). This approach at gesture set design follows the methodology of Vatavu et al. [Vatavu 2011b], and aligns with current observations from the gesture literature that shape familiarity and practice affect human performance when articulating gestures. For instance, Cao and Zhai [Cao 2007] argue that familiarity affects actual performance time due to practice. The idea is that a more practiced gesture will result in a lower performance time in spite of high objective geometric complexity. We also note that the familiarity of a gesture is more related to practice than to the complexity of its geometrical shape [Vatavu 2011b]. For instance, a gesture can be considered familiar and, consequently, rated "easy to perform" by users although it is geometrically complex, and the opposite is also true (e.g., see letter g and discussion from [Vatavu 2011b]). In this work, we investigate symbolic multi-touch gestures instead of standard gestures traditionally used for multi-touch interaction (*e.g.*, pan, zoom, rotate, etc.), because symbols are more versatile to generalize for other applications.

5.1.6 Design

Our experiment was a within-participants design with four independent variables:

1. FINGER-COUNT, ordinal with 3 values corresponding to using one (1F), two (2F), and three or more fingers (3+F).
2. STROKE-COUNT, ordinal with 3 values corresponding to producing one (1S), two (2S), and three or more strokes (3+S).
3. SYNCHRONICITY, nominal with 2 values: one hand sequential input (SEQ) and bimanual synchronous input (SYNC).
4. GESTURE, nominal with 10 values per task, and 30 values for the entire experiment (see Figure 5.1).

The dependent variables were participants' absolute RATING and relative RANKING of perceived articulation difficulty, which were collected during a post-experiment questionnaire:

1. RATING, ordinal variable, measures the absolute difficulty of articulating a multi-touch gesture into five levels, and was presented in the form of a Likert scale (see Table 5.2).
2. RANKING, ordinal variable, measures the relative difficulty of articulating multi-touch gestures. RANKING takes values in the set $\{1,2,\ldots 30\}$ for the first two tasks (corresponding to 10 gesture types $\times$ 3 conditions), and in the set $\{1,2,\ldots 20\}$ for

the third task (*i.e.*, 10 gestures $\times$ 2 conditions).

5.2 Assumptions about the Familiarity of Gesture Types

Half of the gestures we employed (15 gestures) were selected to look familiar to participants (see our definition of familiarity in the section 5.1.5), while the other half were new gestures, specifically designed for our experimental tasks. In order to validate our initial assumptions about gesture familiarity, we asked participants to report which gesture types looked familiar to them. In the following, we report results on the assumptions of the familiarity for gestures of the (1) finger count, (2) stroke count, and (3) articulation synchronicity experimental tasks.

Assumptions of familiarity for gestures of finger count experiment. Across all 18 participants there were 7 deviations (6.7% (=7/180) of the total responses) from our gestures set's assumed Familiarity. The 7 deviations were assumed familiar gestures: 3 participants found the double curlicue gesture familiar and 4 for spring gesture. They witnessed no previous practice with these shapes, however, because, these two gesture types are composed on a set of series of twists and 180-degrees turns, they felt familiar with them.

Assumptions of familiarity for gesture of the stroke count experiment. Across all 18 participants there were 23 deviations (12.77% (=23/180) of the total responses) from our gestures set's assumed Familiarity. 20 deviations were assumed familiar gestures instead of unfamiliar: 5 participants found the cat gesture familiar, 6 four curlicue, 7 square chain and 2 for together. Again, participants witnessed no previous practice with these gestures. The 3 deviations that were assumed unfamiliar gestures instead of familiar are: 2 participants found that delete gesture is not familiar gesture and 1 of them found also that double triangle was unfamiliar for him.

Assumptions of familiarity for gestures of the synchronicity experiment. Across all 18 participants there were 13 deviations (7.22% (=13/180) of the total responses) from our gestures set's assumed Familiarity. The 13 deviations were assumed familiar gestures: 4 participants found the ball gesture familiar, 3 butterfly tie, 3 car (all of them consider this gesture a musical note), 2 fir tree and 1 for the flower. For these gesture, participants felt that they have articulated in their childhood a long years before (more than 10 years) when they are at the kindergarten but not after.

Summary. Overall, we counted 43 deviations from our gesture set assumed familiarity out of 540 total answers, which represents an error rate of less than 8%. Gestures such as ball, butterfly-tie, four curlicue, and square chain were classified as familiar by some participants, although they witnessed no previous practice or a very old practice more than 10 years ago with these shapes. Under these circumstances, we can safely consider the assumptions of familiarity being met for our gesture set.

5.3 Users' Consensus on Perceived Articulation Difficulty

We are interested in this section in the level of agreement between users in terms of perceived difficulty of articulating multi-touch gestures under various articulation conditions. To this end, we report and analyze 144 individual ratings of absolute difficulty and 54 rankings of relative difficulty collected from 18 participants. In the following, we report results on the effects of (1) finger count, (2) stroke count, and (3) articulation synchronicity on the user-perceived difficulty of gesture articulation.

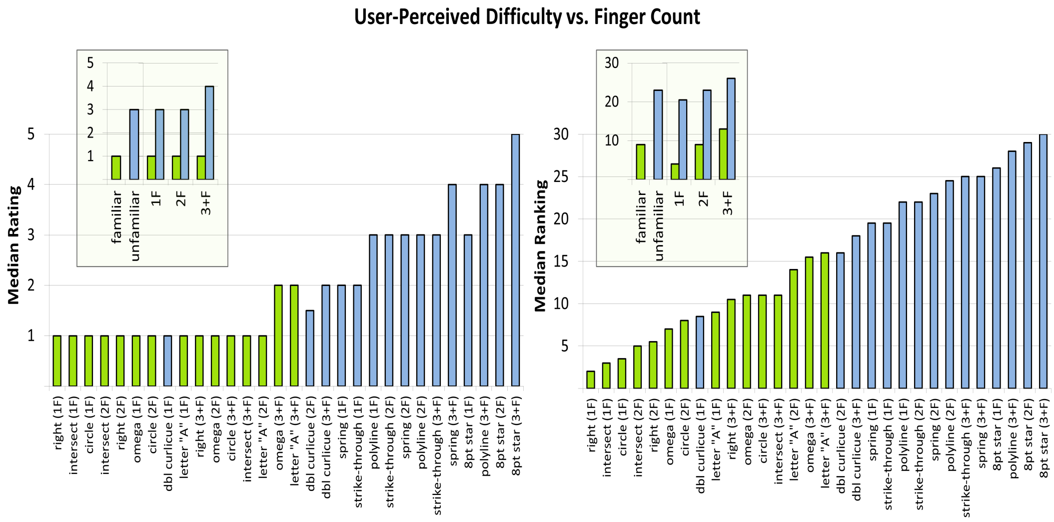


Figure 5.2: Median values for the user-perceived difficulty of articulating multi-touch gestures, measured as absolute RATING (left) and relative RANKING (right), function of the number of employed fingers: one (1F), two (2F), and three or more fingers (3+F). NOTE: In both graphs, gestures are ordered by ascending RANKING values; higher numerical values show larger articulation difficulty.

Effect of finger count on articulation difficulty. Figure 5.2 illustrates the participants' responses of perceived difficulty for multi-touch gestures articulated with one (1F), two (2F), and three or more fingers (3+F). Overall, we found a high degree of agreement between participants' responses, as reflected by Kendall's coefficient of concordance[1] (W=.77, $\chi^2(29)$=403.340, p<.001 for RATING and W=.82, $\chi^2(29)$=428.748, p<.001 for RANKING). The degree of agreement between participants also stayed high when we ran the analysis for each value of the finger count condition (i.e., one, two, and three or more fingers). These agreements stayed above .80 for RATING (W=.83, $\chi^2(9)$=134.318, p<.001 for 1F ; W=.80, $\chi^2(9)$=128.683, p<.001 for 2F and W=.83, $\chi^2(9)$=134.221, p<.001 for 3+F) and above .85 for RANKING (W=.87, $\chi^2(9)$=140.497, p<.001 for 1F ; W=.85, $\chi^2(9)$=137.758, p<.001 for 2F and W=.86, $\chi^2(9)$=139.333, p<.001 for 3+F).

[1]Kendall's coefficient of concordance is a normalization of the statistic of the Friedman test, used to asses community of judgement among multiple individuals. W takes values in [0..1], where 0 denotes no agreement at all and 1 perfect agreement [Kendall 1939].

We, also, conducted a series of Friedman tests to determine if there is or not an effect of the number of fingers on the perceived difficulty and where differences may lie. We found a significant effect of the number of employed fingers on perceived articulation difficulty measured as both RATING ($\chi^2(2)$=23.130, p<.001) and RANKING ($\chi^2(2)$=21.778, p<.001), with more fingers leading to an increase in the perceived difficulty of gesture articulations (see Figure 5.2, inner graphs). Post-hoc Wilcoxon signed-rank tests (Bonferroni corrected at p=.01/2=.005) confirmed a significant difference between two and three or more fingers (with a medium to large effect, r=.51), and a $n.s.$ difference between one and two fingers.

Familiar gestures were rated as being less difficult to execute (median RATING=1 corresponding to "very easy to execute", median RANKING=9) than unfamiliar gestures (median RATING=3 or "moderate difficulty", and median RANKING=23). These differences were significant, as confirmed by Wilcoxon signed-rank tests ($z_{(N=18)}=-3.782$, p<.001, $r=-.63$). The fifteen familiar gestures were among the first sixteen gestures in ascending order of RATING and RANKING (Figure 5.2), with 13/15=87% of familiar gestures being rated as "very easy to execute" (RATING=1). At the same time, 10/15=67% of the unfamiliar gesture articulations were rated from "moderate" to "very difficult to execute" (RATING=3 to 5). We also found people being less consistent when rating familiar than unfamiliar gestures (W=.28 with $\chi^2(14)$=70.631 versus .65 with $\chi^2(14)$=162.463, all p<.001) for RATING and W=.56 with $\chi^2(14)$=141.221 versus .72 with $\chi^2(14)$=180.283 for RANKING, all p<.001), which suggests that people develop different preferences in articulating gestures with practice in terms of preferred number of fingers.

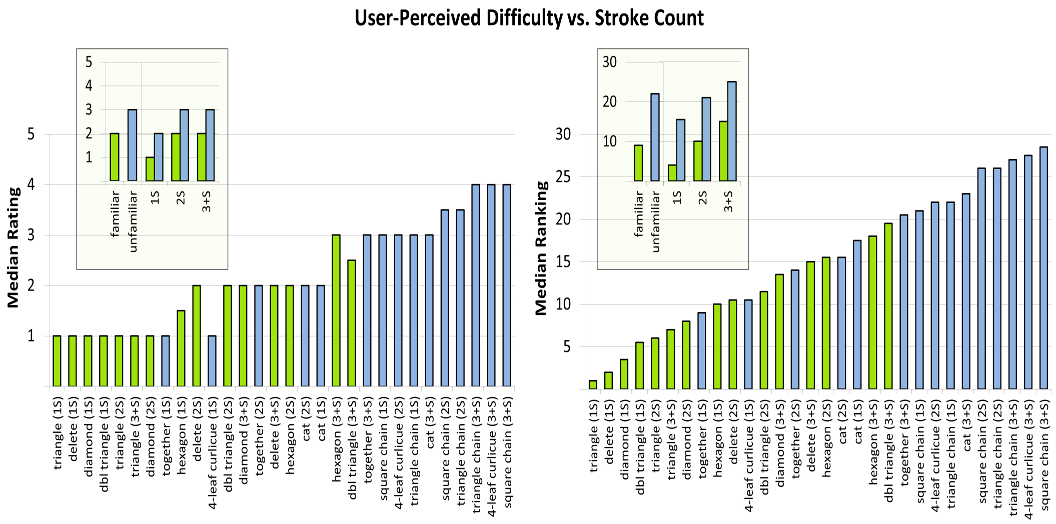


Figure 5.3: Median values for the user-perceived difficulty of articulating multi-touch gestures, measured as absolute RATING (left) and relative RANKING (right), function of the number of gesture strokes employed during articulation: one (1S), two (2S), and three or more strokes (3+S). NOTE: In both graphs, gestures are ordered by ascending RANKING values; higher numerical values show larger articulation difficulty.

Effect of stroke count on articulation difficulty. Figure 5.3 illustrates participants' RATING and RANKING assessments of the self-perceived difficulty when articulating multi-touch gestures with one (1S), two (2S), and three or more strokes (3+S). This time, we found a lower degree of consensus between participants' assessments of articulation difficulty, as opposed to the finger count experiment. However, Kendall's coefficients of concordance stayed above .50, which shows large Cohen effect sizes[2] (W=.58, $\chi^2(29)$=308.816, p<.001 for RATING and W=.71, $\chi^2(29)$=369.095, p<.001 for RANKING). When calculating agreement for each stroke condition, Kendall's W coefficients stayed above .50 for RATING (W=.51, $\chi^2(9)$=82.041, p<.001 for 1S ; W=.52, $\chi^2(9)$=84.262, p<.001 for 2S and W=.64, $\chi^2(9)$=104.060, p<.001 for 3+S) and above .60 for RANKING (W=.73, $\chi^2(14)$=118.461, p<.001 for 1S ; W=.60, $\chi^2(9)$=96.182, p<.001 for 2S and W=.71, $\chi^2(9)$=115.515, p<.001 for 3+S). The lower degree of consensus compared to the previous, finger count experiment, suggests STROKE-COUNT a factor with a stronger influence on the self-perceived difficulty of articulated gestures than FINGER-COUNT.

We, than, conducted a series of Friedman tests to determine if there is or not an effect of the number of strokes on the perceived difficulty and where differences may lie. We found a significant effect of the number of strokes on perceived difficulty for both RATING ($\chi^2(2) = 23.049$, $p < .001$) and RANKING measures ($\chi^2(2) = 32.444$, $p < .001$), with more strokes leading to an increase in the self-perceived difficulty of gesture articulation (see Figure 5.3, inner graphs). In average, gestures articulated with one stroke were perceived as "very easy" and "easy to execute" (median RATING = 1.5), two-stroke gestures as "easy to execute" (median RATING = 2), and gestures articulated with three or more strokes having "moderate difficulty" (RATING = 3). Post-hoc Wilcoxon signed-rank tests confirmed significant differences (at p=.01/2=.005) between the 1S and 2S condition, as well as between 2S and 3+S (with medium to large effect sizes, r>.50).

As in the previous experiment, familiar gestures were rated less difficult to execute (median RATING=2 corresponding to "easy to execute", and median RANKING=9) than unfamiliar gestures (median RATING=3, "moderate difficulty", and median RANKING=22.5). These differences were significant, as confirmed by Wilcoxon signed-rank tests ($z_{(N=18)}=-3.535$, p<.001, r=−.60). The fifteen familiar gestures were among the first twenty gestures in ascending order of RATING and RANKING (Figure 5.3), with 14/15=93% of familiar gestures being rated as "very easy" and "easy to execute" (RATING values 1 and 2). At the same time, 10/15=67% of the unfamiliar gesture articulations were rated from "moderate" (RATING=3) to "difficult to execute" (RATING=4). Again, we found people being less consistent when rating familiar than unfamiliar gestures (W=.40 versus .51 for RATING and W=.61 versus .57 for RANKING, all p<.001), which suggests that people develop with practice different preferences in articulating gestures in terms of number of strokes.

[2]Kendall's W coefficient is related to the average of $\binom{n}{2}$ Spearman rank correlation coefficients between pairs of the n rankings [Kendall 1939] (p. 276). Therefore, we use Cohen's suggested limits of .10, .30, and .50 for interpreting the magnitude of the effect size [Cohen 1992].

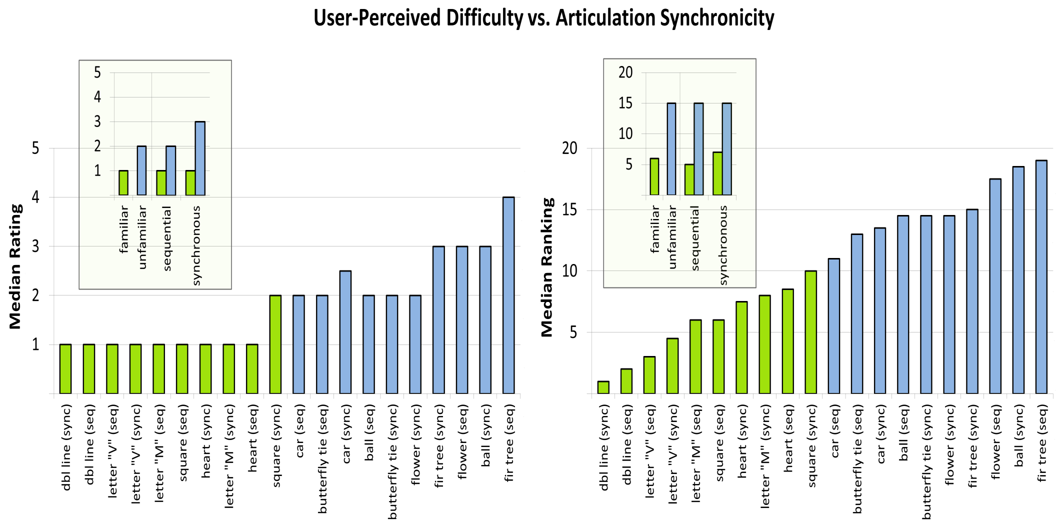


Figure 5.4: Median values for the user-perceived difficulty of articulating multi-touch gestures, measured as absolute RATING (left) and relative RANKING (right), function of the synchronicity of gesture articulation: sequential and synchronous. NOTE: In both graphs, gestures are ordered by ascending RANKING values; higher numerical values show larger articulation difficulty.

Effect of synchronicity on articulation difficulty. Figure 5.4 illustrates participants' assessments of difficulty in terms of RATING and RANKING for multi-touch gestures articulated with strokes in sequential order (i.e., one stroke at a time) and synchronously (i.e., strokes articulated in parallel). The degree of consensus between participants, when asked to rate and rank the difficulty of gestures articulated under synchronous and sequential conditions, stayed above .59 (W=.59, $\chi^2(19)$=201.477, p<.001 for RATING and W=.75, $\chi^2(19)$=257.921, p<.001 for RANKING). We found a higher degree of consensus among participants when rating gestures articulated with strokes in sequential order than in parallel (W=.73 with $\chi^2(9)$=117.441 versus .53 with $\chi^2(9)$=85.906 for RATING and W=.86 with $\chi^2(9)$=139.648 versus .70 with $\chi^2(9)$=113.903 for RANKING, all p<.001). However, a Wilcoxon signed-rank test shows that there was no significant effect of articulation type (sequential or synchronous) on the self-perceived difficulty for either RATING ($z_{(N=18)}=-1.363$, *n.s.*) or RANKING measures ($z_{(N=18)}=-1.244$, *n.s.*).

As in the other experimental tasks, familiar gestures were rated less difficult to execute (median RATING=1 corresponding to "very easy to execute", and median RANKING=6) than unfamiliar gestures (median RATING=2, "easy to execute", and median RANKING=15). These differences were significant, as confirmed by Wilcoxon signed-rank tests ($z_{(N=18)}=-3.680$, p<.001, $r=-.61$). The ten familiar gestures were the first in ascending order of RATING and RANKING (Figure 5.4), with 9/10=90% of familiar gestures being rated as "very easy" (RATING=1). Again, our participants showed less agreement when rating familiar than unfamiliar gestures (W=.26 with $\chi^2(9)$=41.858 versus .37with $\chi^2(9)$=59.167 for RATING and W=.61 with $\chi^2(9)$=98.400 versus .39 with $\chi^2(9)$=62.618 for RANKING, all p<.001), which suggests that different articulation preferences are developed with practice.

Summary. Our results complement the findings of Vatavu et al. [Vatavu 2011b] for unistrokes by extending them to multi-stroke multi-touch bimanual gestures, which possess considerably more degrees of freedom, and are more complex to characterize and articulate in terms of finger count, stroke count, and single and bimanual input. During our series of three experimental tasks, we found participants highly consistent in terms of assessing the difficulty of articulating multi-touch gestures under various FINGER-COUNT, STROKE-COUNT, and SYNCHRONICITY articulation conditions, as indicated by Kendall's coefficients of concordance. Participants were more consistent in their assessments of the difficulty of gestures articulated with various number of fingers (W=.77 and .82) than under varying number of strokes (W=.58 and .71) or single and bimanual articulation (W =.59 and .75). We found less consensus between participants' RANKING s of familiar than unfamiliar gestures (average W =.31 versus .51), which suggest that people develop with practice different preferences for articulating multi-touch gestures in terms of number of fingers, strokes, and single and bimanual input. Overall, more fingers and more strokes were significantly related to more perceived difficulty during gesture articulation. However, there were also interesting, no significant differences between the self-perceived difficulty of gestures articulated with one and two fingers, or between synchronous and sequential articulations.

5.4 Articulation Differences of Multi-Touch Gestures of Various Difficulty Levels

While the analysis of Rating and Ranking scores have given us the level of agreement between users in terms of perceived difficulty of articulating multi-touch gestures under various articulation conditions, we are also interested in evaluating the effect of the various articulations conditions on the structure, the geometric and kinematic gesture descriptors. Structure descriptors measure how consistent users are in producing stroke gestures by looking at the difference in number of strokes and number of points. Geometric descriptors reflect how well users are able to reproduce a gesture, given its geometric shape alone. Kinematic descriptor captures differences in the time domain and, therefore, informs how fluent or smooth the gesture path is. We apply these descriptors to our dataset that is composed of $7,200$ samples of 30 gesture types collected from 18 participants under various articulation conditions.

5.4.1 Gesture Descriptors

We selected several gesture descriptors commonly employed in the gesture recognition and analysis literature [Anthony 2013b, Blagojevic 2010, Rubine 1991a, Vatavu 2011b, Anthony 2013a, Willems 2009] in order to characterize the articulation of gesture types belonging to different difficulty classes. In order to not overload our discussion with exten-

No.	Gesture descriptor	Definition	Units
A. Gesture structure			
1	Number of points	$\|p\| = n$	count
2	Number of strokes	number of finger-down / finger-up events required to produce the gesture structure	count
3	*Actual* number of strokes	number of finger-down / finger-up events *actually* occurring, computed cumulatively for all employed fingers (e.g., letter A articulated with 3 strokes and 2 fingers results in value 6 for this descriptor, compared to 3 for the previous one. In case of finger slips, the reported value will be >6).	count
B. Gesture geometry and visual appearance			
4	Path length	sum of Euclidean distances between adjacent stroke points, $L = \sum_{i=2}^{n} \|\|p_i - p_{i-1}\|\| = \sum_{i=2}^{n} (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2$. This measure is independent of the number of fingers (e.g., path lengths of letter A articulated with 1 and 3 fingers will roughly have the same value). To compute this measure, strokes articulated by different fingers were averaged together.	pixels
5	*Actual* path length	sum of Euclidean distances between adjacent points inside a stroke, computed cumulatively for all employed fingers (e.g., a two finger letter A will be approximately double in path length than when articulated with one finger only).	pixels
6	Size	area of the gesture bounding box, $(max(x_i) - min(x_i)) \cdot (max(y_i) - min(y_i))$	pixels^2
7	Aspect ratio	width of the bounding box divided by its height, $(max(x_i) - min(x_i)) / (max(y_i) - min(y_i))$	-
8	Total absolute turning angle	sum of absolute angles between adjacent segments, $\sum_{i=2}^{n-1} \|\phi_i\|$. This measure refers to f_{10} Rubine's feature [Rubine 1991a] (p. 333).	radians
9	*Actual* total absolute turning angle	sum of absolute angles between adjacent stroke segments, computed cumulatively for all employed fingers	radians.
C. Gesture kinematics			
10	Production time	total articulation time, $T = t_n - t_1$	ms
11	Average speed	path length divided by production time, L/T	pixels/ ms

Table 5.3: Gesture descriptors employed in this study, selected from previous works [Anthony 2013b, Blagojevic 2010, Rubine 1991a, Vatavu 2011b, Anthony 2013a, Willems 2009]. NOTE: A gesture is represented as a set of two-dimensional points (x_i, y_i) with associated timestamps t_i, $p = \{p_i = (x_i, y_i, t_i) \,|\, i = 1..n\}$.

sive numerical data, we limit our analysis to a subset of 11 representative descriptors that we believe adequate to characterize gesture articulations in terms of (1) gesture structure, (2) geometry and visual appearance, and (3) kinematics [3](see Table 5.3). For example, we characterize the structure of a multi-touch gesture by its number of touch points and number of strokes; gesture geometry is characterized by path length, bounding box area size, and several measures that employ turning angles; and production time and speed are our measure of choice to study the kinematics of multi-touch gestures. We are also interested in capturing subtle details specific only to multi-touch input. To this end, we proposed variations for computing some of the most used gesture features, and we denote these variations as *actual* number of strokes, *actual* path length, and *actual* total absolute turning angle. The term *actual* placed in front of a descriptor means the descriptor is computed using data points from all fingers touching the surface. For example, the *actual* path length is defined as the cumulative length produced by all fingers touching the surface, whereas (simple) path length produces a value independent of finger count (reflective of the path length irrespective of how it was articulated). Please refer to Table 5.3 for a summary of all gesture descriptors with accompanying calculation formulas and examples.

We computed Pearson correlations between all gesture descriptors and participants' self-perceived articulation difficulty reported as absolute RATING and relative RANKING measures. Tables 5.4, 5.5 and 5.6 show the correlation values for each experimental condition. The largest value in each column is highlighted as there are all the other values not significantly different from it. In the following, we discuss specific findings for each gesture descriptor category.

5.4.1.1 Gesture Structure and Articulation Difficulty

We measure the structure of a multi-touch gesture as its number of touch points, number of strokes, and the *actual* number of produced strokes (Figure 5.5). Number of touch points refers to the total number of touch events registered during a gesture, cumulatively over all touch-strokes. Number of strokes corresponds to the number of key strokes produced. *Actual* number of produced strokes refers to the total number of finger-down to finger-up periods registered during a gesture that includes the effect of employing multiple fingers or unintentional finger slips.

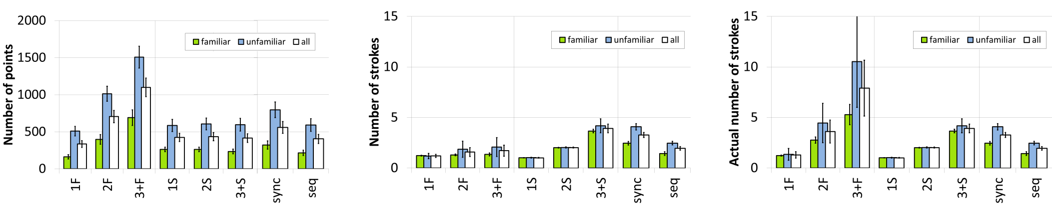


Figure 5.5: Mean number of points (left), number of strokes (middle), and *actual* number of strokes (right) for multi-touch gestures articulated with different number of fingers (1F, 2F, 3+F), strokes (1S, 2S, 3+S), and synchronicity (sync / seq).

[3]See Blagojevic et al. [Blagojevic 2010] for a comprehensive set of gesture features.

Effect on finger count. Clearly, (and without surprise) more fingers touching the surface will generate more touch points sampled by the device for the gesture path, with an expected and uninteresting linear relationship between the number of sampled points and the number of employed fingers (Figure 5.5, left). At the same time, there was a significant effect of FINGER-COUNT on the number of produced strokes ($\chi^2(2)$=127.96, p<.001) as well as on the *actual* number of strokes ($\chi^2(2)$=1677.32, p<.001), with more fingers related to more strokes being produced (Figure 5.5, middle and right).

Effect on stroke count. Interestingly, we found that strokes count do not affect the mean number of points produced (Figure 5.5, left). In fact, we found that number of points measure increased slightly with gestures produced with two strokes compared to those produced with one stroke and was approximately the same for gestures produced with three or more strokes (*mean*=424.14 ; *SD*=252.42 for 1S; *mean*=434.08; *SD*=263.36 for 2S and *mean*=415.20; *SD*=268.84 for 3+S). However, a Friedman test revealed that this slight growth is significant ($\chi^2(2)$=31.77, p<.001). A Post-hoc Wilcoxon signed-rank tests with Bonferroni correlation showed the significant differences between one and two strokes groups (p<.05).

Effect on synchronicity. Significantly more touch points and more strokes (≈1.5 times more) were produced when participants were asked to articulate strokes synchronously rather than sequentially ($z_{(N=900)}$=23.56, p<.001, r=.56 for touch points and $z_{(N=900)}$=22.87, p<.001, r=.54 for number of strokes).

Effect on gesture familiarity. We also found a significantly larger number of points produced for unfamiliar gestures that was more than two times the number of points produced for familiar ones ($z_{(N=1350)}=-31.68$, p<.001, $r=-.61$ in the finger count, $z_{(N=1350)}=-31.66$, p<.001, $r=-.61$ in stroke count experimental tasks and $z_{(N=900)}=-25.85$, p<.001, $r=-.61$ in the synchronicity experiment). At the same time, familiar gestures were articulated with significantly less *actual* number of strokes ($z_{(N=1350)}=-3.10$, p<.005, $r=-.06$ in the finger count experimental task and $z_{(N=900)}=-19.45$, p<.001, $r=-.46$ for the synchronicity experiment).

These findings confirm the fact that people tend to be more confident and, consequently, faster when articulating familiar gestures [Cao 2007, Vatavu 2013b], but they also show that people are more careful when synchronizing bimanual movements involved in the articulation of multi-touch gestures.

5.4.1.2 Gesture Geometry and Articulation Difficulty

We characterize gesture geometry in terms of path length (and corresponding *actual* path length), size of the bounding box area, total absolute turning angle (and the correspondent *actual* form), and aspect ratio measurements (Figures 5.6 and 5.7). The three first descriptors are related to the gesture size, the 6 flowed are related to gesture smoothness and the last one is related to gesture deformation. For the 2 smoothness descriptors, we first preprocessed all collected gestures by normalizing without deformation, centering on the origin, and re-sampling uniformly into n=32 points before extracting them.

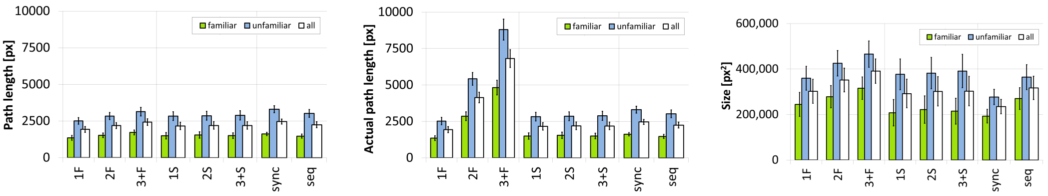


Figure 5.6: Mean path length (left), *actual* path length (middle) and size of the bounding box area (right) for multi-touch gestures articulated with different number of fingers (1F, 2F, 3+F), strokes (1S, 2S, 3+S), and synchronicity (sync / seq).

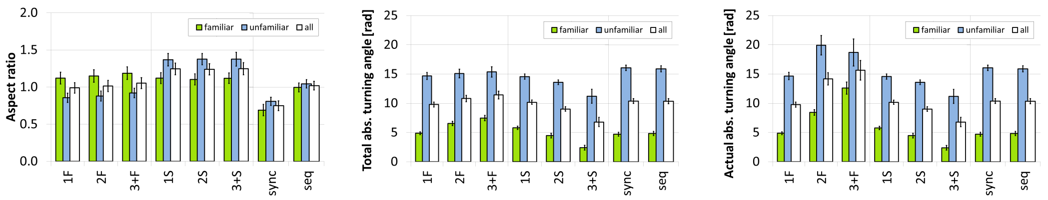


Figure 5.7: Aspect ratio (left), Mean total absolute turning angle (middle) and *actual* total absolute turning angle (right) for multi-touch gestures articulated with different number of fingers (1F, 2F, 3+F), strokes (1S, 2S, 3+S), and synchronicity (sync / seq).

Effect on finger count. Overall, we found a significant effect of FINGER-COUNT on all gesture geometry descriptors. While this finding is obvious and was expected for the actual path length descriptor due to the way it was defined to capture the effect of more fingers touching the surface, the finding is particularly interesting for the other descriptors. Specifically, participants produced gestures significantly longer in path length by $10-25\%$ ($\chi^2(2)$=902.11, p<.001) and larger in area size by $10-29\%$ ($\chi^2(2)$=494.18, p<.001) when more fingers were in touch with the surface. A significant effect was found for the aspect ratio measure ($\chi^2(2)$=98.99, p<.001): participants seemed to deform their gesture shapes by putting more emphasis on the horizontal axis, and producing narrower gestures by 4.35% in average with more fingers. We found also an interesting significant effect for total absolute turning angle ($\chi^2(2)$=559.74, p<.001) (as well as for actual total absolute tuning angle ($\chi^2(2)$=1058.44, p<.001)), with more fingers leading a decrease in the paths smoothness.

Effect on stroke count. Interestingly, path length measure was slightly longer and gesture size was slightly larger for gestures produced with two strokes then those produced with one stroke and approximately the same measures were found for three or more strokes. This difference is coherent with the observation that participants expressed that they fell less satisfied and frustrated with multi-stroke gestures and particularly with two strokes. A Friedman test revealed a significant effect on number of strokes on path length ($\chi^2(2)$=17.84, p<.001) and on gesture size ($\chi^2(2)$=19.10, p<.001). Although significant, the difference is small and we hypothesize that this is due to the hesitation of our participants on how and when dividing the gesture. A Post-hoc Wilcoxon signed-rank tests with Bonferroni correlation showed the significant differences between one and three or more strokes groups (p<.001). However, aspect ratio measure was not affected by STROKE-COUNT ($\chi^2(2)$=1.97, *n.s.*). A Friedman test revealed a significant effect of STROKE-COUNT on the actual total absolute tuning angle ($\chi^2(2)$=1058.45, p<.001). Similarly, a significant effect of STROKE-COUNT was found for actual total absolute tuning angle ($\chi^2(2)$=1046.56, p<.001). Unlike gestures articulated with more fingers, gestures with more strokes presented smoother paths.

Effect on synchronicity. Participants produce gestures with longer path lengths and smaller area size during synchronous articulations compared to sequential ones ($z_{(N=900)}=17.06$, $p<.001$, $r=.40$ for path length and $z_{(N=900)}=-17.04$, $p<.001$, $r=-.40$ for area size). Variations in path length were not complemented by variations in the turning angle measures ($z_{(N=900)}=-0.36$, *n.s.*). However, we found an interesting significant effect of SYNCHRONICITY on the aspect ratios of gesture articulations: participants seemed to deform their gesture shapes significantly more for synchronous articulations ($z_{(N=900)}=-21.54$, $p<.001$, $r=-.51$) than for sequential ones, by putting more emphasis on the vertical axis, and producing narrower gestures by 35% in average.

Effect on gesture familiarity. Familiar gestures were significantly shorter ($\approx$1.8 times shorter) than unfamiliar ones ($z_{(N=1350)}=-31.45$, $p<.001$, $r=-.61$ in the finger count, $z_{(N=1350)}=-31.82$, $p<.001$, $r=-.61$ in stroke count experimental tasks and $z_{(N=900)}=-25.99$, $p<.001$, $r=-.61$ in the synchronicity experiment) and were smaller in area size ($z_{(N=1350)}=-26.97$, $p<.001$, $r=-.52$ in the finger count, $z_{(N=1350)}=-31.06$, $p<.001$, $r=-.60$ in stroke count experimental tasks and $z_{(N=900)}=-18.79$, $p<.001$, $r=-.44$ in the synchronicity experiment). We also found a significant effect of FAMILIARITY on the total absolute turning angle measures ($z_{(N=1350)}=-28.82$, $p<.001$, $r=-.55$ for finger count experiment, $z_{(N=1350)}=-31.80$, $p<.001$, $r=-.61$ for stroke count experimental task and $z_{(N=900)}=-25.98$, $p<.001$, $r=-.61$ for synchronicity experiment), with familiar gestures presenting smoother paths.

5.4.1.3 Multi-touch kinematics and Perceived Difficulty

We employ production time and average speed to describe the kinematic dimension of gesture articulation (see Figure 5.8).

Effect on finger count. We found significant effects of FINGER-COUNT on production time ($\chi^2(2)$=900.05, p<.001) and on average speed ($\chi^2(2)$=111.72, p<.001), with more fingers touching the surface leading to larger articulation times and lower average speed.

Effect on stroke count. Similar to finger count, more strokes led to larger production times ($\chi^2(2)$=953.71, p<.001) and lower average speeds ($\chi^2(2)$=934.22, p<.001).

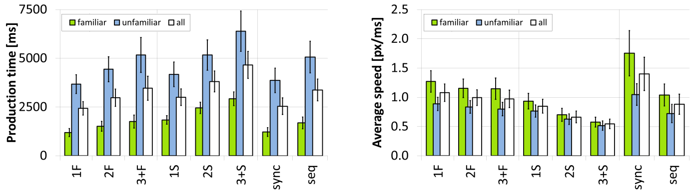


Figure 5.8: Mean production time (left) and average speed (right) for multi-touch gestures articulated with different number of fingers (1F, 2F, 3+F), strokes (1S, 2S, 3+S), and synchronicity (sync / seq).

Effect on synchronicity. Synchronous bimanual articulations were faster than sequential articulation of strokes ($z_{(N=900)}$=−19.66, p<.001, r=−.46), by 33%. We attribute this result to the fact that users tend to be more careful and, therefore, slower when synchronizing hand movements. A Wilcoxon signed-rank test revealed a significant effect of synchronicity on speed ($z_{(N=900)}$=23.86, p<.001, r=.56).

Effect on gesture familiarity. We confirm once more that familiar gestures are produced faster than unfamiliar ones ($z_{(N=1350)}$=−31.72, p<.001, r=−.61 for finger count experiment, $z_{(N=1350)}$=−31.32, p<.001, r=−.60 for stroke count experimental task and $z_{(N=900)}$=−25.91, p<.001, r=−.61 for synchronicity experiment). Familiar gestures are also faster then unfamiliar ones ($z_{(N=1350)}$=24.32, p<.001, r=.47 for finger count, $z_{(N=1350)}$=11.41, p<.001, r=.22 for stroke count and $z_{(N=900)}$=24.21, p<.005, r=.57 for synchronicity experiment).

5.4.2 Correlations between Gesture Descriptors and Participants' Self-Perceived Articulation Difficulty

We computed Pearson correlation coefficients between all gesture descriptors and participants' self-perceived articulation difficulty reported as median absolute RATING and relative RANKING. Tables 5.4, 5.5 and 5.6 show the correlation coefficients computed for each experimental task overall as well as individually for each FINGER-COUNT, STROKE-COUNT and SYNCHRONICITY conditions. The coefficient having the largest value is highlighted in each table together with all the coefficients that were not significantly different from this largest value. In the following, we discuss specific findings for each category of gesture descriptors.

Effect on finger count. Number of touch points has the highest correlations with RATING and RANKING overall; and, it is among the highest correlations when tested separately with 1F, 2F and 3+F gestures groups as well as with familiar and unfamiliar gesture groups. Production time had the second highest correlation flowed by path length, *actual* path length and average speed (negative). (see table 5.4)

	All		1F		2F		3+F		Familiar		Unfamiliar	
	RAT.	RANK.	RAT.	RANK.	RAT.G	RANK.	RAT.	RANK.	RAT.	RANK.	RAT.	RANK.
A. Gesture structure												
Number of points	**.89****	.84**	**.85****	.88**	**.94****	**.91****	**.93****	**.90****	.65*	.86**	**.88****	**.80****
Number of strokes	.65**	.58**	*n.s.*	*n.s.*	.67*	.69*	.71*	.67*	*n.s.*	.54*	.75**	.67*
Actual number of strokes	.71**	.61*	*n.s.*	*n.s.*	.78*	.78*	.84*	.79*	**.70****	.87**	.78**	.65*
B. Gesture geometry												
Path length	.83**	**.88****	.72*	**.89****	.88**	.89**	.84**	.88**	*n.s.*	.72**	.62*	.65*
Actual path length	.73**	.73**	.72*	**.89****	.87**	.89**	.80**	.85**	.60*	.84**	.65*	.57*
Size	.79**	.80**	.78*	.76*	.78**	.81**	.76**	.77**	*n.s.*	.63*	.69**	.68**
Aspect ratio	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
Total absolute turning angle	.65**	.68**	*n.s.*	.70*	.69*	.71*	.73*	.66*	*n.s.*	.54*	*n.s.*	*n.s.*
Actual total abs. turning angle	.53**	.63**	*n.s.*	.70*	.62*	.66*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
C. Gesture kinematics												
Production time	.88**	.85**	.84**	.87**	.91**	.87**	.90**	.86**	.61*	**.88****	.80**	**.80****
Average speed	-.81**	-.84**	-.82**	-.85**	-.79*	-.82**	-.91**	-.87**	*n.s.*	-.62*	-.75**	-.78*

NOTE: Pearson correlation coefficients are reported at $p = .005$ (**) and $p = .05$ (*) significance levels. $N = 30$ for all, $N = 10$ for the 1F, 2F and 3+F conditions, and $N = 15$ for familiar and unfamiliar gestures. The largest coefficient in each column is highlighted. Correlation coefficients not significantly different (at $p = .05$) as indicated by the t statistic for dependent r_s [Chen 2002] than the maximum value in each column are also highlighted.

Table 5.4: Pearson correlations coefficients with median RATING and RANKING for the finger count experimental task.

Effect on stroke count. In table 5.5 we report Pearson correlation coefficients with median RATING and RANKING for the stroke count experiment. Production time has the highest correlations with RATING and RANKING overall; and, it is among the highest correlations when tested separately with 1S, 2S and 3+S gestures groups as well as with familiar and unfamiliar gesture groups. Number of touch points, path length and *actual* path length and gesture size have also high correlation when we tested overall and separately with 1S, 2S and 3+S. We can also remark that for geometric descriptors, there are no significant correlation when testing separably with familiar and unfamiliar gestures groups except path length, *actual* path length and gesture size who correlate moderately with unfamiliar gestures. Intuitively, the larger gestures with longer paths may be more complex, and thus be more difficult to execute. The tendency for geometric descriptors to exhibit higher coefficients in either familiar or unfamiliar gesture groups is most likely because they cannot adapt to the effect of practice. Interestingly, average speed correlate very well

	All		1S		2S		3+S		Familiar		Unfamiliar	
	RAT.	RANK.	RAT.	RANK.	RAT.G	RANK.	RAT.	RANK.	RAT.	RANK.	RAT.	RANK.
Number of touch points	.74**	.79**	.82**	.92**	**.90****	**.95****	.91**	**.91****	*n.s.*	*n.s.*	.61*	.63*
Number of strokes	.60**	.55**	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	.76**	.79**	.66*	.63*
Actual number of strokes	.60**	.55**	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	.76**	.79**	.66*	.63*
B. Gesture geometry												
Path length	.71**	.76**	.69*	.82**	.89**	.90**	.88**	**.91****	*n.s.*	*n.s.*	.54*	.52*
Actual path length	.71**	.76**	.69*	.82**	.89**	.90**	.88**	**.91****	*n.s.*	*n.s.*	.54*	.52*
Size	.73**	.79**	**.84****	**.94****	.81**	.88**	.84**	.86**	*n.s.*	*n.s.*	.66*	.69**
Aspect ratio	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
Total absolute turning angle	.51**	.58**	.66*	.81**	.82**	.87**	.84**	.87**	*n.s.*	*n.s.*	*n.s.*	*n.s.*
Actual total abs. turning angle	.51**	.58**	.66*	.81**	.82**	.87**	.84**	.87**	*n.s.*	*n.s.*	*n.s.*	*n.s.*
C. Gesture kinematics												
Production time	**.90****	**.91****	.83**	.91**	**.90****	**.95****	**.92****	**.91****	**.84****	**.91****	.86**	.85**
Average speed	-.69**	-.73**	-.63*	-.71*	*n.s.*	-.65*	-.65*	*n.s.*	-.63*	-.77**	**-.89****	**-.90****

NOTE: Pearson correlation coefficients are reported at $p = .005$ (**) and $p = .05$ (*) significance levels. $N = 30$ for all, $N = 10$ for the 1S, 2S and 3+S conditions, and $N = 15$ for familiar and unfamiliar gestures. The largest coefficient in each column is highlighted. Correlation coefficients not significantly different (at $p = .05$) as indicated by the t statistic for dependent r_s [Chen 2002] than the maximum value in each column are also highlighted.

Table 5.5: Pearson correlations coefficients with median RATING and RANKING for the stroke count experimental task.

(negative) with familiar and unfamiliar gestures.

Effect on synchronicity. In table 5.6 we report Pearson correlation coefficients with median RATING and RANKING for the stroke count experiment. Production time has the highest correlations with RATING and RANKING overall; and, it is among the highest correlations when tested separately with **Sync** and **Seq** gestures groups as well as with unfamiliar gesture groups. Followed by number of touch points, path length and *actual* path length.

Summary. Out of all structure descriptors, the number of points had one of the strongest Pearson correlations with both RATING and RANKING measures (average r=.84 for RATING and r=.85 for RANKING overall, p<.005). Out of all geometric descriptors, path length presented the highest correlation coefficients with RATING and RANKING (average Pearson r=.80 for RATING and r=.86 for RANKING overall, p<.005). Out of all kinemat-

	All		Sync		Seq		Familiar		Unfamiliar	
	RAT.	RANK.	RAT.	RANK.	RAT.G	RANK.	RAT.	RANK.	RAT.	RANK.
A. Gesture structure										
Number of touch points	.88**	.92**	.89**	.93**	**.96****	.96**	*n.s.*	**.90****	.67*	.66*
Number of strokes	*n.s.*	.49*	*n.s*	.69*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
Actual number of strokes	*n.s.*	.49*	*n.s*	.69*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
B. Gesture geometry										
Path length	.86**	**.95****	.83**	.92**	.91**	**.99****	*n.s.*	.88**	*n.s*	.62*
Actual path length	.86**	**.95****	.83**	.92**	.91**	**.99****	*n.s.*	.88**	*n.s*	.62*
Size	.68*	.59*	.76*	.78*	.87**	.74*	*n.s.*	*n.s.*	.75*	*n.s.*
Aspect ratio	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*	*n.s.*
Total absolute turning angle	.74**	.82**	.65*	.76*	.81**	.87**	*n.s.*	.88*	*n.s.*	*n.s.*
Actual total abs. turning angle	.74**	.82**	.65*	.76*	.81**	.87**	*n.s.*	.88*	*n.s.*	*n.s.*
C. Gesture kinematics										
Production time	**.90****	.91**	**.90****	**.95****	.95**	.97**	*n.s.*	.70*	**.77***	**.78***
Average speed	-.53*	-.60*	-.86**	**-.95****	-.65*	-.76*	*n.s.*	*n.s.*	*n.s.*	*n.s.*

NOTE: Pearson correlation coefficients are reported at $p = .005$ (**) and $p = .05$ (*) significance levels. $N = 20$ for all, $N = 10$ for the Sync and Seq conditions, and $N = 10$ for familiar and unfamiliar gestures. The largest coefficient in each column is highlighted. Correlation coefficients not significantly different (at $p = .05$) as indicated by the t statistic for dependent r_s [Chen 2002] than the maximum value in each column are also highlighted.

Table 5.6: Pearson correlations coefficients with median RATING and RANKING for the synchronicity experimental task.

ics descriptors, production time presented the highest correlation coefficients with RATING and RANKING (average Pearson r=.89 for RATING and r=.89 for RANKING overall, p<.005).

5.5 User Feedback on Perceived Difficulty of Multi-touch Gestures

In this section, we accompany our quantitative data with a qualitative data that capture users' mental models as they articulate a gesture and estimate its level of difficulty.

Effect of finger count on perceived articulation difficulty. Sixteen of 18 participants felt that the number of fingers has the effect of increasing the overall difficulty of performing a gesture. This feeling was especially pronounced for unfamiliar gestures. In fact, participants commented that the impact of finger count is mostly perceivable when the gesture is new to them or when it has a “complicated geometry”. Participants were typically defining a gesture with a complicated geometry as a gesture with a shape that they did not practice before; and when several direction changes/breaks in fingers’ movements are required. For instance, one participant quoted “if the form is natural like the upper gesture and contains one unique movement, I find it easy to perform regardless to the number of fingers. However, when the form is exotic like 8 point star, the more fingers I use the more I should stay concentrated to perform the gesture correctly”. Another participant quoted “For gesture types that I already know, performing the gestures with one or more fingers does not have an effect on difficulty because the image of the gesture is already recorded in my memory; thus, I only have to adapt the number of fingers while I am performing it as usual. However, when the gesture type is unknown to me, I should concentrate both to do it correctly and also to be sure that my fingers are correctly touching the surface."

Most participants expressed that performing a gesture with one or two fingers is almost same; but considering three or more fingers increases the perceived difficulty even for familiar gestures. Two participants commented that using just one finger is more fluid and less constrained. Some participants commented that using three fingers requires more effort and may not be practical, even though it does not impact difficulty. For instance, two participants quoted: “using the ring finger in a three finger gesture is not comfortable". Three participants remarked that “more space is required to perform the gesture correctly when using more than two fingers". Another participant felt that the more she uses fingers the more she needs to segment the gestures to many strokes. Two other participants commented that three fingers gestures are not intuitive because they are “used to engage either 1 or 2 fingers when performing gestures on personal phones!".

Interestingly, two of our participants argued that it is more easy and more conformable for them to perform gestures when many fingers are allowed. One of them felt that “using many fingers reduces the effort of making the gesture because the movements of fingers can be better controlled by the movement of the arm”. The other participant felt more comfortable with many fingers since he had the feeling that “fingers are free to move as if they were drawing in the sand”.

We also remarked that the use of 1 finger is related to the index finger, 2 fingers is always related to the use of the index finger with the middle finger, 3 fingers are related to the use of the index, the middle and ring fingers and 4 fingers refer to the use of all fingers except the thumb.

Effect of gesture stroke count on perceived articulation difficulty. Eliciting the perception of gesture articulation difficulty for stroke count is complicated. Although some general tendencies were clear from our participant feedback; their general feeling was mitigated and highly dependent on gesture types even for familiar gestures. Eight of 18 participants were feeling that in general using many strokes make the gesture more difficult to perform. All of them commented that they have to think about where and when to divide the gesture in order to come with the required number of strokes. For instance, one participant quoted: “before and while I’m performing a gesture, I’m asking my self the following questions: what have I already done? what do I have to do next? Where should I stop next time? And will I have the required space to continue my gesture?”.

Some participants expressed that gestures composed of different directional lines are more natural to perform with multiple strokes; since it require less mental activity to distinguish between strokes compared to gestures containing only curved lines. For instance, one participant quoted “I feel destabilized to stop in the middle of the gesture in order to start a new stroke’ and that it is “less distracting to stop at the end of a direction in order to start a new one”. Another participant remarked that “when the gesture type is a closed form it is easier to perform with one stroke".

This feeling was shared with almost all participants and more or less pronounced depending on gesture types. For instance, with respect to the four curlicue gesture, participants argued that since it contains only curved lines, it is more difficult to decide how to divide the gesture into several parts; and thus, performing it with only one stroke is the easiest. With respect to the cat gesture, ten participants felt that it is easier to perform with two strokes because this particular gesture is a mixture of directional lines and curved lines — in such a way its subdivision into two parts is natural and straightforward. One participant formulated the following argument to describe her general feeling about stroke count: “In general, the rule to follow when performing a gesture is to use as many strokes as distinct forms composing the corresponding symbol, e.g., in together it is clear that it is composed of the repetition of the same form!".

For unfamiliar gestures, two participants expressed that decomposing a form with “a complicated geometry” into several strokes makes it more natural and easier to perform. Interestingly, two other participants felt that the first time that they have performed some gestures (e.g., square chain, triangle chain), it was more easy to perform with three strokes or more; but after repeating the gesture, they were able to memorize it so that they had the feeling that a one stroke gesture is easier. For instance, one of them quoted that “many strokes is nice when I do not know the gesture since it helps me doing it in many steps; but then when I became familiar with gesture I would prefer a single stroke, it is faster!". In accordance with this comment, four other participants expressed that even if gesture count does not impact the difficulty level, it could be “unnecessary” (equivalently “not useful” or “a waist of time") to perform a gesture with many strokes.

Effect of articulation synchronicity on perceived difficulty. Seventeen of 18 participants expressed that synchronizing their two hands to make different strokes simultaneously in parallel can help them to perform some gestures.

For familiar gestures, even though most participants felt that synchronicity has no impact on the level of difficulty, they expressed some *preferences* of one articulation among the other. For instance, 8 participants commented that for heart and double lines they would prefer parallel movements because hands are better coordinated and perform faster. On the other side, seven participants commented that for letters (e.g., M and V), they would prefer performing gestures using sequential asynchronous strokes because this is the usual way to proceed with a pen in their daily life.

For unfamiliar gestures, participants felt that using synchronous hands coordination is easier when the gesture contains some symmetry. Five participants noticed that using parallel movements in the fir tree gesture “is very practice because it contains a clear symmetry and hands can be well coordinated to pack each part of the shape”. Another participant quoted that “it is easy to make parallel movements when the axe of symmetry is parallel to my body". Some participants were also feeling “to perform faster so that less attention is needed compared to a sequential gesture”. Four participants felt that when the gesture shape contains only directional lines or only curved lines, it is easier to perform with synchronous articulations. In the opposite, some remarked that when the gesture shape is a mixture of directional and curved lines, it could be difficult to perform with synchronous articulations. For instance, one participant quoted that “when there is many sub-shapes in the gesture, it is difficult to distinguish between the strokes that can be made simultaneously”.

For the three gestures ball, car and butterfly-tie, participants expressed that there is some difficulty to make them using *only* parallel synchronous articulation. For instance, in the case of the ball gesture, four participants felt that they would prefer making the two vertical lines using only one hand in an asynchronous sequential style; while making the rest of the gesture in parallel by synchronizing both hands. Two participants felt “frustrated to perform the two lines at the same time”. Both of were in fact feeling confused and did not know if they should move both hands in the same or in the opposite directions. Similarly, most of participants felt confused to perform the circles in the car gesture at the same time. They commented that if they were allowed to draw the lines in synchronous articulation, but the circles in asynchronous articulation; then the gesture would be easier to perform. The same remark was also formulated the butterfly-tie where participants expressed that drawing first the circle with one stroke, then drawing the rest of the gesture with parallel movements would reduce the level of difficulty compared to using only synchronous articulations throughout the whole gesture.

Only one participant felt that synchronous hand movements are more difficult to perform than sequential asynchronous ones. He argued that he is “used to use one hand to write or draw on a paper sheet”, so that “it requires more attention to understand how to coordinate both hands at the same time to perform well”. He also commented that when using both hands in his daily life, “each hand is used to make a complementary and different task, e.g., like using the fork and knife when eating; this is why parallel movements are more difficult when hands are making the same thing”.

We also remarked that the use of 1 hand is related to the use of the dominant hand. Besides, the use of two hands is always accompanied with the use of the same finger type (the index finger) on each hand.

5.6 Discussion and Design Guidelines

We found participants highly consistent when assessing the difficulty of articulating multi-touch gestures under various FINGER-COUNT, STROKE-COUNT, and SYNCHRONICITY conditions, as indicated by Kendall’s W coefficients of concordance between .58 and .82.

At the same time, we found less consensus between participants' ratings for familiar and unfamiliar gestures (average W=.31 and .51 respectively), which suggests that people develop different preferences *with practice* for articulating multi-touch gestures.

Overall, more fingers and more strokes were significantly related to more perceived difficulty during gesture articulation. More fingers touching the surface caused longer path lengths, larger gesture sizes, longer production times, but also more strokes being produced, which all suggest more effort to articulate gestures. Articulating gestures with more strokes resulted in longer production times, which is explained by the transition times required to move fingers in air between consecutive strokes. Bimanual input resulted in more strokes, longer path lengths, and faster executions when compared to sequential input, and, interestingly, caused horizontal deformations in gesture shape by 35% in average. Familiar gestures were produced faster, with less strokes, and were shorter, smaller, and smoother than unfamiliar shapes. These results on multi-touch input add to the body of knowledge on gesture articulation difficulty by complementing the findings of Vatavu et al. that investigated unistrokes [Vatavu 2011b].

Informed by our findings, we are able to outline a number of 14 guidelines for designing multi-touch gesture interfaces that address (1) gesture ergonomics, (2) gesture recognizer development, and (3) principles of gesture set design. Note that some of these guidelines are common-sense, however we deliberately chose to state them explicitly, because they followed naturally from our study. Finally, our set of guidelines are in agreement with other recommendations available in the gesture literature [Oh 2013, Wobbrock 2009, Anthony 2013b], while they also open new opportunities for user interface practitioners to design easy-to-produce gestures from the users' perspective.

❶ Multi-touch gesture ergonomics guidelines.

(a) Single-touch unistrokes should be preferred to multi-finger gestures whenever possible (see also [Oh 2013, Anthony 2013b], p. 92 and p. 1133), as they are generally perceived easier to articulate. (Single-touch unistrokes are also produced faster than all other articulations.)

(b) Two-finger gestures should be equally exploited, as they were perceived not more difficult to produce than single-finger articulations (as long as the choice of employed fingers is left to the user to decide). This guideline was suggested by our participants and later confirmed by our gesture descriptor analysis.

(c) Where possible, privilege bimanual over sequential articulations, as they are faster and are perceived no more difficult to produce than sequential articulations. Note however that different users may produce different patterns during bimanual articulation. Also, avoid synchronous input for shapes that do not present easily identifiable axes of symmetry. For instance, we observed that our participants preferred gestures for which the geometrical shape presented a vertical axis of symmetry.

(d) Design for flexible input by allowing users to employ their preferred choice for the number of strokes that structure a multi-stroke gesture. Otherwise, before designing the multi-stroke structure of a gesture shape, observe how users naturally decompose that shape into strokes. Our participants decomposed gestures into lines and curves with similar shape and orientation to maximize parallelization during bimanual input.

We also observed that our participants decomposed directional gestures into multiple line segments whenever there was a direction change. However, they did not like decomposing curved shapes, for which participants usually preferred unistroke gestures.

(e) Prefer familiar to unfamiliar shapes (also see [Anthony 2013b], p. 92), as they are produced faster, smoother and are perceived less difficult to articulate under all tested conditions (*i.e.,* number of fingers, number of strokes and single-handed and bimanual input).

(f) Gesture shapes with complex geometries (i.e., mixtures of lines and curves) should be designed so that (1) they are easy to articulate such that learning and memorization are facilitated (a recommendation suggested by our participants during their first trials) and (2) encourage bimanual (and, therefore, faster) articulations. Design such gesture shapes with an easily identifiable axis of symmetry to guide users during bimanual input (in this regard, also see our guideline (c) above).

(g) Design articulation patterns (*i.e.*, ways to produce gestures) that connect to users' previous gesture practice whenever possible, *e.g.,* our participants preferred to use one finger sequential strokes for letters and numbers as these symbols have always been produced with one contact point in pen writing.

(h) Use objective gesture descriptors to inform gesture design be privileging shapes perceived by users less difficult to articulate. As informed by the results of our correlation analysis, gestures that take longer to produce, are longer in path length and larger in size were also perceived as more difficult to articulate (see Tables 5.4, 5.5 and 5.6).

❷ Multi-touch gesture recognizers guidelines.

(i) Design flexible recognizers that are invariant to users' preferred articulation patterns, especially when these patterns are likely to be perceived equally difficult to articulate, such as employing one or two fingers.

(j) Train gesture recognizers with different articulation patterns in terms of finger count, stroke count, and single and bimanual input. Equivalently, this guideline suggests training recognizers in *articulation-independent* scenarios, which is a step further beyond the current practice of employing *user-dependent* and *user-independent* training procedures.

(k) Detect whether two hands are touching the surface simultaneously and make use of this knowledge to increase the tolerance of shape recognizers to horizontal deformations, as we found users horizontally stretching their gesture shapes by 35% during bimanual synchronous input.

(l) More fingers touching the surface produce longer path lengths, larger sizes, and larger production times. Consequently, employ with caution recognizers that rely on geometric and kinematic gesture descriptors, such as [Rubine 1991a] (p. 335), and do not apply them directly to multi-touch input classification without *a priori* analysis. Instead, investigate which calculation form for gesture descriptors (*i.e.*, standard or *actual*), is more suited to capture the variation present within multi-touch input.

❸ Strategies to map gestures to functions.

(m) Exploit the number of fingers as parameter to increase the expressiveness of gesture input (also see [Wobbrock 2009], p. 1091). For example, assign multi-finger gestures to more complex tasks to intuitively match users' perceptions of articulation difficulty with task complexity. Or, associate the number of fingers with different parameter values by following a proportional mapping, *e.g.,* more fingers employed cause the brush to paint thicker for a drawing application. For instance, our participants felt that they were drawing a thicker stroke when touching the surface with more fingers during single-handed input. Another example is to use more fingers to scroll faster when browsing a document. As the most simple scenario, privilege designs employing one and two finger gestures, as we found them being perceived as equally difficult to articulate.

(n) Use thoughtfully the number of strokes as parameter for gesture input, as we found our participants often expressing= dissatisfaction when entering multiple strokes. When suitable, connect the number of strokes to the complexity of the task to execute, as we found that gestures with more strokes were perceived more difficult to articulate.

Gesture dataset availability. As a service to the community interested in replicating our results or in conducting further investigations on multi-touch difficulty, we make available our set of 7,200 samples of 30 gesture types from 18 users annotated with RATING and RANKING data.[4]

5.7 Chapter Summary

This study represents the first investigation of user-perceived difficulty of articulating multi-touch gestures that is measured as subjective gesture *difficulty ratings* and *rankings*, by examining the effect of finger count, stroke count, and single and bimanual articulation conditions. By employing gesture descriptors and correlation analysis, we were able to report significant high correlations between the subjective perception of difficulty and commonly-employed measures used to describe gesture articulation. In this process, we introduce a new variant of computing multi-touch gesture descriptors. We report novel findings on multi-touch gestures articulated under different conditions enabled by the use of structure, geometric and kinematic descriptors, *e.g.* bimanual articulations result in gesture shapes that are horizontally stretched. Our body of results enable us to compile a set of guidelines for multi-touch gesture set design that correlate multi-touch ergonomics, user-perceived difficulty, gesture recognition and potential gesture to function mappings. In the future, our body of results can prove useful to gesture interface designers for improved gesture set designs by considering practical aspects of user-perceived articulation difficulty to maximize the ergonomicity and ease of execution of their gesture set designs. This study also have helped establish that work is needed on the technical challenges of multi-touch recognition in order to provide interfaces that support flexible interactions. The next chapters focus on tools and techniques that support multi-touch unconstrained gestures.

[4]The dataset can be downloaded for free at `https://sites.google.com/site/yosrarekikresearch/projects/gesturedifficulty`.

Part II

Tools and Techniques for Unconstrained Multi-touch Interaction

"Accommodating gesture variability is a key property of any recognizer."

Wobbrock, Jacob, Wilson, Andrew and Li, Yang (2007)

" The programming of these [multi-touch] gestures remains an art."

Lü, Hao and Li, Yang (2012)

6 Match-Up & Conquer: Structuring and Recognizing Multi-Touch Input

Based on all the results presented previously in this dissertation, it has been shown that multi-touch gesture articulations are versatile. Our findings point to several design recommendations to structure fingers' movements as perceived by users, as well as to the fact that recognition should be improved to allow more flexible interaction. Therefore, this chapter discuss how unconstrained bimanual and multi-finger touch input can be structured to be accurately incorporated in multi-touch recognition.

The versatility of multi-touch input makes prototyping multi-touch gesture recognizers a difficult task that requires dedicated effort because, in many cases, "the programming of these [multi-touch] gestures remains an art" [Lü 2012] (p. 1). To our best knowledge, there are no simple techniques in the style of the $-family [Anthony 2010, Vatavu 2012c, Wobbrock 2007] to be employed by designers for recognizing multi-touch gestures under the large variety of users' articulation behaviors that we have evoked in previous chapters (see Figure 4.3 in Chapter. 4 for some examples). Consequently, designers have recurred to multi-touch declarative formalisms [Kin 2012b] that require dealing with sometimes complex regular expressions; to using toolkits that may be limited to specific platforms only [Lü 2012]; and to adapting single-stroke recognizers for multi-touch [Jiang 2012], in which case the expressibility of multi-touch input might be affected (See also the throughout discussion of Section 2.5 in Chapter 2).

Such a lack of simple techniques for multi-touch gesture recognition has many causes, such as the complexity of multi-touch input with many degrees of freedom, our today's limited understanding of how multi-touch gestures are articulated, and, we believe, *a lack of algorithmic knowledge to preprocess multi-touch input* by considering its specific characteristics. For example, there are many useful techniques available to preprocess raw gestures, such as scale normalization, motion resampling, translation to origin, and rotation to indicative angle that make stroke gesture recognizers invariant to translation, scale, and rotation (see the pseudocode available in [Anthony 2010, Anthony 2012b, Li 2010, Vatavu 2012c, Wobbrock 2007]). Although these techniques are general enough and may be applied to multi-touch gestures as well, they cannot address the specific variability that occurs during multi-touch articulation, such as the use of different number of fingers, strokes, and bimanual input, leaving multi-touch gesture recognizers *non-invariant* to these aspects of articulation.

In this chapter, we address two main research questions which can be formulated as follows:

- How can we structure and process touch points traces in a properly interlocked style to user perception of fingers movements?
- How can we make the accuracy of multi-touch recognition insensible to the versatility of unconstrained bimanual and multi-finger touch input?

In particular, to overcome the lack of algorithmic knowledge to assist in recognizing multi-touch input, we propose a new preprocessing step that is specific to multi-touch gesture articulations. We demonstrate the usefulness of our new preprocessing technique for recognizing multi-touch gestures independently of how they were articulated. To deal with the complex problem of handling users' variations in articulating multi-touch input (see our previous chapters), we inspire from the "Divide & Conquer" paradigm from algorithm design [Cormen 2003] (p. 28), in which a complex problem is broke down successively into two or more subproblems that are expected to be easier to solve. Accordingly, we are able to group together individual strokes produced by different fingers with similar articulation patterns (the "Match-Up" step) that we found to improve the accuracy of subsequent recognition procedures (the "Conquer" step). Match-Up & Conquer (M&C) is thus a two-step technique able to recognize multi-touch gestures independently of how they are articulated, *i.e.*, using one or two hands, one or multiple fingers, one or multiple strokes and synchronous and asynchronous stroke input.

6.1 Multi-Touch Gestures and $P

We employ in this work the $P recognizer [Vatavu 2012c] because previous works found it accurate for classifying multi-stroke gesture input under various conditions [Anthony pear, Vatavu 2012c]. $P employs point clouds to represent gestures and, therefore, it manages to ignore users' articulation variations in terms of number of strokes, stroke types, and stroke ordering. $P has been validated so far on gestures articulated with single-touch strokes, which is typically the case for pen and single-finger input [Anthony pear, Vatavu 2012c]. However, multi-touch gestures exhibit considerably more degrees of freedom, with users articulating gestures with one or both hands, variable number of fingers, and following synchronous and asynchronous gesture production mechanisms. For example, users frequently employ multiple fingers to produce gestures, and even use different fingers to simultaneously articulate (atomic) strokes with different shapes (see Chapters 3 and 4). Figure 4.3 provides an illustration of various articulation patterns captured from participants in our studies when asked to produce a square. For such articulations, extracting the atomic strokes (*e.g.*, the four strokes that make up the square) is a challenging task, and techniques like those discussed in [Jiang 2012] for reusing the $1 recognizer [Wobbrock 2007] on multi-touch gestures are not applicable when fingers move in parallel (*e.g.*, each finger drawing half the square as in Figure 4.3c).

As $P does not depend on the notion of stroke [Vatavu 2012c], a potential approach to recognize multi-touch gestures would be to directly apply the matching technique of $P on the point clouds resulted from sampling the multi-touch input. However, we argue that such an approach has limitations endorsed by the very nature of multi-touch gesture articulation. For example, Figure 6.1 shows a specific situation in which two distinct gesture types, "spiral" and "circle", have similar point cloud representations because of additional points introduced in the cloud when employing multiple fingers. This limitation was confirmed by running the $P recognizer on a set of 22 multi-touch gesture types, for which the recognition accuracy, between 82% and 98%, was lower than in previous works that evaluated $P on similar, but single-touch gestures [Vatavu 2012c] (we provide complete details on the evaluation procedure later in this chapter). The accuracy problem could be partially handled by sampling more points from the multi-touch input that would provide more resolution for representing gestures and, potentially, more opportunity for $P to discriminate between different gesture types. However, previous works have shown that the performance of many gesture metrics[1] does not necessarily improve when more sampling resolution is available [Vatavu 2011a, Vatavu 2012c]. Also, increasing the size of the point cloud increases the time required to compute the $P cost function, which depends quadratically with the sampling rate $O(n^{2.5})$ [Vatavu 2012c] (p. 278). Consequently, such an approach may prove problematic for low-resource devices that need particular attention in terms of dimensionality representation in order to be able to sense and recognize gestures with real-time performance [Vatavu 2013a].

6.2 Match-Up & Conquer

We have shown that the point cloud representation of gestures can affect the performance of the $P recognizer for scenarios in which multi-touch input is required by the application or is simply employed by users. Therefore, we identify the problem of constructing proper point cloud representations for multi-touch input to leverage the robustness of today's gesture recognition techniques that operate under unconstrained user gesture articulation, such as $P [Vatavu 2012c]. In the following, we describe the M&C technique that addresses this problem.

6.2.1 Match-Up: Preprocessing For Multi-Touch Input

The first step of the M&C technique consists in running a clustering procedure to group touch points that belong to finger movements that are similar in direction and path shape. Two finger movements are considered similar if they are produced simultaneously and are relatively "close" to each other. The goal of this procedure is to identify the atomic strokes

[1] As $P was not covered by these works [Vatavu 2011a, Vatavu 2013a], we cannot estimate its behavior with increased sampling resolution. However, the *peaking phenomenon* was often observed in the pattern recognition community, according to which adding more features up from one point does not improve, but actually increases classification error [Sima 2008].

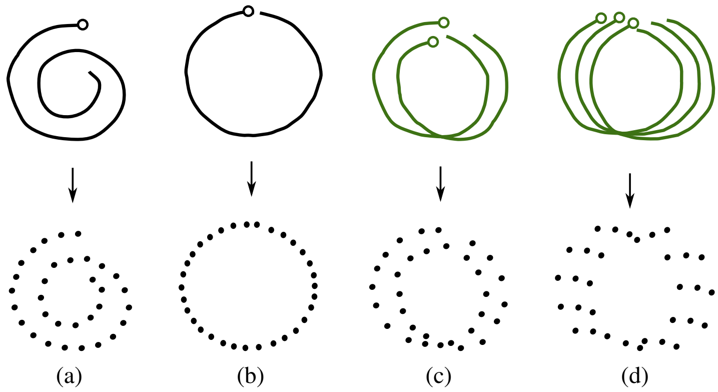


Figure 6.1: Adopting a direct point cloud representation for gestures produced with multiple fingers can lead to situations in which different classes of gestures have similar point clouds, such as the "spiral" (a) and the two-finger "circle" (c), which was not a problem before for single-touch input (b). More fingers lead to point clouds that are even more problematic to discriminate (d vs. a).

which are uniquely identifiable movements in the multi-touch gesture (see Figure 6.2). An atomic stroke may be composed of one stroke/trace only (*i.e.*, only one finger touches the surface), or it may consist in multiple individual strokes when multiple fingers touch the surface simultaneously and they all move in the same direction and follow the same path shape.

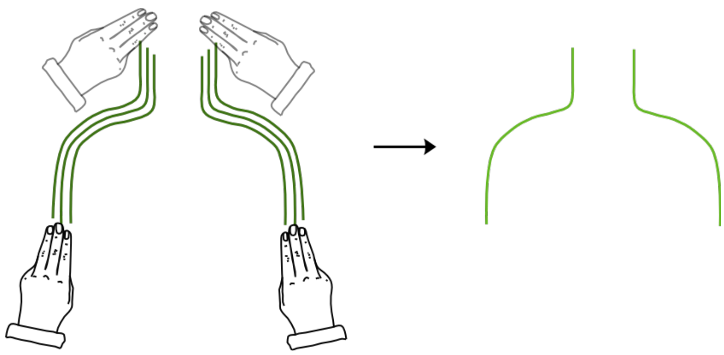


Figure 6.2: A multi-touch gesture articulated with two hands and six fingers (left) that has two atomic strokes (right).

We describe a touch point p by its 2-D coordinates and timestamp, $p = (x_p, y_p, t_p) \in \mathbb{R}^3$. A multi-touch gesture is represented by a set of points, $\mathscr{P} = \left\{p_i = \left(x_p^i, y_p^i, t_p^i\right) | i = 1..n\right\}$. The displacement vector of point p between two consecutive timestamps $t_{i-1} < t_i$ is defined as $\vec{D}_p^i = \left(x_p^i - x_p^{i-1}, y_p^i - y_p^{i-1}\right)$ and the angle between vectors $\vec{D}_p^i$ and $\vec{D}_q^i$ of points p and q is given by:

$$\theta_{p,q} = \arccos\left(\frac{\vec{D}_p^i \cdot \vec{D}_q^i}{\|\vec{D}_p^i\| \, \|\vec{D}_q^i\|}\right) \tag{6.1}$$

We consider two points p and q as part of two similar finger movements ($p \approx q$) if their displacement vectors are approximately collinear and p and q are sufficiently close together:

$$p \approx q \Leftrightarrow \theta_{p,q} \leq \varepsilon_\theta \text{ and } \|p-q\| \leq \varepsilon_d \tag{6.2}$$

where $\|p-q\|$ represents the Euclidean distance between points p and q, and ε_θ and ε_d are two thresholds.

With these considerations, we describe our clustering procedure that groups together touch points of similar strokes at each timestamp of the articulation timeline. The procedure is based on the agglomerative hierarchical classification algorithm [Webb 2002] (p. 363):

1. Construct clusters for each touch point p available at timestamp t, $\mathscr{C}_j = \{p \in \mathscr{P} \mid t_p = t\}$. Initially, all $|\mathscr{C}_j| = 1$. If a touch is detected for the first time, delay its cluster assignment until next timestamp.
2. For each pair of clusters $(\mathscr{C}_j, \mathscr{C}_k)$, compute their minimum angle $\theta_{j,k}$=min$\{\theta_{p,q} \mid p \in \mathscr{C}_j, q \in \mathscr{C}_k\}$ and minimum distance $\delta_{j,k}$=min$\{\|p-q\| \mid p \in \mathscr{C}_j, q \in \mathscr{C}_k\}$.
3. Find the pair of clusters $(\mathscr{C}_j, \mathscr{C}_k)$ for which $\theta_{j,k}$ and $\delta_{j,k}$ satisfy equation 6.2 and $\theta_{j,k}$ is minimized.
4. If no such pair exists, stop. Otherwise, merge $\mathscr{C}_j$ and $\mathscr{C}_k$.
5. If there is only one cluster left, stop. Otherwise, go to 2.

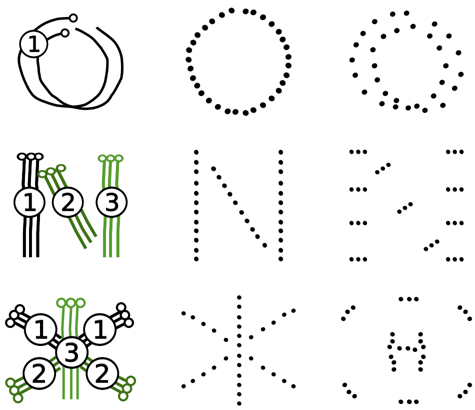
The result of the clustering process clearly depends on ε_θ and ε_d, for which we provide a detailed analysis later in this chapter.

Once clusters are computed for each timestamp t_i, they are analyzed to derive the strokes of the multi-touch gesture. To consistently group clusters into atomic strokes, we track cluster evolution over time and assign clusters of time t_i with the same stroke identifiers used for the previous timestamp $t_{i-1}<t_i$. If no clusters exist at t_{i-1}, all the clusters are assigned new stroke identifiers, which corresponds to the case in which users touch the surface for the first time or when they momentarily release the finger from the surface. Otherwise, we examine the touch points of every cluster and compare their point structure between times t_{i-1} and t_i. Each cluster $\mathscr{C}_i$ at moment t_i will take the identifier of a previous cluster $\mathscr{C}_{i-1}$ at time $t_{i-1}<t_i$ if the following conditions are met: (i) there exists a subset of points from $\mathscr{C}_i$ that also appears in $\mathscr{C}_{i-1}$, *i.e.*, $\mathscr{C}_i \cap \mathscr{C}_{i-1} \neq \emptyset$, (ii) all the other points of $\mathscr{C}_i$ appeared for the first time at moment t, and (iii) all the other points from $\mathscr{C}_{i-1}$ were released from the surface. Otherwise, a new stroke identifier is assigned to $\mathscr{C}_i$. The result of this process is a set of clusters $\left\{\mathscr{C}_i^k \mid k = 1..K\right\}$ that reflect the atomic strokes of the multi-touch gesture (*e.g.*, K=2 for the example in Figure 6.2).

Pseudocode. We provide pseudocode for the Match-Up to assist practitioners in implementing Match-Up into their gesture interface prototype. Technical specifications are described in Appendix A – "Match-Up & Conquer Pseudocode".

6.2.2 Conquer: Recognizing Multi-Touch Gesture Input

By computing clusters of points that belong to similar strokes, we extract a representative set of points for each atomic stroke. To do that, we use the trail of centroids across all



(a) (b) (c)

Figure 6.3: Gesture resampling with the Match-Up step (b) compared to direct resampling (c) for several symbols in our set (a).

timestamps t for clusters that were assigned the same stroke identifier $k \in \{1..K\}$:

$$c_{k,t} = \frac{1}{|C_k|} \sum_{p \in C_k} p \tag{6.3}$$

with atomic stroke k represented by the set $\{c_{k,t}\}$. Atomic strokes are normalized and resampled with standard gesture preprocessing procedures [Anthony 2010, Vatavu 2012c, Wobbrock 2007]. The resulted representation is then fed into a gesture recognizer, which is the $P recognizer [Vatavu 2012c] in this work. Figure 6.3 shows the result of the point resampling procedure before and after running the Match-Up technique for three multi-touch gestures from our set: "circle", "N", and "asterisk".

Match-Up & Conquer Library. We make available the Match-up & Conquer c++ library[2] distributed under the LGPL version2 license agreement.

6.3 Experiment Design

To investigate and understand the effectiveness of our stroke clustering step (Match-Up), we conducted a recognition experiment, in which we compared the M&C technique employing the $P recognizer in the second step (Conquer) vs. the unmodified $P [Vatavu 2012c]. We chose to use $P as baseline as previous works found it to be the most accurate $-like recognizer for classifying multi-stroke gesture input under various conditions [Anthony pear, Vatavu 2012c]. We made the evaluation on the dataset collected in the user study of Chapter 4. We chose to use this dataset because the collected data contain unconstrained multi-touch gesture that are proposed by the participants without any constraint or guideline that can influence their choice. Overall, the dataset represents gestures of 22 distinct gesture types and contains 5,155 total samples.

[2] https://sites.google.com/site/yosrarekikresearch/projects/matchup

We then employ several ways of selecting the training data for the recognition engine where the first matter is to determine the most effective method of choosing a minimal subset of data that yields the best unconstrained bimanual and multi-touch-independent performance. These methods all involve randomly drawing from the entire available pool of training data with varying enforcements over the selection. Overall, four mains experiments are conducted. Two experiments are based on the training of the user independently on the gesture articulations. The two others experiments are attached to the gesture articulation class. As the results of the Match-Up step depend on the values chosen for the ε_θ and ε_d thresholds (see equation 6.2), we conducted a preliminary study to determine the optimal values of these parameters, which we found to be $\varepsilon_\theta = 30°$ and $\varepsilon_d = 12.5\%$. Later in this chapter we discuss in detail the impact of these two parameters on the performance of M&C. In the following we provide a detailed description of each set of experiment scenarios.

6.3.1 User Training

User Dependent Training	**User Independent Training**
16 participants $\times$	15 values for P $\times$
9 values for T^b ; $T = 1..9$ $\times$	100 repetitions for each P^a $\times$
100 repetitions for each T $\times$	4 values for T^b ; $T = 1..9$ $\times$
4 values for n ;$n \in \{8, 16, 32, 64\}$ $\times$	100 repetitions for each T $\times$
22 gestures $\times$	22 gestures $\times$
$\approx 2.5 \times 10^6$ **recognition tests**	$\approx 2.6 \times 10^7$ **recognition tests**

Table 6.1: Controlled variables for the User Dependent Training experiment (left) and Controlled variables for the User Independent Training experiment.

In these experiments, we follow the same experiment methodology as in [Vatavu 2012c] (p. 275) by running both user-dependent and user-independent tests on our dataset as summarized in Table 6.1 and detailed in the following paragraphs.

User Dependent Training. In this scenario, recognition rates are computed individually for each participant. For each gesture type, T samples are randomly selected for training and one other sample is selected for testing. The selection process is repeated 100 times for each T in order to compute an average recognition rate per participant and gesture type. We also controlled the size of the point cloud (n) needed for the matching heuristic of $P. We report results from $\approx 2.5 \times 10^6$ recognition tests by controlling the following factors: (1) the recognition condition, M&C with $P vs. unmodified $P, (2) the number of training samples per gesture type, $T = 1..9$, and (3) the size of the point cloud, $n = 8, 16, 32$, and 64 points (see Table 6.1, left).

User Independent Training. In this scenario, we compute recognition rates for each gesture type by employing data from different participants. For each gesture type, P participants are randomly selected for training and one other participant for testing. For each training participant and each gesture type, T samples are randomly selected for training. One sample for each gesture type is selected from the testing participant. The selection process is repeated 100 times for each P and T in order to compute an average recognition rate per gesture type. The size of the point cloud was $n = 32$ points. We report results from $\approx 2.6 \times 10^7$ recognition tests by controlling the following factors: (1) the recognition condition, M&C with $P vs. unmodified $P, (2) number of training participants, $P = 1..15$, and (3) number of training samples per gesture type, $T = 1..4$ (see Table 6.1, right).

6.3.2 Gesture Class Testing

Gesture-class-dependent testing	**Gesture class Independent Testing**
16 participants $\times$	16 participants $\times$
4 values for T^b ; $T = 1..4$ $\times$	4 values for T^b ; $T = 1..4$ $\times$
100 repetitions for each T $\times$	100 repetitions for each T $\times$
22 gestures $\times$	22 gestures $\times$
4 classes $\times$	4 classes $\times$
$\approx 1.12 \times 10^6$ **recognition tests**	$\approx 1.12 \times 10^6$ **recognition tests**

Table 6.2: Controlled variables for the Gesture class Dependent Testing (left), and Controlled variables for the Gesture class Independent Testing (right) experiments.

To investigate further the importance of our Match-Up technique to harness gesture variability, we divided gestures into four classes according to (1) the number of fingers employed per atomic stroke (*i.e.*, one or multiple) and (2) the relative order of stroke articulation (sequential or parallel), as per the taxonomy proposed in the chapter 4. We took this approach as we hypothesized large number of points resulting from multi-finger articulations cause $P to deliver decreased performance just because point clouds are not representative enough (see the “circle” and “spiral” in Figure 6.1). Please note that for strokes synchronization we use only two variations sequential or parallel. Sequential means that strokes are produced in a row. Parallel means that some strokes (typically two) are drawn by different fingers at the same time. In case a gesture articulation contained both parallel and sequential strokes, we considered it part of the parallel classes as our goal is to evaluate if our recognizer still have good performance when the gesture articulation contains synchronous parallel articulation. We then identified four classes: SS (Single-touch Sequential), SP (Single-touch Parallel), MS (Multi-touch Sequential), and MP (Multi-touch Parallel). Recognition rates were computed for each gesture class under the two following conditions.

Gesture Class Dependent Testing. In this scenario, recognition rates are computed individually for each participant. Training samples for each gesture type are selected independently of class, while testing samples from one class at a time, *e.g.*, test with single and multi-finger gestures separately. The selection process is repeated 100 times for each T in order to compute an average recognition rate per participant and gesture type. For these tests we report recognition rates for T in $\{1,2,3,4\}$ because not all participants produced gestures in all classes for all symbols. However, participants did produce at least five samples for each variation they proposed. Also, some symbols were omitted in the case there were not enough samples to support this requirement (*e.g.*, gestures from the parallel classes were rarely produced for symbols that are not symmetrical in shape, such as "spiral", "S", "P", etc.). The size of the point cloud is fixed to $n = 32$ points. We report results from $\approx 1.12 \times 10^6$ recognition tests by controlling the following factors: (1) the recognition condition, M&C with $P vs. unmodified $P, (2) the number of training samples per gesture type and $T = 1..4$, and (3) the gesture class condition: SS, SP, SP and MP (see Table 6.2, left).

Gesture Class Independent Testing. In this scenario, recognition rates are computed individually for each participant. Training samples for each gesture type are selected from one class at a time, while testing samples are selected independently of the class, *e.g.*, we train with single-finger articulations, but test with both single and multi-finger articulations. The selection process is repeated 100 times for each T in order to compute an average recognition rate per participant and gesture type. For these tests we report recognition rates for T in $\{1,2,3,4\}$ because not all participants produced gestures in all classes for all symbols. The size of the point cloud is fixed to $n = 32$ points. We report results from $\approx 1.12 \times 10^6$ recognition tests by controlling the following factors: (1) the recognition condition, M&C with $P vs. unmodified $P, (2) the number of training samples per gesture type and $T = 1..4$ and (3) the gesture class condition: SS, SP, SP and MP (see Table 6.2, right).

6.4 Recognition Performance

In this section, we report the recognition accuracy results for the four experiments described in the previous section. Generally speaking, it was found that using Match-Up & Conquer improve significantly recognition accuracy compared to $P: the influence of the Match-Up step was larger for smaller point clouds (9.4% for $n = 8$) and smaller for more fine-grained representations of the point clouds (0.9% for $n = 64$). Our technique, also, provides higher accuracy for gestures that contain multi-touch atomic strokes that are articulated in sequential or in parallel and no difference was found for uni-touch gestures. A comprehensive analysis is given in the following.

6.4.1 User-Dependent Training

In this scenario, results showed an average recognition accuracy of 90.7% for M&C, significantly larger than that delivered by the $P recognizer running without our clus-

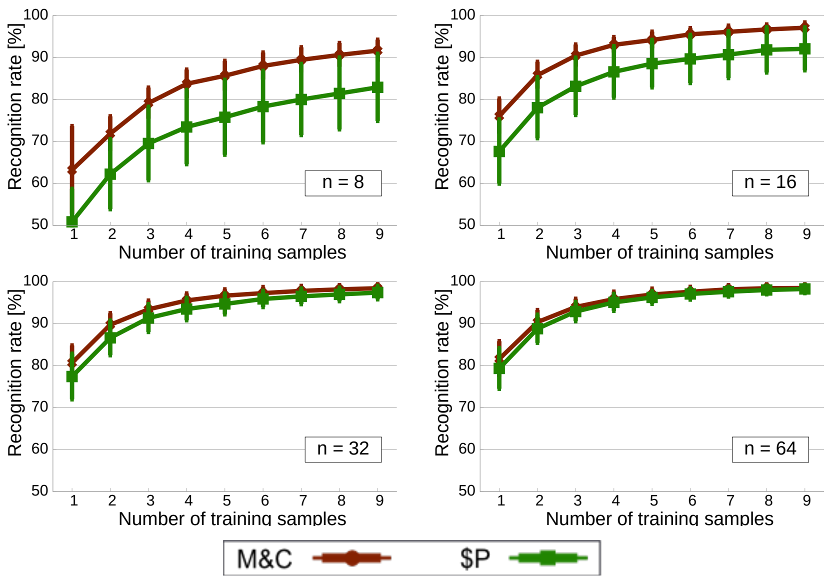


Figure 6.4: Recognition rates for M&C vs. unmodified $P.

tering step, which was 86.1% ($Z = 5.23$, $p < .001$). For $n = 32$ points (recommended in [Vatavu 2012c]), M&C delivered 95.5% accuracy with 4 training samples per gesture type and reached 98.4% with 9 samples, while $P delivered 93.5% and 97.4% under the same conditions. Figure 6.4 illustrates the influence of both number of training samples T per gesture type and the size of the cloud n on recognition accuracy. M&C was significantly more accurate than $P for all pairs of $T \in \{1...9\}$ and $n \in \{8, 16, 32, 64\}$ ($p < .05$), except for $T = 9$ and $n = 64$.

6.4.2 User-Independent Training

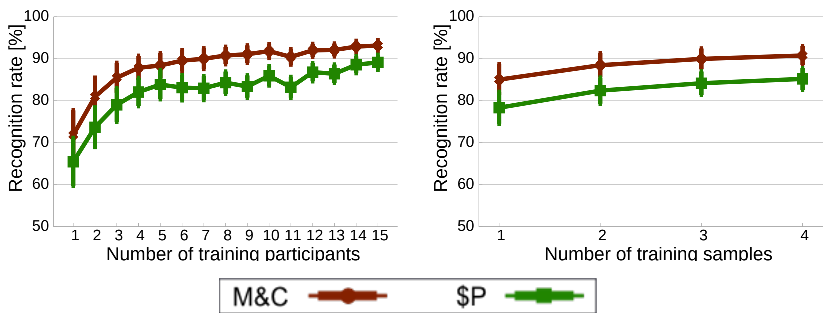


Figure 6.5: Recognition rates for the user-independent scenario. NOTE: $n = 32$ points; error bars show 95% CI.

Results showed M&C significantly outperforming the $P recognizer running without the clustering preprocessing step, with 88.9% vs. 82.5% ($Z = 3.4$, $p < .001$). For the maximum number of tested participants ($P = 15$) and training samples ($T = 4$), M&C achieved 94.2%, significantly higher than 90.8% delivered by $P ($Z = 8.14$, $p < .001$). The

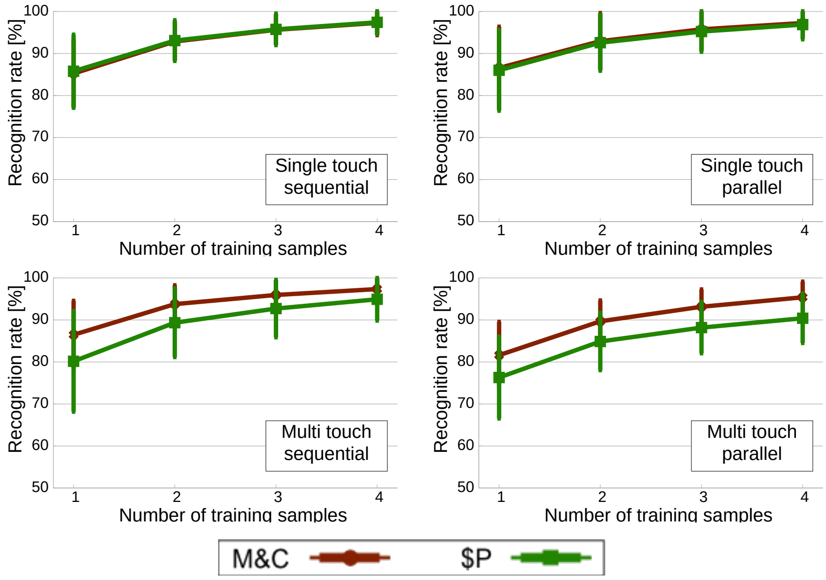


Figure 6.6: Recognition rates for gesture-class-dependent testing. NOTE: Rates are averaged for all gesture types and participants; $n = 32$ points; error bars show 95% CI.

number of training participants had a significant effect on recognition accuracy ($\chi^2(14) = 55.82$, $p < .001$), with M&C delivering higher recognition rates than $P (Figure 6.5, left). As expected, both the performance of M&C and $P improved as the number of training participants increased, from 71.86% and 65.46% with one training participant to 93.14% and 89.15% respectively with 15 training participants. The number of training samples per gesture type also had a significant effect on recognition accuracy ($\chi^2(3) = 45$, $p < .001$), with both M&C and $P delivering higher rates with more samples: 85.0% and 78.3% with one sample to 90.8% and 85.2% respectively with 4 samples per gesture type (Figure 6.5, right).

6.4.3 Gesture Class Dependent Testing

Figure 6.6 shows the recognition performance of M&C and $P under the gesture class dependent testing condition. Results showed M&C more accurate than $P for cases in which participants articulated gestures with multiple fingers: 93.4% vs. 89.2% for the multi-touch sequential class ($Z = 6.23$, $p < .001$), and 89.9% vs. 84.9% for multi-touch parallel ($Z = 5.35$, $p < .001$). Results were not significant any longer for the other two classes involving single-finger gesture articulations: 92.7% vs. 92.9% for single-finger sequential and 93.1% vs. 92.6% for single-finger parallel, *n.s.*

6.4.4 Gesture Class Independent Testing

Figure 6.7 shows the recognition performance of M&C and $P under the gesture-class-independent testing condition. Results showed M&C more accurate than the unmodified $P for three out of four classes: 87.4% vs. 84.5% for SS ($Z = 5.98$, $p < .001$), 87.1% vs.

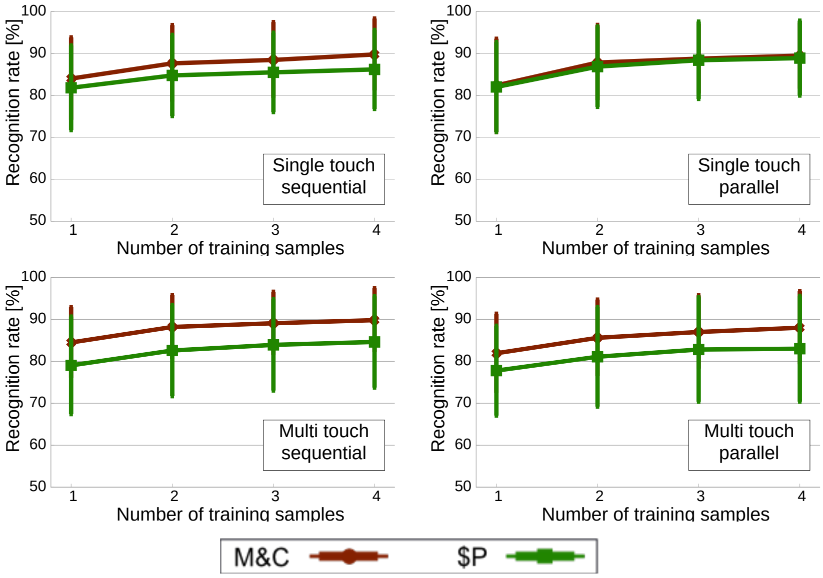


Figure 6.7: Recognition rates for gesture-class-independent testing. NOTE: Rates are averaged for all gesture types and participants; $n = 32$ points; error bars show 95% CI.

86.5% for SP (*n.s*), 87.8% vs. 82.5% for MS ($Z = 6.07$, $p < .001$), and 85.6% vs. 81.1% ($Z = 2.94$, $p < .005$).

6.4.5 Gesture Recognition per Participant

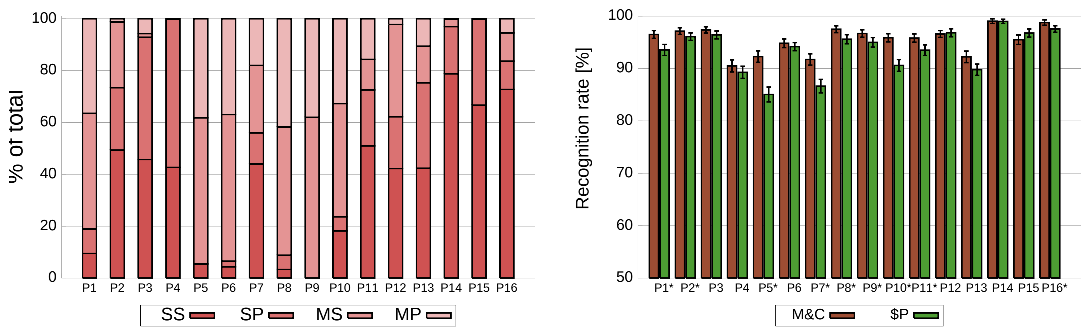


Figure 6.8: Distribution of gesture classes (top) and recognition rates per participant (bottom). Note how the recognition rates for M&C are particularly higher for participants that employed multiple fingers. NOTE: An asterisk * next to participant number shows difference is significant at $p < .05$; error bars show 95% CI.

As we found that M&C scored higher for participants that articulated gestures with more fingers, we investigated this finding further. Figure 6.8 illustrates the percentage of gesture classes *per* participant, as well as the corresponding recognition rates computed for $T = 4$ and $n = 32$. The figure reveals the fact that recognition rates for M&C are higher than those delivered by the unmodified $P for those participants who employed more fingers during gesture articulation (*i.e.*, higher percentage for classes MS and MP). A Spearman test confirmed this observation, showing significant correlation between the difference in recognition rates of M&C and $P (for all T) and the percentage of the MS and MP classes employed by participants ($\rho = .630$, $p < .001$).

The results of these 3 last subsections confirm our hypothesis that multi-finger gestures with more points (from more fingers being used) significantly influence the resampling step of $P when constructing the point cloud, which affects recognition performance. At the same time, M&C manages to group similar strokes together, and constructs a representative point cloud for multi-touch gestures suited for the $P classification step. Results, also, indicate that it is recommended to include in the training samples gestures from different classes in order to achieve better recognition accuracy.

6.5 Discussion

In this section we present additional details on the recognition performance of M&C, and discuss gestures that are confusable for our technique as well as reasons why this is so. We also point to future work ideas, such as providing users with on-line feedback on the atomic stroke extraction process.

6.5.1 Effect of Match-Up Parameters

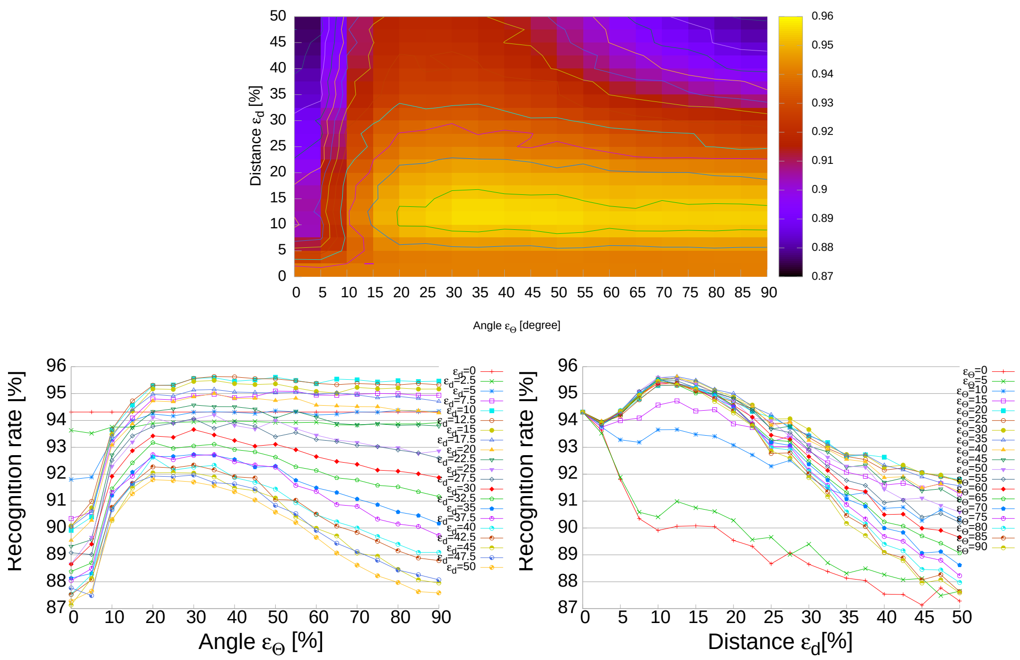


Figure 6.9: Effect of parameters ε_θ and ε_d on the recognition accuracy of M&C. Combined (top) and individual effects (bottom).

We mentioned before that the atomic stroke extraction results of the Match-Up technique depend on the values of the ε_θ and ε_d parameters of equation 6.2. The recognition experiment reported in the previous section employed optimal values for these parameters that we found to maximize recognition accuracy for our set of gestures, *i.e.*, ε_θ=30° and ε_d=12.5%. In this section we present an in-depth analysis of the effect of these parameters on the recognition performance of M&C, for which we ran the user-dependent recognition experiment and controlled the values of ε_θ and ε_d (T=4 training samples per gesture type and n=32 points were used this time to keep the running time manageable). We varied the values of the angle ε_θ in the range $[0,90]$ by a step of 5°. For ε_d, we normalized the Euclidean distance to the input area and varied ε_d in the range $[0,50]$ by 2.5%. Overall, we report results for the user-dependent recognition accuracy of M&C under $19 \times 21 = 399$ different combinations for ε_θ and ε_d in Figure 6.9.

As expected, accuracy is low when the angle parameter ε_θ is small ($\leq$10°), which makes nearby to be incorrectly assigned to distinct clusters. As ε_θ increases, points are clustered correctly, even though they do not follow a perfectly collinear stroke path. For $\varepsilon_\theta \geq 20°$, recognition accuracy is impacted by the distance parameter, ε_d. For example, when ε_d is low ($\leq$5%), points are incorrectly considered as individual clusters. As ε_d increases, points that are both collinear and close together are more likely to be clustered correctly. For $\varepsilon_d \geq$15%, recognition rates start to decrease significantly (p<.001). This result is explained by the fact that distant points that move in approximately the same direction can be incorrectly clustered together for large ε_d values.

The highest recognition accuracy was obtained for $\varepsilon_\theta \geq$30° and $\varepsilon_d \in [10,15]$. For ε_θ=30° and ε_d=12.5%, we observed that stroke extraction matched the structure of the articulated gesture. These values were therefore selected to evaluate the M&C technique in the previous section. Note however that the optimal values might need slight adjustment for other datasets. However, we estimate $[10,45]$ and $[10,15]$ as safe intervals for ε_θ and ε_d.

6.5.2 Confusable Gestures

Confused pairs	Occurrence	Gestures	Occurrence
Circle × Square	0.56%	Square	0.65%
N × H	0.28%	Circle	0.58%
Circle × Heart	0.28%	N	0.32%
D × Circle	0.24%	D	0.31%
X × Asterisk	0.23%	X	0.28%

Table 6.3: Top-5 most confusable pairs (left) and gestures (right). NOTE: user-dependent testing, T=4 samples, and n=32 points.

We found gesture type to have a significant effect on recognition accuracy ($\chi^2(21)$=113.81, p<.001), with some gestures exhibiting lower recognition rates. Re-

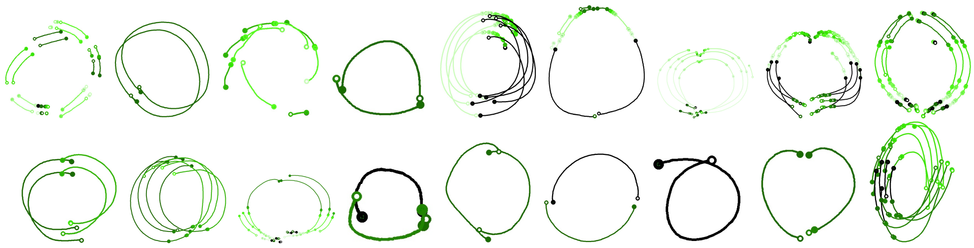


Figure 6.10: Several articulations for the “circle” symbol captured in our dataset.

member that we did not constrain participants to mind the number of fingers, nor did participants follow any training procedure before entering gestures. As a result, our multi-touch dataset includes versatile gestures that expose the intrinsic variability of articulating multi-touch input, and, consequently, some of the articulations are challenging to recognize. For example, Figure 6.10 illustrates several ways in how participants articulated circles. Table 6.3 shows the top-5 most frequently misclassified gesture pairs by M&C. As the stroke structure of these gestures is similar, small deviations in their articulations likely leads to recognition conflicts. However, maximum error rates were 0.56% for confused pairs and 0.65% for individual gestures.

6.5.3 Execution Time

The execution time of M&C is composed of the time required to identify atomic strokes (Match-Up) and the time to run the recognizer (Conquer). In practice, the execution time of the Match-Up step averaged over all participants and gesture types from our dataset was 22.8 ms (measured on a Intel® Xeon® CPU 2.67 GHz), which makes the Match-Up technique suitable for real-time processing.

6.5.4 On-line Computation of Match-Up

According to the definition of the clustering procedure, the Match-Up technique may run *on-line* during actual articulation. In fact, the procedure only needs to keep track of points detected at the previous timestamp and their corresponding cluster IDs to compute identifiers for current points. The on-line computability feature of Match-Up has interesting implications. First, atomic strokes may be computed in real-time and, thus, allows us to elicit the articulation type the user is performing (*e.g.*, Match-Up can detect that two strokes are being performed at the same time when the user performs a gesture with two parallel movements). This goes beyond gesture recognition toward discriminating between gesture variation classes. Second, computing multi-touch strokes at runtime opens new interaction possibilities. For instance, Match-Up makes it possible to deliver users with a feedback by displaying the stroke movements made by fingers as they occur. Such feedback may serve to help users learn gestures in a flexible and consistent manner.

6.5.5 Any Gesture Recognizer

In this work we employed the $P recognizer [Vatavu 2012c] in the second step (Conquer) of M&C and showed how Match-Up improves its classification accuracy for multi-finger gestures. We note however that any other recognizer may be used in conjunction with Match-Up. As a result, we deliver Match-Up as a new add-on to the practitioners' toolkit of gesture processing techniques, leading toward new algorithmic knowledge for processing multi-touch gestures.

6.6 Chapter Summary

We described in this chapter a simple, two-step technique for recognizing multi-touch gestures under unconstrained user articulation behavior. Through our technique we introduced, for the first time in the gesture literature, a preprocessing technique specific for multi-touch gestures that clusters similar finger strokes. By doing so, this technique consistently structures the shape of a multi-touch gesture. The result in the form of a multi-touch point cloud is fed into the $P recognizer [Vatavu 2012c], leading to improved recognition accuracy of multi-touch gestures in a manner that is independent of how users actually articulate gestures (*e.g.*, using one or two hands, one or multiple fingers, synchronous or asynchronous stroke input, and even erroneous input in the form of finger slips. This improvement has been proved through a rigorous experiment that test the accuracy of our technique under different conditions. In the future, other recognition techniques may be found or built to perform even better in general or on this corpus of gestures, and as such, the methodology used in this chapter will advance our algorithmic knowledge for multi-touch gestures and can lead to the design of more efficient and accurate recognizers for touch sensitive surfaces.

"Nothing happens until something moves."

Albert Einstein – Theoretical physicist and
philosopher of science (1879 – 1955)

7
Rigid-Movement based Interaction

All along this dissertation we advocated for a better understanding of users' gestures as well as for robust tools to deal with the versatility of multi-touch gestures. The technique provided in the previous chapter is in line with the efforts made by the community in order to bridge the gap between multi-touch gesture definition and users behavior and preferences in how to effectively articulate a gesture. Besides, a multitude of guidelines are being developed in order to sustain this general interaction principle from different perspectives, *e.g.*, gesture reproducibility, learning, reusability, etc. This chapter is devoted to explore new and alternative interaction techniques raised by the ability of users in representing and producing multi-touch gestures.

Gesture-based applications often aim at being as close as possible to the way people act in their surrounding environment [Nacenta 2009]. However, implementing flexible multi-touch gestural interaction requires to satisfy a difficult compromise between the desires and capabilities of users, and constraints of the machine. In order to limit ambiguities and misinterpretations, this generally leads to simplistic gesture commands or otherwise to gestures that are difficult to learn and to execute. For users, multi-finger gestural interaction can lack flexibility when imposing constraints they do not understand or do not always know how to respect. Hence, increasing the flexibility of interaction while preserving its technical efficiency is clearly one of the challenges posed by multi-finger touch interfaces. For instance, many interactions techniques on touchpads, screens or multi-finger touch tables are based on basic predefined gestures (*e.g.*, swipe, pinch, rotate) or on free movements. However, the number of fingers often occurs very early in the definition and the classification of gestures and it is often implemented in an ad-hoc manner. The trajectory of fingers also appears at the basis of modern interaction techniques. This induces unavoidable constraints and limitations in the sense that it requires the users to provide the system with a strict representation of the gesture. The idea developed in this chapter is motivated by the general goal of pushing a step toward riding multi-touch gestural interaction from such issues.

Basically, the corner stone of the work presented in this chapter is built on the collective and unconscious ability of users to move their fingers over an interactive surface in a very specific, non-random, and yet free, fashion. In previous chapters, we were able to observe that users have common behaviors when coordinating the movements of their fingers and their hands. For instance, a user can move several fingers in the same direction while simultaneously moving one or more other finger(s) in a different direction. The user studies conducted in Chapter 3 informs us that symbolic gestures are mainly conceived as the aggregation of several atomic movements that rise naturally from the users mental representation of the gesture. The idea developed in this chapter is to further explore the movement abilities of users at the aim of setting up and designing new flexible interaction tools. In fact, we argue that the free movements induced by fingers contain rich features that can be explicitly exploited to make interaction stronger. The major challenge that we are tackling can hence be formulated as follows:

- *What* kind of relevant information one could extract from touch movements?
- *How* it can be benefit in a gestural interaction context?

To address this challenge, we shall introduce the concept of *rigid-movement*. Like in the physical word, the concept of rigid-movements considers that contact points moving in the same way over a certain amount of time as if they were forming one unique coherent entity. This is seemingly inspired by the atomic movements emerging in our previous observations of users' behavior in conceiving multi-touch gestures. By providing a *system-oriented* definition of the concept rigid-movements and exploring their dynamics, we are able propose new gesture-based interaction tools which are applicable independently to how the user is effectively generating those rigid movements.

In the remainder, we first formalize the above idea and propose a set of enabling techniques allowing us to make it fully compatible with multi-touch systems. Then-after, we propose some design principles underlying rigid movements and we draw a comprehensive picture of the different interaction possibilities that they afford. To illustrate the effectiveness of our approach, the last parts of this chapter shall be devoted to describe and discuss several examples of techniques for direct and indirect multitouch interaction.

7.1 Rigid-Movement Concept and Tools

In classical approaches, a microscopic perspective is adopted when defining and processing multi-touch gestures, *i.e.*, every single and individual touch is constituting a unitary building block. We argue that this can introduce difficult constraints for users when articulating gestures; but also for designers and engineers when defining and effectively processing gestures. In line with the movement level elicited from the user study conducted in Chapter 3, we propose to adopt a macroscopic perspective when defining and processing multitouch gestures in an attempt to harness their global dynamics. For this purpose, we introduce the concept of *rigid-movement*. The term *rigid* is inspired by the physical world, in the sense that we shall view users' gestures as being induced by groups of fingers moving in an uniform and compact way, as if they were forming one physical object. In particular, such a perspective shall allow us to abstract multi-touch gestures from the number of fingers,

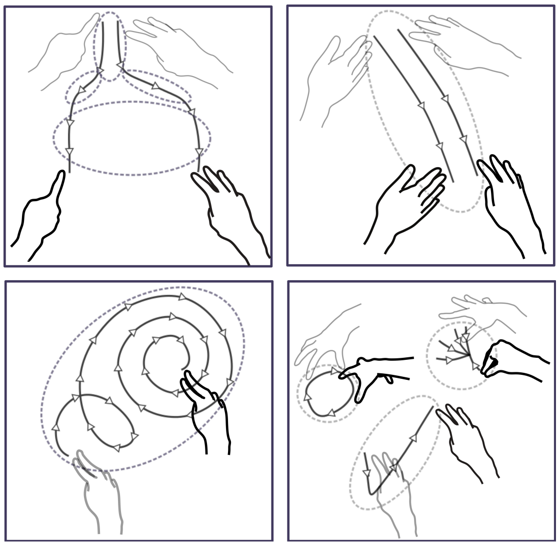


Figure 7.1: Rigid movement examples.

and also from the way the corresponding groups of fingers (possibly belonging to different hands/users) are formed.

7.1.1 Terminology

In our terminology, a rigid movement refers to an arbitrary number of touches making the same kind of movement between two consecutive time periods. More formally, we define a rigid movement as a group of contact-points moving at the same speed and making the same rotation, translation and scale in a given time period. This definition is fully guided by the *relative movements* performed by individual touches. It should be clear that our definition implies that rigid-movements can evolve dynamically at runtime depending on how touches (fingers) are moving over the surface as illustrated in Figure 7.1. From the birth to the death of one rigid movement, the corresponding touches can globally evolve in different manners (*e.g.*, direction, speed, shape, etc). However, between two consecutive time periods, they can only evolve in the same manner by the virtue of our definition.

Notice that our definition of rigid-movement differs from the standard notion of strokes in symbolic gestures by few subtile features. First, we consider that touches being far away one another could form the same rigid-movement. Second, rigid-movements are *not* with respect to a pre-determined shape or symbol. More importantly, rigid-movements are defined from a pure multi-touch system perspective although they directly relate to the user-study described in the beginning of this dissertation. In other words, they are our own interpretation and tentative formalization (as HCI researchers) of users' atomic movements.

7.1.2 Runtime Extraction of Rigid Movements

From our definition, rigid movements can only be extracted by finding similarities between the characteristics of multi-contact movements *dynamically at runtime* and without any a-prior knowledge on the gestures being performed or any other specific rule such as the shape or the size of contact points.

Knowing that interactive multi-touch surfaces typically provide basic information concerning individual contact-points and their dynamics (*e.g.*, identifier, position, etc), the first challenge is then to elicit rigid-movements at runtime based on this basic information. For that purpose, we shall process contact-points at successive time periods defined by successive timestamps denoted by t_i. It is worth-noticing that the frequency of this processing period can be less than the frequency at which the interactive system updates the states of detected contact-points. At each timestamp t_i, we address the following two questions:

- How to decide whether contact-points were moving in a rigid way (or not) in time period $[t_{i-1}, t_i]$? We recall that following our definition, contact points forming a rigid-movement should have the same speed and make the same rotation, translation and scale between t_{i-1} and t_i. To answer this question, we propose to compute an analytical estimation of the movement of contact-points. In the case the virtual positions approximated by this analytical estimation are close to the actual positions detected by the surface, contact-points should be considered as forming the same rigid-movement. Otherwise, we can conclude that contact-points are not moving in a rigid way and we can proceed with the next question.

- How to effectively extract the set of rigid movements? In fact, once we can decide that the whole contact points are *not* moving in a rigid way, we need to elicit what subgroups of contact points are separately doing so. To answer this question, we shall use a clustering-based procedure that classifies contact-points in a recursive manner according to the similarity of their estimated movements.

In the rest of this section, we describe in details the algorithmic components allowing us to answer these two questions, as well as the different enabling tools allowing us to fully harness the concept of rigid-movements for interactive multi-touch surfaces.

7.1.2.1 Movement Estimation

We first want to compute analytically how rigid is the movement of contact points given their positions between two timestamps t_{i-1} and t_i. For this purpose, we propose to estimate their movements using three basic geometric transformations: a translation, a rotation and a scale. Computing the optimal transformations is difficult and we shall instead consider approximated ones as described in the following.

The translation factor, denoted by T_i, is simply computed with respect to the center of mass of contact-points. It is given by the following equations (where P refers to the position of a contact point):

$$C_{i-1} = \frac{1}{n}\sum_k P_i^k \quad ; \quad C_i = \frac{1}{n}\sum_k P_i^k \quad ; \quad T_i = C_i - C_{i-1} \tag{7.1}$$

To approximate the rotation angle, we can minimize the squared rotation error function $\rho(\theta)$ defined as:

$$\rho_i(\theta) = \sum_{k=1}^{n} \left\| R(\theta) \cdot (P_{i-1}^k - C_{i-1}) + C_i - P_i^k \right\|^2 \tag{7.2}$$

where the $2D$ rotation matrix $R_i(\theta)$ has the following form:

$$R_i(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{7.3}$$

Hence, by a routine calculus and taking the derivative, the value θ_i^* that minimizes $\rho_i(\theta)$ must satisfy the following equation:

$$tan(\theta_i^*) = \frac{\sum_k det(\overrightarrow{C_{i-1}^k P_{i-1}^k}, \overrightarrow{C_i^k P_i^k})}{\sum_k \overrightarrow{C_{i-1}^k P_{i-1}^k} \cdot \overrightarrow{C_i^k P_i^k}} \tag{7.4}$$

where $det(u,v)$ is the standard notation referring to the determinant of two vectors, *i.e.*, $det(u,v) = u_x v_y - u_y v_x$. From Eq. (7.4) and by the properties of the *tan* function, two values of the rotation angle that we are seeking for can be extracted, namely, θ_i^* and $\theta_i^* + \pi$. We thus retain one of the two values that minimizes Eq. 7.2.

Having computed the translation and the rotation factors, it remains to compute the scale factor. For this, we can translate and rotate contact-points using T_i and R_i and minimize the squared scale error function defined as:

$$\sigma_i(s) = \sum_{\ell=1}^{n} \left\| S_i(s) \cdot R_i(\theta_i^*) \cdot (P_{i-1}^k - C_{i-1}) - (P_i^k - C_i) \right\|^2 \tag{7.5}$$

where $S^i(s)$ is a scale matrix and s the scale factor. Again, by a routine calculus, the value s_i^* that minimizes σ_i is given by:

$$s_i^* = \frac{\sum_k \overrightarrow{R_i(\theta_i^*) \cdot (C_{i-1}^k P_{i-1}^k)} \cdot \overrightarrow{C_i^k P_i^k}}{\sum_k \left\| \overrightarrow{R_i(\theta_i^*) \cdot (C_{i-1}^k P_{i-1}^k)} \right\|^2} \tag{7.6}$$

Using the previous equations, we are able to estimate the global translation, rotation and scale experienced by a set of contact-points between two timestamps t_i and t_{i-1}. As we will detail in the next section, movement estimation is the key ingredient that shall allow us to decide whether a group of contact-points is defining a rigid movement and to classify them accordingly.

7.1.2.2 Mouvement Classification

Our goal is to extract rigid-movements based on the runtime behavior of contact-points. For this purpose, we design a classification procedure that partition contact-points into several groups such that the movements of contact-points in a single group verify a high degree of similarity, and the global movements of contact-points being in different groups verify a high degree of dissimilarity. For the sake of clarity and in order to not overload the reader with unnecessary technicalities, we shall only sketch the major algorithmic components of our rigid-movement classifier.

At every timestamp t_i, we form an initial group containing all contact points, then we recursively refine this initial clustering using the following high-level procedure:

1. We compute the estimated translation, rotation and scale of a group as detailed in the previous section.

2. We apply these transformations to every contact-point P_{i-1} of a group observed at the previous timestamp t_{i-1}. This allows us to analytically estimate the position of P_{i-1} at current timestamp t_i, which we denote by $\widehat{P_i}$.

3. We evaluate how rigid is the movement of contact points belonging to a group by computing the average euclidian distance between $\widehat{P_i}$ and P_i. The idea is that if a group is forming a rigid movement, then its estimated position should not differ from its actual position.

4. If this distance is beyond a threshold, then we can conclude that the considered group does not imply a rigid-movement. In this case, we split it into two subgroups using the classical *k-means clustering algorithm* [1] [Tan 2006] with $k = 2$ and a similarity criterion taking into account the position and the velocity of contact-points.

5. A side effect of the previous step is to create several subgroups implying similar rigid-movements, eventually ending with each single contact point in one group. Hence, in this step, we manage to merge the subgroups having similar estimated movements (by applying the previously described geometric transformation).

6. The previous steps are repeated recursively until no new groups are split nor merged.

Because the above classification procedure uses as a key criterion the movement of contact points, it tends to group together contact points that did not move or those that have a very small velocity. However, informal tests have indicated that this corresponds very rarely to users' expectations. Hence, in the implementation of our classification procedure, we shall ignore contact points with this type of behavior. Moreover, it is not possible to estimate the movement of contact-points that are newly detected by the surface. This may typically happen because the user unintentionally holds his fingers from the surface for a very short period of time, or because the user put new fingers on the surface. Hence, to classify the newly detected contact-points, we proceed as follows. We consider the group (computed at the end of the above procedure) having the closest center of mass to a newly detected point. We compute the euclidian distance between the group center-of-mass and the newly detected point. If this distance is less than a threshold, the contact point is included to that group, *i.e.*, the contact point is considered as performing the same rigid

[1] http://en.wikipedia.org/wiki/K-means_clustering

movement than the closest group. Otherwise, the contact-point is considered as forming a single rigid-movement.

7.1.3 Rigid-Movements Life-story

The previously described procedure maps every contact points to one rigid movement at every timestamp. By keeping this mapping in the system memory, it becomes possible to keep track of the dynamics of rigid-movements. In other words, we are able to track the progress of the rigid-movements implied by contact-points over time and not only at a given timestamp. We propose to capture the life-story of a rigid-movements in a generic way by defining a set of events that provide basic information about computed movements at every timestamp. The proposed events are neither exhaustive nor prescriptive, in the sense that one could define other events reflecting different dynamics. We start with three basic events informing about a single rigid movement:

- *New* Event: This notifies the fact that a new group of contact points have been identified to create a new rigid movement at the current timestamp.

- *Update* Event: This notifies the fact that the properties of a whole set of contact-points belonging to a previously created group have changed, *e.g.*, their positions have changed, one or more new contact-points join the group, one or more contact-points quit the group, etc.

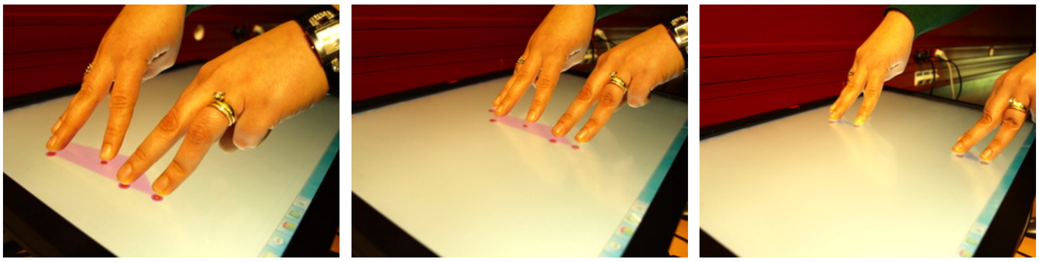
- *End* Event: This notifies the fact that a whole set of contact points forming a previously created group is no longer active on the interactive surface.

We further define two simple events corresponding to simple state transitions of different rigid movements. As we will see later, these very simple events reporting how rigid movements are acting/evolving one with respect to the others, are of special interest.

- *Merge* Event: This notifies the fact that at least two or more previously created groups of contact points have been merged into the same group, *i.e.*, they are now moving in the same rigid way.

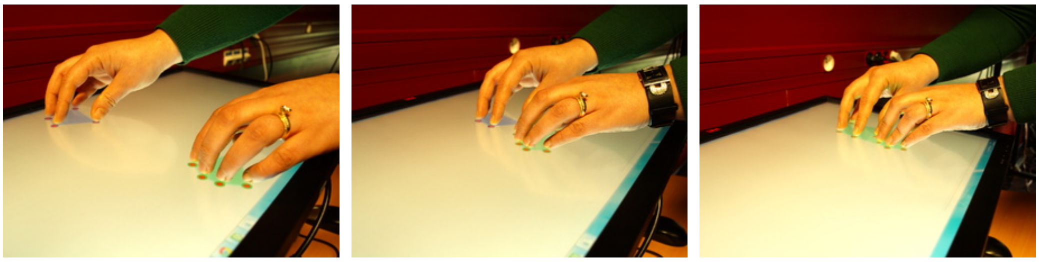
- *Split* Event: This notifies the fact that a group of contact points which were in the same group, have now stopped moving within the same rigid movement, *i.e.*, they are now forming a new set of rigid groups.

In Figure 7.2, we depict the typical scenario for producing a split event which corresponds to the user first moving some fingers of his two hands in the same direction, then moving each hand in a different direction. Similarly, a merge event would typically corresponds to the scenario of Figure 7.3 where a user starts by moving the fingers of each hand following different directions, then moving all the fingers again in the same direction.



(a) (b) (c)

Figure 7.2: Example of gesture producing a *Split* event: (a,b) depict one rigid movement evolving over time and (c) depicts its split into two different rigid movements.



(a) (b) (c)

Figure 7.3: Example of gesture producing a *Merge* event: (a,b) depict two rigid movements evolving independently over time and (c) depicts their merge into the same rigid movement.

7.1.4 Rigid Movements Secondary Parameters

It is important to remark that the key features needed to grasp the rigid-movement concept leave many other parameters uncontrolled and free to both users and designers. A first example is the number of contact points which is clearly not needed in anyway in order to define, perform nor extract rigid movements. A second example is the flexibility implied by the specification of the previously described events. In fact, we provides some specific examples of how split and merge can be achieved using two movements. But, these two specific movements can take different forms, speeds, directions, and can be initiated anywhere on the interactive surface using a variable number of fingers/hands. The user can feel free to produce them in many different ways if the designer does not impose any further constraint. For instance, using just two fingers of one hand by respectively moving the thumb and the forefinger close or far away would respectively produce a split or a merge event. Similarly, by moving the two hands following other global directions while being free to fix the number of fingers, the distance between hands, and/or the scale of the induced movements, the two events can perfectly be reproduced.

7.2 Interaction Design Space

The design space of interactive systems can be significantly influenced by the available techniques and technologies. We argue that our rigid-movement based approach enables to consider new possibilities when designing gesture-based applications for multi-touch interactive surfaces. In this respect, we propose three prospective guidelines that can be drawn from our approach in order to expand the range of design possibilities in a generic way. Several illustrative techniques assessing the validity of the investigated design space are delayed to later in this chapter.

7.2.1 Prospect #1: Rigid-Movement as a Self-Coherent Entity

Rigid-movements can be viewed as *"meta-touches"* with specific intrinsic properties. Our first guideline is to conceive interaction with respect to this *"meta-touch"* considered as a global and indivisible entity. First, the rotation, the scale and the translation that characterize each rigid movement are well identified components that can be specialized to deal with classical interactions tasks such as scrolling, pinching, rotating, etc. Second, users can be left free to use any number of fingers to produce a single rigid movement in a multi-touch gesture. This enables to abstract away the number of fingers and to increase the flexibility of gestural interaction. For instance, this can enable users to optimize the way they interact with the system, since they are not constrained by a pre-defined number of fingers; but instead, they can dynamically adapt the number of fingers at runtime depending on the interaction context. Third, the number, the shape and the area of fingers in a rigid movement can be arbitrary. This can be used to set up the secondary parameters of an interaction task. For instance, the global trajectory of a group of fingers (*e.g.*, their center of mass) can be used to define a global gesture, while the same gesture can be interpreted differently depending on how many fingers are being used or on the relative position of individual fingers.

7.2.2 Prospect #2: Rigid-Movements State Evolution

The state evolution of rigid movements provides a global runtime view of users' movements. We propose to define new gestural-based interaction actions on the basis of how each rigid movement is acting with respect to the other ones over time. In particular, we shall explore the possibility of introducing a new class of gestures by specifying simple semantic rules that should be verified by rigid movement state transitions. Such semantic rules can be easily defined on the basis of the life-story of rigid-movements as captured by the different events elicited in the previous section. For instance, a simple gesture could be defined by the transition from the state where one rigid movement is detected to the state where two rigid movements are co-evolving, *i.e.*, Split event.

As a straightforward consequence, the so-defined gestures would not require to use a pre-defined number of fingers, nor to move fingers following a pre-defined direction nor shape, but in such a manner a specific transition rule is fulfilled. This would imply that a particular rigid-movement state transition can activate an interaction task in possibly different manners anywhere on the interactive surface depending on users perception of the corresponding transition rules.

7.2.3 Prospect #3: Complementary Rigid Movements

Different rigid-movements can co-exist and complement one another in order to coordinate different subparts of the same interaction task. In particular, multiple rigid-movements should *not* necessary be considered with respect to multiple independent interaction tasks.

The coordination of different rigid-movements is related to the issue of matching the separability and integrality of different components of the same interactive task. The concepts of separability and integrality [Garner 1974, Jacob 1994, Nacenta 2009] are intended to capture the more general hypothesis that the performance of an interactive system is impacted by how the perceptual space of an interactive task is matched with the structural properties of the corresponding enabling technique. For instance, it has been proved in [Jacob 1994] that "*if the input device supports the type of motion required by the task, then the task can be performed in an efficient manner*". Rigid-movements can be used to identify and to emulate different types of motions upon the interactive surface. Complementary rigid-movements can thus enable to match the different dimensions/attributes of an interaction task.

Notice however that the coordination of several rigid-movements should not lead to a complex design which would be hard to enjoy by users. Hence, it should be easy for users to understand what every rigid movement is intended for and how the whole interaction task can be decomposed without introducing much complexity.

7.3 Illustrative Interaction Prototypes

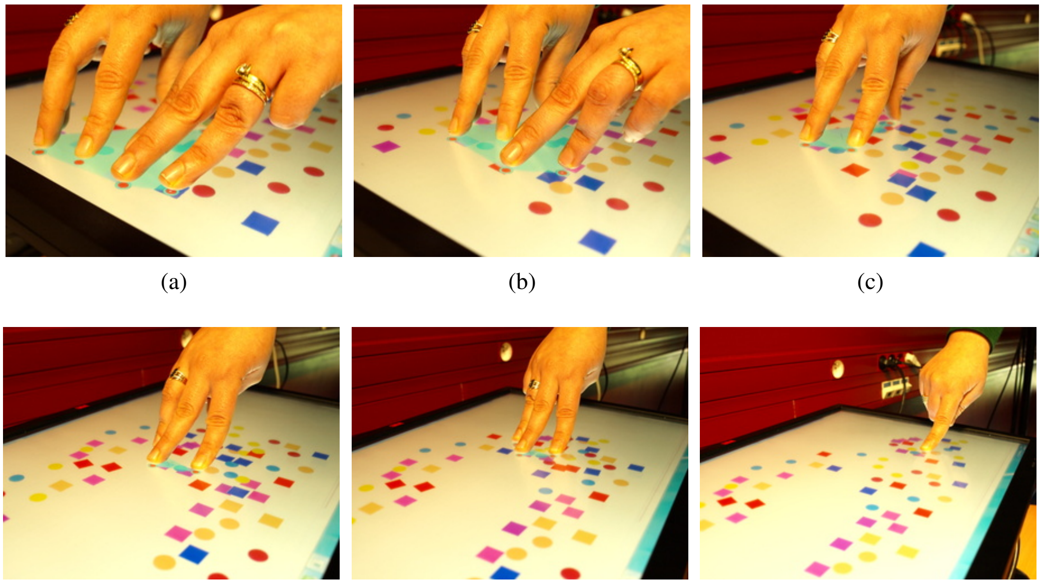
In this section, we present interaction techniques that we have designed in order to illustrate the practical interest of the guidelines discussed in the previous section and their compatibility with respect to the processing tools described before. The proposed techniques deal with the direct and indirect manipulation of multiple objects on an interactive surface. We first illustrate how the variability of the number of fingers within a rigid movement can allow the user to dynamically select and manipulate many objects at once (Prospect 1). We then illustrate how the state transition of rigid movements (Prospect 2) can allow the user to activate specific interaction actions. Finally, we tackle the more sophisticated issue of coordinating several separable movements within a global interaction task (Prospect 3) by describing variants of a cursor-based technique.

7.3.1 Sweep with Adjustable Size (Prospect 1)

Our first technique is a direct consequence of our first guideline. To enable the user to manipulate simple geometric forms, we make every rigid movement detected on the surface act like a selection tool. The objects intersecting the convex polygon defined by the contact-points of a rigid-movement are attached to it, and the same transformations (translation, scale, rotation) characterizing the rigid-movement are applied to these objects. Since objects are attached to the whole rigid-movement and not to individual contact-points considered separately, the number of fingers can vary arbitrary all along the selection task

(a) (b) (c)

Figure 7.4: Example of a group of objects being spanned and collected by a single rigid-movement (a,b,c).



(a) (b) (c)

(d) (e) (f)

Figure 7.5: Example of object attachment to a rigid movement and dynamic adaptation of the collection area: one *single* rigid movement induced by two hands (a,b), then one hand with variable number of fingers (c-h).

without loosing objects. Besides being extremely simple, this technique exhibits other interesting features which are discussed in the following paragraphs.

Adaptation of the number of fingers. The user can collect many objects at once using very few fingers as long as those fingers are moving in a rigid way. Since the number of fingers does not influence the definition of a rigid movement, he can adapt the surface induced by fingers at runtime, thus adjusting the selection surface dynamically. For instance, using both hands to construct a rigid-movement (Figure 7.4), the user can quickly collect a large number of small objects. He can also hold one hand or some fingers in order to decrease the

(a) (b)

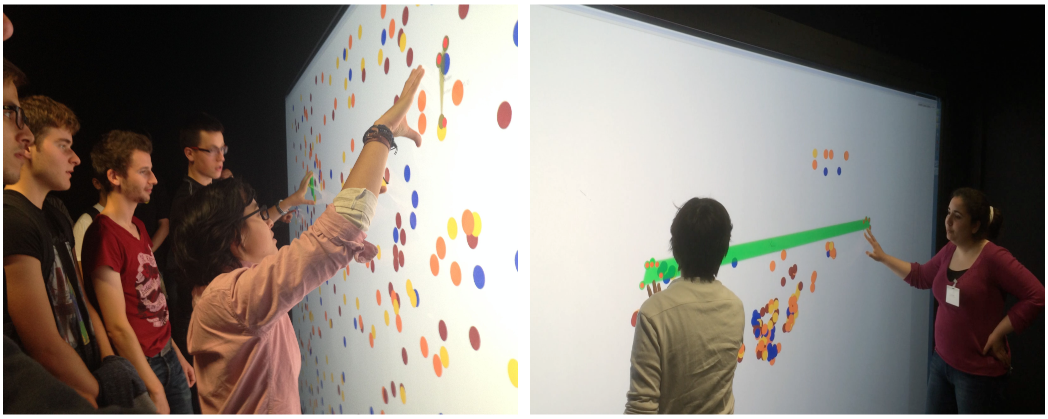
Figure 7.6: Two simultaneous rigid movement induced by one user using two hands on a 3M screen (a) and a large screen measuring 4 x 2 meters (b).

surface of the selection tool; hence, increasing the precision of the selection while keeping the control over the already selected objects (Figure 7.5). The user can thus adjust the range of its selection dynamically by holding and releasing fingers or hands freely as needed.

Mapping of users' movements. Rotating and scaling objects becomes a very simple task since it is directly mapped to the 'natural' movement of users' fingers and not by a pre-defined specific gesture. Like in the real world, when a user wants to manipulate a physical object, he may vary the relative pause of his fingers as well as their number. Selecting and manipulating objects by attaching them to the rigid movement mimics this behavior.

Simultaneity of movements. The previous discussion can be extended to the scenario where several rigid movements are detected on the interactive surface. In this case, each rigid movement corresponding to a group of fingers is responsible for selecting and manipulating a group of objects in a completely independent manner. Thus, several groups of objects can be manipulated simultaneously using as many rigid movements as groups (see Figure 7.6).

Large screen applicability. Our technique can also be deployed in the context of large interactive screens as depicted in Figure 7.7. Different users can collaborate to perform one single rigid movement that spans a large part of the surface which would be impossible otherwise. They can also not do so and decide to perform rigid movements independently of one another. Interestingly, these different modes are not contradictory in the sense that a user can first collect some objects by his own and then coordinate with another user in order to perform a common rigid-movement.



(a) (b)

Figure 7.7: Examples of deploying our technique on a large interactive surface (4×2 meters): three users performing three independent simultaneous rigid movements (a) and two users collaborating to form the same rigid movement (b).

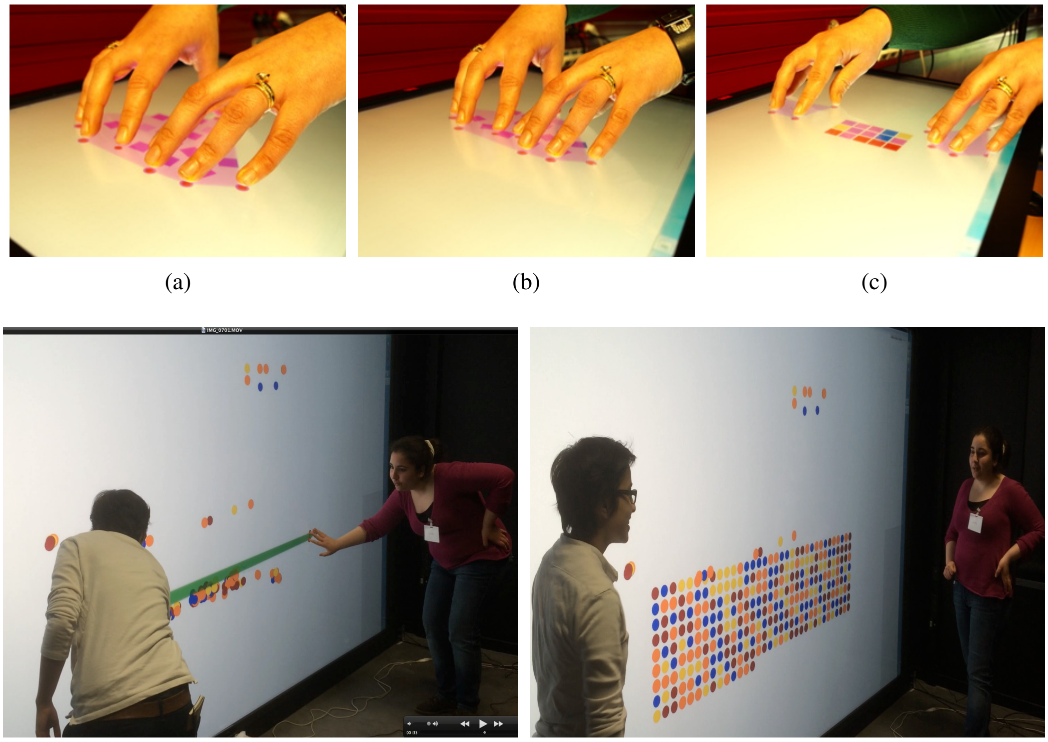
7.3.2 Hooking and Arranging (Prospect 2)

To illustrate the benefit of our second guideline, we extend the previous selection tool by exploring the possibility to activate specific actions when particular rigid-movement events are detected.

The user can use two rigid movements to select independent groups of objects, *e.g.*, using two hands. Then, whenever a merge event is detected, that is when all fingers start making the same movement, the selected objects get hooked together. Hooking objects means that the objects are virtually attached one to the other, and can be manipulated as if they were forming one single meta-object. By touching only a part of hooked objects with one rigid movement, the user can continue manipulating the whole hooked objects. For example, to add a target object to a group of objects that were already selected by a rigid-movement inferred by the right hand, we can select the target object by the left hand, and then move the left and right hands freely in a similar way to generate a merge event. Using a real-world metaphor, merging two rigid movements is then equivalent to the physical situation where the user holds two separate groups of objects in his two hands, and then glue them by joining hands together.

In the implemented technique, objects stay hooked together even if the user releases all his fingers, selects other objects, and then-after, restart manipulating the previously hooked objects. This enables to emulate a multi-selection tool where objects get hooked recursively by generating and merging as much rigid movements as needed. For instance, the user can start hooking two objects, possibly far away one the other, and manipulate them within a single rigid movement. Afterwards, the user can hook two other objects to form a second meta object, which can further be hooked to the first meta-object and so on.

We can symmetrically use a split movement to cancel the virtual hooking of objects. When a split event is detected over a hooked group of objects, we propose to rearrange



(a) (b) (c)

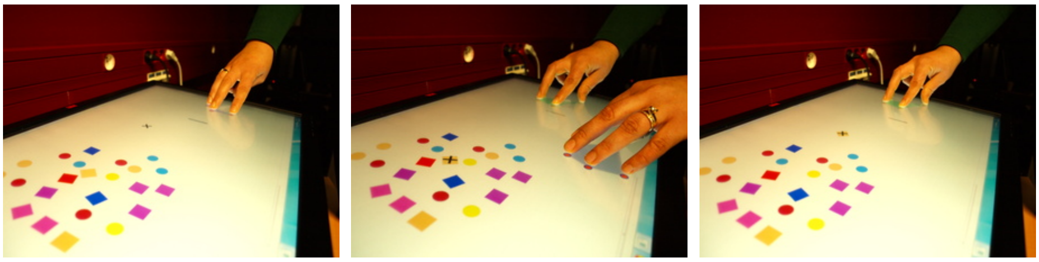
(d) (e)

Figure 7.8: Examples of a split event activating object arrangement: the split event is induced by one user using two hands (a,b,c) and two users, each one using one hand (d,e).

them smoothly one behind the other (Figure 7.8). For example, some objects can get hooked, then moved toward a specific region of the interactive surface using one single rigid-movement, then a free hand gesture with two fingers moving in different directions would activate their smooth arrangement. Using the physical metaphor, this corresponds to holding an object and tearing up some parts of it.

7.3.3 Indirect Cursor-Based Techniques (Prospect 3)

In this section, we illustrate how rigid-movements can be specialized in order to control sub-parts of a whole interaction task (Prospect 3). As a target application, we investigate the design of multi-finger cursors for object manipulation. The choice of multi-touch cursors as an illustrative example stems from the fact that they combine different functionalities which can be perceptually different for users, *e.g.*, cursor creation, object pointing and selection, object manipulations, etc. We note that our main concern is to provide a proof-of-concept prototype application that shows how rigid-movements can be specialized. We shall precisely show that coordinated and independent rigid movements can be used to implement and to control the functionalities of a cursor, taken as an illustrative example, in a simultaneous and transparent manner. More precisely, we describe a step-by-step design of a relatively simple multi-touch cursor enabled by two separable rigid-movements.



(a) Controlling the cursor (b) Selecting an object (c) Translating the selected object

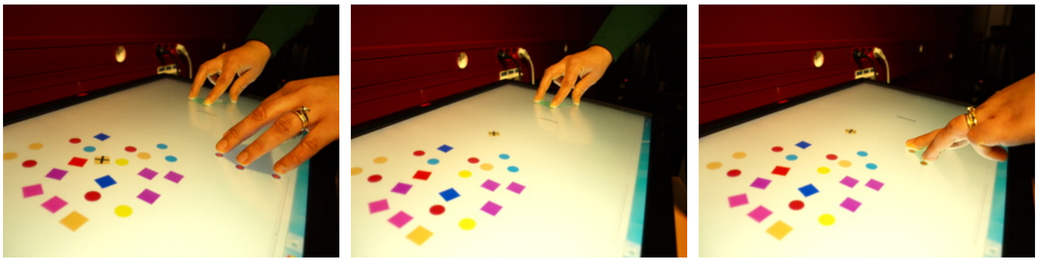
Figure 7.9: Example of object selection through the specialization of the left rigid-movement.

Cursor creation and destruction. We propose to use the split and the merge rigid-movement events in order to manage the transitions between direct and indirect interaction modes. More precisely, we create the cursor once a split event is detected on a free space of the interactive surface. This allows us to keep the direct interaction mode accessible even if the indirect interaction mode has been activated, *i.e.*, even if the cursor is already created. Initially, the position of the cursor is set to be the position where the split event has been detected.

The split event is further used to subdivide the interactive surface into two sub-surfaces by virtually drawing a vertical line passing through the initial cursor position. This enables to dynamically define two virtual sub-surfaces, the first one being at the right of the virtual line, and the second one being at the left of the virtual line. To kill the cursor, that is to deactivate the indirect mode, we proceed in a symmetric way using a merge event with respect to two rigid movements, each one located at one of the previously defined sub-surfaces.

Pointing, selection and deselection. The previously defined sub-surfaces enable to distinguish where a rigid movement is being performed. The relative position of a rigid-movement is used to specialize a specific behavior of the cursor. We use the rigid movement detected in the *right* sub-surface to control the position of the cursor. The cursor movement is then mapped to the translation experienced by the rigid-movement on the right sub-surface. To select and deselect an object, we specialize the rigid movement being on the *left* virtual sub-surface. Object selection is activated when a new rigid-movement is detected while it is unactivated when all rigid movements vanished in both the left and right sub-surfaces, *i.e.*, when all fingers are released. Hence, we can select an object by simply holding fingers on the left sub-surface, while being able to concurrently point a target object using the right rigid-movement (see Figure 7.9).

To summarize, the right rigid-movement is used for object selection (like the left-button of a mouse device), while the left rigid-movement is mapped to cursor movements and pointing. In the following, we show that these two rigid movements can be further specialized in order to manipulate selected objects in a separable and flexible manner.



(a) Selecting an object (b) Translating the selected object (c) Rotating the selected object

Figure 7.10: Example of object translation (b) and rotation (c) through the use of two independent rigid movements.

Object manipulation. First, we recall that for an object to stay selected, at least one rigid movement is required to stay alive either on the left or the right sub-surfaces. A selected object can then be translated and rotated *separately* following the same rigid transformation detected respectively on the right and the left sub-surfaces (Figure 7.10). More precisely, once an object is selected, the task consisting in modifying its position is mapped to the input task of moving the cursor following the translation induced by the left-side rigid movement of the left sub-surface. The task consisting in rotating the selected object is matched to the input task of producing a rotational right-side rigid-movement.

We remark that the separability between translation and rotation in different rigid movements makes object manipulation transparent, since translating and rotating objects are two well identified tasks which can be performed in a separable way either concurrently or sequentially, depending on the application context or on users' preferences.

Extensions. The previously described technique can be extended in several manners. In fact, by slightly modifying/specializing the role of each rigid movement, our movement-based design methodology can leverage different types of multi-finger cursors at the price of very little effort, and without using any external devices but the interactive surface itself. We provide in Appendix B two multi-touch cursor variants to further illustrate the benefit of separating and specializing rigid-movements.

7.4 Informal Evaluation and Discussion

The techniques described in the previous section are of general purpose and were not targeting a specific interaction task. Their primary aim is there-by to illustrate the feasibility of the rigid-movement concept introduced at the beginning of this chapter, and to point out in a more practical manner the accuracy of the prospective guidelines described consequently. All our prototypes were tested internally by the permanent and visiting members in our research team, and we were able to assess their accuracy to exploit the rigid-movement concept. A standard performance assessment study with quantitative metrics that one might perform in the typical case of a comparative study is simply not feasible in the current stage of our development. Nevertheless, and in order to further appreciate the pros and cons of our approach, we considered to experiment our prototype applications in a real-world situation and to expose them to users' objective opinion. As such, we had the opportunity to deploy some of our prototypes for a couple of time in different scientific and educational meetings. In the following, we report our informal observations concerning users' feedback.

The first time we experimented our prototypes in a public place was during the forth edition of FITG[2] which took place at the Imaginarium[3]. FITG is a national forum dedicated to tactile and gestural interaction which gathered hundred of people which are not necessary expert in interactive surfaces, but rather interested in their applications and development in our everyday life. The Imaginarium is an economic and scientific center dedicated to digital creativity and innovation. It offers an outstanding opportunity for people to develop and to exhibit new ideas and technologies focusing on several areas such as: 3D, serious games, image, interaction, etc. In this respect, a very large tactile screen (4×2 meters) was made available for the first time by the Imaginarium for those who wanted to illustrate new current trends in interactive techniques and their applications. Our prototype application for manipulating objects was first deployed and tested in this context. Since then, our application is still installed and regularly used at the Imaginarium for exhibition purposes. Therefore, we had the opportunity to experience users' feedbacks in other situations, especially in two particular meetings. The first one was during a party, including officials, companies and anonymous people, which was held to celebrate the activities at the Imaginarium. The second one was during another party where the master' students at the different departments of our university were invited to experiment the different applications exhibited at the Imaginarium.

Since our prototype for object manipulation is based on merge and split rigid-movements, we were able to collect interesting observations and feedback concerning the ability of users to perform this kind of movements. First, all users were enthusiastic when freely manipulating objects by one or several rigid-movements. They in fact expressed that this was comfortable and natural for them. In particular, users were able to perform one single rigid movement with both their hands and thus to dynamically span a variable size area in the interactive space. This was particularly appealing for users who tried to perform the movements in different manners, like if they were testing the different possibilities offered by the system and pushing it to its limit. After few attempts, users were able to master the split and the merge events and to manipulate objects in a free manner. Some users found that although split movements were simple and easy to perform, the merge movements required more attention to be activated. In addition, some users expressed that a visual feedback with respect to the objects being selected with one rigid movement could have been more appealing. This suggests that future developments should consider more carefully to provide a runtime feedback with respect to the actions attached to rigid-movements. More interestingly, and since our prototype was deployed on very large screen, we observed a tendency of users to collaborate together (which was not the primary goal of

[2]http://fitg.lille.inria.fr/

[3]http://www.plaine-images.fr/

our prototype). Many users find it interesting to collect objects using rigid movements performed cooperatively which we attribute to the flexibility in performing rigid-movements. However, one limitation was raised when many users (up to four) attempted to alternate between personal rigid-movements and cooperative rigid-movements. In this situation, some movements performed separately by different users were detected as forming one single rigid-movement by our system although the users did not intend to do so. This situation appeared to disturb a very little the corresponding users. Surprisingly, they started to coordinate their movements without any instructions and to speak together in order to continue interacting in a free manner. This limitation suggests that future developments should consider alternative rules to activate the collaborative or the individual achievement of rigid-movements performed simultaneously by a group of people especially on large screens.

7.5 Chapter Summary

In this chapter, we introduced the rigid-movement concept and illustrated its possible usage and advantages. The main goal of rigid-movements is to offer more flexibility, to open innovative interaction possibilities and to alleviate interaction from unnecessary constraints such as fixing the number of fingers or the shape of a gesture. In a first step, we provided a comprehensive methodology to extract rigid-movements at runtime based on the position of contact-points detected by the interactive surface. In a second step, we proposed and discussed some prospective design principles that can serve to design interaction techniques. Finally, we illustrated our approach by describing some prototype applications for direct and indirect object manipulation. In the future, we aim at enlarging the applicability domain of rigid-movements and to further explore the different interaction possibilities they can offer. One interesting issue is to comprehensively implement our concept in a high level library which would allow one to integrate our approach in existing applications with a minimum development cost.

8 Conclusion and Perspectives

Multi-touch surfaces offer a rich set of degree of freedom that can be independently controlled during gesture articulation. Researchers have proposed a set of tools and studies to define a gesture sets resulting in a large panel of multi-touch gestures. In this thesis, we advocate for deeper user studies to better understand how users interact using multi-touch gestures, identify their characteristics in a rigorous manner and take full benefit from users' gestures when designing interaction techniques. As gestures are versatile, we argue for designing interaction techniques that support many-to-one mapping between gestures and commands. In our research methodology, we start by exploring the variability of multi-touch gesture input from a user-centric perspective. We conducted a pair of user studies allowing us to investigate and to understand how users handle variability within multi-touch gestures. From the first study, we leverage a taxonomy of multi-touch gestures and introduced the concept of atomic movement. In the second study, we provide a more deeper analysis from which we characterize in a comprehensive manner the main sources of variations. In particular, we were able to analyze the link between gesture shape and gesture articulation, and to outline eight representative gesture classes characterized with geometric and kinematic descriptors. This motivated us to investigate how users perceive the difficulty of multi-touch gestures. To achieve this goal, we conducted a third user study, where users' were asked to rank and rate their gestures performed in different articulation classes. This allowed us to study the effect of each source of variation on the gesture articulation difficulty through a rigorous and comprehensive statistical analysis. Besides, we outlined a set of guidelines for the design of multi-touch gesture sets which correlate multi-touch ergonomics, multi-touch gesture recognizer and potential gesture to function mappings. In a second step, we aimed at dealing with the variability of users from a more system-oriented perspective. We thereby designed a new preprocessing technique that allows us to structure finger movements in a consistent manner. This allowed us to design more efficient and accurate recognizer for multi-touch gesture under unconstrained articulation. We also investigated the concept of rigid movements and investigated its potential usage in order to strength interaction and offer users more flexibility in articulating gestures. In particular, we showed how the concept of rigid movement enables to free users from a prefixed number of fingers, as well as from a predefined trace, when articulating a gesture.

It is our hope that the results presented in this thesis can serve as a starting point for future insightful research. Hopefully, many research issues remain open and deserve further investigations. We sketch some of them in the following. Notice that we basically organize our future research ideas in two parts: The first one deals with the design and the development of new interaction techniques, and the second one deals with the study of users' gestures.

Interaction techniques. It is worth noting that further investigations on how to take full benefit from users' variability to design new interaction techniques and how to analyze the accuracy of such techniques are the first straightforward consequence of our work. A good starting point could be to extend and to enhance the application prototypes described at the end of this dissertation in order to fully integrate them in a complete interaction system and to evaluate them within a formal user study.

Besides, other techniques could be explored as well. One can for instance ask how to benefit from the gesture articulation variability to increase the learning and memorization of a gesture set. To give a concrete example, and in line with work done in marking menus, it can be interesting to design a technique that augment the menubar on multi-touch surfaces using gesture variability as follows: semantically related commands can be assigned to the same gesture type and gesture articulation class can be used to differentiate between commands. This idea can be applied beyond any specific application, and we believe that it can constitute a generic paradigm that can help the design of strong and flexible interaction techniques.

Users' gestures. From our user studies, we found that users articulate gestures using single finger as well as multiple fingers by stroke, single and multiple strokes and synchronous and asynchronous strokes. This also allowed us to construct a big database containing different gestures articulations and samples. An interesting research idea can be to adopt a dual approach in order to better understand how users conceive gestures. More precisely, what if we conduct an experiment where we show a representative set of gesture samples to users, for instance taken from our user-defined database, and ask them to assess the differences and similarities of shown gestures? How will they classify those gestures and what will be their mental procedure in doing so? These questions are difficult to answer because of two issues. First, we have to choose accurately the set of samples to show to users; otherwise the potential results of the experiment could be biased. Second, we are not users and hence we have to design the right procedure allowing to understand how users will effectively proceed. Actually, these questions are tightly related to the so-called Gestalt theory which is often used to explain how people perceive points and images in two dimensional spaces. Such an experiment would allow to make a bridge between Gestalt laws and multi-touch users' variability which would provide new tools and paradigms to harness multi-touch gestures.

Besides, in our user study on the perceived difficulty on multi-touch gestures, we found that users are consistent in their assessment of the articulation difficulty under the many degrees of freedom offered by multi-touch devices. An challenging question is to design a predictive model to estimate the articulation difficulty of multi-touch gestures in the absence of a formal experiment. Such a model would for instance help designer in choosing gesture sets. We have already found that there exist correlation between gesture descriptors and the RATING and RANKING scores. A good starting point would be to use those descriptors to estimate the articulation difficulty. One major issue would be to assess the accuracy of such a model knowing the versatility of users.

Furthermore, our users' studies open several questions. For instance, in our user study on understanding multi-touch gesture variability, participants were given total freedom to articulate their gestures in terms of number of fingers, number of hands, etc. Inversely, in the study on understanding gesture difficulty, participants were asked to enter gestures by following our specific instructions *e.g.*, "draw a rectangle with two fingers" or "draw a start with three or more strokes". In future work, we plan to investigate the difference between gesture articulated in unconstrained conditions and those articulated in supervised condition. During all our studies, we get remarks from our participants concerning the availability of visual feedback when articulating multi-touch gestures. Accordingly, we plan to study the effect of providing visual feedback for multi-touch gesture articulation and to study what type of feedback we should give to users? Our techniques for structuring gesture articulation will be of great help in designing an accurate feedback which would be a nice feature that can be used independently in other interaction techniques. Over all studies, participants were adults. In future work, we plan to investigate younger children as well and to investigate whether similar results can be obtained.

Finally, a difficult an fully open research area would be to investigate users variability in the context of mid-air gestures or gestures mixing mid-air and multi-touch articulations. In fact, the new available technology makes it possible to investigate such issue and to better understand how gestural interaction can be embodied in users behaviors.

Publications

Refereed International Conferences

- ♣ [Rekik 2014b]: **Yosra Rekik**, Radu-Daniel Vatavu and Laurent Grisoni. Understanding Users' Perceived Difficulty of Multi-Touch Gesture Articulation. In Proceedings of the 16^{th} International Conference on Advanced Visual Interfaces, to appear, November 2014, Istanbul, Turkey. 8 pages (ACM).

- ♣ [Rekik 2014a]: **Yosra Rekik**, Radu-Daniel Vatavu and Laurent Grisoni. Match-up & Conquer: A Two-step Technique for Recognizing Unconstrained Bimanual and Multi-finger Touch Input. In Proceedings of the 12^{th} International Conference on Advanced Visual Interfaces, pages 201–208, May 2014, Como, Italy. 8 pages (ACM).

- ♣ [Rekik 2013]: **Yosra Rekik**, Laurent Grisoni and Nicolas Roussel. Towards Many Gestures to One Command: A User Study for Tabletops. In Proceedings of the 14^{th} IFIP TCI3 Conference on Human-Computer Interaction, pages 246–263. September 2013, Cap Town, South Africa. 18 pages (Springer).

Refereed Domestic Conferences

- ♣ [Rekik 2012]: **Yosra Rekik**, Nicolas Roussel and Laurent Grisoni. Mouvements Pseudo-rigides Pour Des Interactions Multi-doigts Plus Flexibles. In Proceedings of the 24^{th} Conference on Ergonomie Et Interaction Homme-machine, Ergo'IHM'12, pages 241–244, October 2012, Biaritz, France. 4 pages (ACM).

Datasets

- ♣ [DS1] Creativity Dataset: A database of multi-touch gestures composed of 5.155 samples of 22 gesture types collected from 16 participants[1]

- ♣ [DS2] Supervised Dataset: A dataset of 7.200 samples of 30 gesture types collected from 18 participants annotated with RATING and RANKING data[2].

Libraries

- ♣ [B1] Match-Up & Conquer Library: A c++ library distributed under the LGPL version2 license agreement[3].

[1] https://sites.google.com/site/yosrarekikresearch/projects/matchup
[2] https://sites.google.com/site/yosrarekikresearch/projects/gesturedifficulty
[3] https://sites.google.com/site/yosrarekikresearch/projects/matchup

Bibliography

[Aigner 2012] Roland Aigner, Daniel Wigdor, Hrvoje Benko, Michael Haller, David Lindbauer, Alexandra Ion, Shengdong Zhao, and Jeffrey Tzu Kwan Valino Koh. *Understanding Mid-Air Hand Gestures: A Study of Human Preferences in Usage of Gesture Types for HCI*. article MSR-TR-2012-111, Microsoft Research, November 2012. 10 pages. (Cited on page 19.)

[Anthony 2010] Lisa Anthony and Jacob O. Wobbrock. *A lightweight multistroke recognizer for user interface prototypes*. In Proceedings of GI '10, pages 245–252, Toronto, Ont., Canada, Canada, 2010. Canadian Information Processing Society. (Cited on pages 6, 11, 25, 26, 27, 36, 54, 57, 105 and 110.)

[Anthony 2012a] Lisa Anthony, Quincy Brown, Jaye Nias, Berthel Tate and Shreya Mohan. *Interaction and recognition challenges in interpreting children's touch and gesture input on mobile devices*. In Proceedings of ITS '12, pages 225–234, New York, NY, USA, 2012. ACM. (Cited on pages 36 and 54.)

[Anthony 2012b] Lisa Anthony and Jacob O. Wobbrock. *$N-protractor: a fast and accurate multistroke recognizer*. In Proceedings of GI '12, pages 117–120, Toronto, Ont., Canada, Canada, 2012. Canadian Information Processing Society. (Cited on pages 6, 11, 25, 26, 57 and 105.)

[Anthony 2013a] Lisa Anthony, Quincy Brown, Jaye Nias and Berthel Tate. *Examining the Need for Visual Feedback During Gesture Interaction on Mobile Touchscreen Devices for Kids*. In Proceedings of IDC '13, pages 157–164, New York, NY, USA, 2013. ACM. (Cited on pages 22, 86 and 87.)

[Anthony 2013b] Lisa Anthony, Radu-Daniel Vatavu and Jacob O. Wobbrock. *Understanding the Consistency of Users' Pen and Finger Stroke Gesture Articulation*. In Proceedings of GI '13, pages 87–94, Toronto, Ont., Canada, Canada, 2013. Canadian Information Processing Society. (Cited on pages 2, 21, 57, 86, 87, 99 and 100.)

[Anthony pear] L. Anthony, Q. Brown, B. Tate, J. Nias, R. Brewer and G. Irwin. *Designing smarter touch-based interfaces for educational contexts*. Journal of Personal and Ubiquitous Computing, 2013. to appear. (Cited on pages 106 and 110.)

[Appert 2009] Caroline Appert and Shumin Zhai. *Using Strokes As Command Shortcuts: Cognitive Benefits and Toolkit Support*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, pages 2289–2298, New York, NY, USA, 2009. ACM. (Cited on pages 11, 15 and 75.)

[Bailly 2012] Gilles Bailly, Jörg Müller and Eric Lecolinet. *Design and evaluation of finger-count interaction: Combining multitouch gestures and menus*. IJHCS, vol. 70, no. 10, pages 673–689, October 2012. (Cited on pages 2, 11, 12, 13, 15, 56 and 73.)

[Balakrishnan 1999] Ravin Balakrishnan and Ken Hinckley. *The Role of Kinesthetic Reference Frames in Two-handed Input Performance*. In Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology, UIST '99, pages 171–178, New York, NY, USA, 1999. ACM. (Cited on page 12.)

[Balakrishnan 2000] Ravin Balakrishnan and Ken Hinckley. *Symmetric Bimanual Interaction*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '00, pages 33–40, New York, NY, USA, 2000. ACM. (Cited on page 12.)

[Bau 2008] Olivier Bau and Wendy E. Mackay. *OctoPocus: a dynamic guide for learning gesture-based command sets*. In Proceedings of UIST '08, pages 37–46, New York, NY, USA, 2008. ACM. (Cited on page 14.)

[Baudel 1993] Thomas Baudel and Michel Beaudouin-Lafon. *Charade: remote control of objects using free-hand gestures*. Commun. ACM, vol. 36, no. 7, pages 28–35, July 1993. (Cited on page 19.)

[Beaudouin-Lafon 2001] Michel Beaudouin-Lafon. *Novel Interaction Techniques for Overlapping Windows*. In Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology, UIST '01, pages 153–154, New York, NY, USA, 2001. ACM. (Cited on page 9.)

[Benko 2006] Hrvoje Benko, Andrew D. Wilson and Patrick Baudisch. *Precise Selection Techniques for Multi-touch Screens*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '06, pages 1263–1272, New York, NY, USA, 2006. ACM. (Cited on page 11.)

[Benko 2009] Hrvoje Benko, T. Scott Saponas, Dan Morris and Desney Tan. *Enhancing Input on and Above the Interactive Surface with Muscle Sensing*. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '09, pages 93–100, New York, NY, USA, 2009. ACM. (Cited on page 58.)

[Blagojevic 2010] Rachel Blagojevic, Samuel Hsiao-Heng Chang and Beryl Plimmer. *The power of automatic feature selection: Rubine on steroids*. In Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium, SBIM '10, pages 79–86, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association. (Cited on pages 20, 69, 86, 87 and 88.)

[Bozinovic 1982] Radmilo Bozinovic and Sargur N. Srihari. *A String Correction Algorithm for Cursive Script Recognition*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 4, no. 6, pages 655–663, November 1982. (Cited on page 25.)

[Bragdon 2010] Andrew Bragdon, Arman Uguray, Daniel Wigdor, Stylianos Anagnostopoulos, Robert Zeleznik and Rutledge Feman. *Gesture Play: Motivating Online Gesture Learning with Fun, Positive Reinforcement and Physical Metaphors*. In ACM International Conference on Interactive Tabletops and Surfaces, ITS '10, pages 39–48, New York, NY, USA, 2010. ACM. (Cited on page 14.)

[Brandl 2008] Peter Brandl, Clifton Forlines, Daniel Wigdor, Michael Haller and Chia Shen. *Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces*. In Proceedings of AVI '08, pages 154–161, New York, NY, USA, 2008. ACM. (Cited on page 14.)

[Buchanan 2013] Sarah Buchanan, Bourke Floyd, Will Holderness and Joseph J. LaViola. *Towards User-defined Multi-touch Gestures for 3D Objects*. In Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces, ITS '13, pages 231–240, New York, NY, USA, 2013. ACM. (Cited on pages 16, 19 and 75.)

[Buxton 1986] W. Buxton and B. Myers. *A study in two-handed input*. In Proceedings of CHI '86, pages 321–326, New York, NY, USA, 1986. ACM. (Cited on page 12.)

[Cao 2007] Xiang Cao and Shumin Zhai. *Modeling human performance of pen stroke gestures*. In Proceedings of CHI '07, pages 1495–1504. ACM, 2007. (Cited on pages 24, 80 and 89.)

[Cao 2008a] Xiang Cao, AD. Wilson, R. Balakrishnan, K. Hinckley and S.E. Hudson. *ShapeTouch: Leveraging contact shape on interactive surfaces*. In Horizontal Interactive Human Computer Systems, 2008. TABLETOP 2008. 3rd IEEE International Workshop on, pages 129–136, Oct 2008. (Cited on page 10.)

[Cao 2008b] Xiang Cao, Andrew D. Wilson, Ravin Balakrishnan, Ken Hinckley and Scott E. Hudson. *ShapeTouch: Leveraging contact shape on interactive surfaces*. In 2008 IEEE International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP), pages 129–136. IEEE, October 2008. (Cited on page 11.)

[Caramiaux 2013] Baptiste Caramiaux, Frederic Bevilacqua and Atau Tanaka. *Beyond Recognition: Using Gesture Variation for Continuous Interaction*. In CHI '13 Extended Abstracts on Human Factors in Computing Systems, CHI EA '13, pages 2109–2118, New York, NY, USA, 2013. ACM. (Cited on page 3.)

[Casalta 1999] Didier Casalta, Yves Guiard and Michel Beaudouin-Lafon. *Evaluating Two-handed Input Techniques: Rectangle Editing and Navigation*. In CHI '99 Extended Abstracts on Human Factors in Computing Systems, CHI EA '99, pages 236–237, New York, NY, USA, 1999. ACM. (Cited on page 12.)

[Chen 2002] P.Y. Chen and P.M. Popovich. *Correlation: Parametric and nonparametric measure*. Thousand Oaks, CA: Sage, 2002. (Cited on pages 93, 94 and 95.)

[Cockburn 2007] Andy Cockburn, Per Ola Kristensson, Jason Alexander and Shumin Zhai. *Hard Lessons: Effort-inducing Interfaces Benefit Spatial Learning*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07, pages 1571–1580, New York, NY, USA, 2007. ACM. (Cited on page 15.)

[Cohé 2012] Aurélie Cohé and Martin Hachet. *Understanding user gestures for manipulating 3D objects from touchscreen inputs*. In Proceedings of GI '12, pages 157–164, Toronto, Ont., Canada, Canada, 2012. Canadian Information Processing Society. (Cited on pages 16, 19 and 34.)

[Cohen 1992] J. Cohen. *A power primer*. Psychol. Bull., vol. 112, no. 1, pages 155–159, 1992. (Cited on page 84.)

[Cormen 2003] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein. *Introduction to Algorithms, 2nd Ed.* MIT Press, 2003. (Cited on page 106.)

[Davidson 2008] Philip L. Davidson and Jefferson Y. Han. *Extending 2D Object Arrangement with Pressure-sensitive Layering Cues*. In Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology, UIST '08, pages 87–90, New York, NY, USA, 2008. ACM. (Cited on page 11.)

[de Almeida Madeira Clemente 2014] Mirko de Almeida Madeira Clemente, Hannes Leitner, Dietrich Kammer, Rainer Groh and André Pinkert. *Novel Interaction Techniques for Object Manipulation on Tabletops: Scoop Net and Pinch Helper*. In Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, AVI '14, pages 321–324, New York, NY, USA, 2014. ACM. (Cited on page 10.)

[Donmez 2012] N. Donmez and K. Singh. *Concepture: A Regular Language Based Framework for Recognizing Gestures with Varying and Repetitive Patterns*. In Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling, SBIM '12, pages 29–37, Aire-la-Ville, Switzerland, Switzerland, 2012. Eurographics Association. (Cited on page 28.)

[Efron 1941] D. Efron. *Gesture and environment*. Oxford, England: King'S Crown Press, 1941. (Cited on pages 7 and 17.)

[Elias 2007] J.G. Elias, W.C. Westerman and M.M. Haggerty. *Multi-touch gesture dictionary*, 2007. US Patent App. 11/619,553. (Cited on page 14.)

[Epps 2006] Julien Epps, Serge Lichman and Mike Wu. *A study of hand shape use in tabletop gesture interaction*. In CHI '06 Extended Abstracts on Human Factors in Computing Systems, CHI EA '06, pages 748–753, New York, NY, USA, 2006. ACM. (Cited on page 16.)

[FingerWorks 2001] FingerWorks. *iGesture products: Quick reference guides*, 2001. (Cited on page 14.)

[Fitzmaurice 1999] George W. Fitzmaurice, Ravin Balakrishnan, Gordon Kurtenbach and Bill Buxton. *An Exploration into Supporting Artwork Orientation in the User Interface*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '99, pages 167–174, New York, NY, USA, 1999. ACM. (Cited on page 9.)

[Forlines 2005] Clifton Forlines, Chia Shen, Frédéric Vernier and Mike Wu. *Under My Finger: Human Factors in Pushing and Rotating Documents Across the Table*. In Proceedings of the 2005 IFIP TC13 International Conference on Human-Computer Interaction, INTERACT'05, pages 994–997, Berlin, Heidelberg, 2005. Springer-Verlag. (Cited on page 9.)

[Freeman 2009] Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris and Daniel Wigdor. *ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures*. In Proceedings of ITS '09, pages 165–172, New York, NY, USA, 2009. ACM. (Cited on pages 15, 19, 20 and 41.)

[Garner 1974] W.R. Garner. *The processing of information and structure*. The Experimental Psychology Series/ Arthur W. Melton consulting ed. L. Erlbaum Associates; distributed by Halsted Press, New York, 1974. (Cited on page 130.)

[Ghomi 2013] Emilien Ghomi, Stéphane Huot, Olivier Bau, Michel Beaudouin-Lafon and Wendy E. Mackay. *Arpège: Learning Multitouch Chord Gestures Vocabularies*. In Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces, ITS '13, pages 209–218, New York, NY, USA, 2013. ACM. (Cited on page 15.)

[Grossman 2005] Tovi Grossman and Ravin Balakrishnan. *The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area*. In Proc. ACM CHI'05, pages 281–290, New York, NY, USA, 2005. (Cited on page 7.)

[Guiard 1987] Y Guiard. *Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model*. Journal of Motor Behavior, vol. 19, pages 486–517, 1987. (Cited on page 12.)

[Guilford 1967] J.P. Guilford. *The nature of human intelligence*. McGraw-Hill series in psychology. McGraw-Hill, 1967. (Cited on page 34.)

[Hager-Ross 2000a] Charlotte Hager-Ross and Marc H. Schieber. *Quantifying the Independence of Human Finger Movements: Comparisons of Digits, Hands, and Movement Frequencies*. The Journal of neuroscience : the official journal of the Society for Neuroscience, vol. 20, no. 22, pages 8542–8550, November 2000. (Cited on page 46.)

[Häger-Ross 2000b] Charlotte Häger-Ross and Marc H. Schieber. *Quantifying the Independence of Human Finger Movements: Comparisons of Digits, Hands, and Movement Frequencies*. The Journal of Neuroscience, vol. 20, no. 22, pages 8542–8550, 2000. (Cited on page 56.)

[Hancock 2006] Mark S. Hancock, Sheelagh Carpendale, Frederic D. Vernier, Daniel Wigdor and Chia Shen. *Rotation and Translation Mechanisms for Tabletop Interaction*. In Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems, TABLETOP '06, pages 79–88, Washington, DC, USA, 2006. IEEE Computer Society. (Cited on page 9.)

[Harrison 2011] Chris Harrison, Julia Schwarz and Scott E. Hudson. *TapSense: Enhancing Finger Interaction on Touch Surfaces*. In Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11, pages 627–636, New York, NY, USA, 2011. ACM. (Cited on pages 2 and 11.)

[Hennecke 2011] Fabian Hennecke, Franz Berwein and Andreas Butz. *Optical Pressure Sensing for Tangible User Interfaces*. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '11, pages 45–48, New York, NY, USA, 2011. ACM. (Cited on page 2.)

[Henry 1990] Tyson R. Henry, Scott E. Hudson and Gary L. Newell. *Integrating Gesture and Snapping into a User Interface Toolkit*. In Proceedings of the 3rd Annual ACM SIGGRAPH Symposium on User Interface Software and Technology, UIST '90, pages 112–122, New York, NY, USA, 1990. ACM. (Cited on page 27.)

[Henze 2010] Niels Henze, Andreas Löcken, Susanne Boll, Tobias Hesselmann and Martin Pielot. *Free-hand gestures for music playback: deriving gestures with a user-centred process*. In Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, MUM '10, pages 16:1–16:10, New York, NY, USA, 2010. ACM. (Cited on page 17.)

[Hinckley 1997] Ken Hinckley, Randy Pausch, Dennis Proffitt, James Patten and Neal Kassell. *Cooperative Bimanual Action*. In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '97, pages 27–34, New York, NY, USA, 1997. ACM. (Cited on page 12.)

[Hinrichs 2011] Uta Hinrichs and Sheelagh Carpendale. *Gestures in the wild: studying multi-touch gesture sequences on interactive tabletop exhibits*. In Proceedings of CHI '11, pages 3023–3032, New York, NY, USA, 2011. ACM. (Cited on pages 2, 17 and 56.)

[Höchtl 2012] Anita Höchtl, Florian Geyer and Harald Reiterer. *A Comparison of Spatial Grouping Techniques on Interactive Surfaces*. In In Proceedings of Mensch und Computer 2012 - Interaktiv Informiert - Konstanz, Germany. Oldenbourg, Sep 2012. (Cited on page 10.)

[Hong 2000] Jason I. Hong and James A. Landay. *SATIN: A Toolkit for Informal Ink-based Applications*. In Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology, UIST '00, pages 63–72, New York, NY, USA, 2000. ACM. (Cited on page 27.)

[Huang 1995] Thomas S. Huang and Vladimir I. Pavlovic. *Hand Gesture Modeling, Analysis, and Synthesis*. In IEEE International Workshop on Automatic Face and Gesture Recognition, pages 73–79, 1995. (Cited on page 40.)

[Isokoski 2001] Poika Isokoski. *Model for Unistroke Writing Time*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '01, pages 357–364, New York, NY, USA, 2001. ACM. (Cited on page 24.)

[Jacob 1994] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane and M. Preston Mullen Jr. *Integrality and separability of input devices*. ACM Trans. Comput.-Hum. Interact., vol. 1, no. 1, pages 3–26, March 1994. (Cited on page 130.)

[Jansen 2012] Eline Jansen. Teaching users how to interact with gesture-based interfaces; a comparison of teaching-methods. Master's thesis, Technical University of Eindhoven, 2012. (Cited on page 17.)

[Jiang 2012] Yingying Jiang, Feng Tian, Xiaolong Zhang, Wei Liu, Guozhong Dai and Hongan Wang. *Unistroke gestures on multi-touch interaction: supporting flexible touches with key stroke extraction*. In Proceedings of IUI '12, pages 85–88, New York, NY, USA, 2012. ACM. (Cited on pages 27, 37, 49, 105 and 106.)

[Jones 2006] Lynette A. Jones and Susan J. Lederman. *Human Hand Function*. Oxford University Press, 2006. (Cited on page 56.)

[Kabbash 1994] Paul Kabbash, William Buxton and Abigail Sellen. *Two-handed Input in a Compound Task*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '94, pages 417–423, New York, NY, USA, 1994. ACM. (Cited on page 12.)

[Kabbash 1995] Paul Kabbash and William A. S. Buxton. *The "prince" technique: Fitts' law and selection using area cursors*. In Proc. ACM CHI'95, pages 273–279, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co. (Cited on page 7.)

[Kammer 2010] Dietrich Kammer, Jan Wojdziak, Mandy Keck, Rainer Groh and Severin Taranko. *Towards a formalization of multi-touch gestures*. In Proceedings of ITS '10, pages 49–58, New York, NY, USA, 2010. ACM. (Cited on pages 28 and 49.)

[Kane 2011] Shaun K. Kane, Jacob O. Wobbrock and Richard E. Ladner. *Usable Gestures for Blind People: Understanding Preference and Performance*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '11, pages 413–422, New York, NY, USA, 2011. ACM. (Cited on page 21.)

[Karam 2005] Maria Karam and m. c. schraefel. *A Taxonomy of Gestures in Human Computer Interactions*. Technical report, University of Southampton, 2005. (Cited on pages 16 and 17.)

[Kendall 1939] M.G. Kendall and B. Babington Smith. *The Problem of m Rankings*. The Annals of Mathematical Statistics, vol. 10, no. 3, pages 275–287, 1939. (Cited on pages 82 and 84.)

[Kendon 1988] Adam. Kendon. *How gestures can become like words*. In Crosscultural Perspectives in Nonverbal Communication, F. Poyatos (ed), pages 131–141, 1988. (Cited on pages 7 and 17.)

[Keskin 2011] Cem Keskin, Ali Taylan Cemgil and Lale Akarun. *DTW Based Clustering to Improve Hand Gesture Recognition*. In Proceedings of the Second International Conference on Human Behavior Unterstanding, HBU'11, pages 72–81, Berlin, Heidelberg, 2011. Springer-Verlag. (Cited on page 25.)

[Kin 2009] Kenrick Kin, Maneesh Agrawala and Tony DeRose. *Determining the Benefits of Direct-touch, Bimanual, and Multifinger Input on a Multitouch Workstation*. In Proceedings of Graphics Interface 2009, GI '09, pages 119–124, Toronto, Ont., Canada, Canada, 2009. Canadian Information Processing Society. (Cited on pages 2, 57, 59 and 60.)

[Kin 2011] Kenrick Kin, Björn Hartmann and Maneesh Agrawala. *Two-handed marking menus for multitouch devices*. ACM Trans. Comput.-Hum. Interact., vol. 18, no. 3, pages 16:1–16:23, August 2011. (Cited on pages 11, 12, 15 and 23.)

[Kin 2012a] Kenrick Kin, Björn Hartmann, Tony DeRose and Maneesh Agrawala. *Proton++: a customizable declarative multitouch framework*. In Proceedings of UIST '12, pages 477–486, New York, NY, USA, 2012. ACM. (Cited on pages 28 and 49.)

[Kin 2012b] Kenrick Kin, Björn Hartmann, Tony DeRose and Maneesh Agrawala. *Proton: multitouch gestures as regular expressions*. In Proceedings of CHI '12, pages 2885–2894, New York, NY, USA, 2012. ACM. (Cited on pages 28 and 105.)

[Kray 2010] Christian Kray, Daniel Nesbitt, John Dawson and Michael Rohs. *User-defined gestures for connecting mobile phones, public displays, and tabletops*. In Proceedings of MobileHCI '10, pages 239–248, New York, NY, USA, 2010. ACM. (Cited on pages 2 and 16.)

[Kristensson 2004] Per-Ola Kristensson and Shumin Zhai. *SHARK2: A Large Vocabulary Shorthand Writing System for Pen-based Computers*. In Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology, UIST '04, pages 43–52, New York, NY, USA, 2004. ACM. (Cited on page 57.)

[Kristensson 2007] Per Ola Kristensson and Shumin Zhai. *Command Strokes with and Without Preview: Using Pen Gestures on Keyboard for Command Selection*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07, pages 1137–1146, New York, NY, USA, 2007. ACM. (Cited on page 14.)

[Kruger 2005] Russell Kruger, Sheelagh Carpendale, Stacey D. Scott and Anthony Tang. *Fluid Integration of Rotation and Translation*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '05, pages 601–610, New York, NY, USA, 2005. ACM. (Cited on page 9.)

[Kurdyukova 2012] Ekaterina Kurdyukova, Matthias Redlin and Elisabeth André. *Studying User-defined iPad Gestures for Interaction in Multi-display Environment*. In Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces, IUI '12, pages 93–96, New York, NY, USA, 2012. ACM. (Cited on page 16.)

[Kurtenbach 1993] Gordon Paul Kurtenbach. *The Design and Evaluation of Marking Menus*. PhD thesis, Toronto, Ont., Canada, Canada, 1993. UMI Order No. GAXNN-82896. (Cited on page 11.)

[Kurtenbach 1994] G. Kurtenbach, T. P. Moran and W. Buxton. *Contextual animation of gestural commands*. In Computer Graphics Forum, pages 83–90, 1994. (Cited on page 14.)

[Kühnel 2011] Christine Kühnel, Tilo Westermann, Fabian Hemmert, Sven Kratz, Alexander Müller and Sebastian Möller. *I'm home: Defining and evaluating a gesture set for smart-home control*. International Journal of Human-Computer Studies, vol. 69, no. 11, pages 693 – 704, 2011. (Cited on page 17.)

[Landay 1993] James A. Landay and Brad A. Myers. *Extending an Existing User Interface Toolkit to Support Gesture Recognition*. In INTERACT '93 and CHI '93 Conference Companion on Human Factors in Computing Systems, CHI '93, pages 91–92, New York, NY, USA, 1993. ACM. (Cited on page 28.)

[Latulipe 2005] Celine Latulipe, Craig S. Kaplan and Charles L. A. Clarke. *Bimanual and Unimanual Image Alignment: An Evaluation of Mouse-based Techniques*. In Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, UIST '05, pages 123–131, New York, NY, USA, 2005. ACM. (Cited on page 12.)

[Li 2010] Yang Li. *Protractor: a fast and accurate gesture recognizer*. In Proceedings of CHI '10, pages 2169–2172, New York, NY, USA, 2010. ACM. (Cited on pages 6, 26, 57 and 105.)

[Lin 2000] John Lin, Ying Wu and T. S. Huang. *Modeling the constraints of human hand motion*. In Proceedings of the Workshop on Human Motion (HUMO'00), HUMO '00, pages 121–, Washington, DC, USA, 2000. IEEE Computer Society. (Cited on page 46.)

[Long 1998] Allan C Long, James A. Landay and Lawrence A. Rowe. *PDA and Gesture Uses in Practice: Insights for Designers of Pen-Based*. Rapport technique, Berkeley, CA, USA, 1998. (Cited on page 23.)

[Long 1999] Allan Christian Long Jr., James A. Landay and Lawrence A. Rowe. *Implications for a gesture design tool*. In Proceedings of CHI '99, pages 40–47, New York, NY, USA, 1999. ACM. (Cited on pages 16, 17 and 75.)

[Long 2000] A. Chris Long Jr., James A. Landay, Lawrence A. Rowe and Joseph Michiels. *Visual similarity of pen gestures*. In Proc. of CHI '00, pages 360–367, New York, NY, USA, 2000. ACM. (Cited on pages 17 and 23.)

[Lü 2012] Hao Lü and Yang Li. *Gesture coder: a tool for programming multi-touch gestures by demonstration*. In Proceedings of CHI '12, pages 2875–2884, New York, NY, USA, 2012. ACM. (Cited on pages 3, 28, 49 and 105.)

[Malik 2004] Shahzad Malik and Joe Laszlo. *Visual Touchpad: A Two-handed Gestural Input Device*. In Proceedings of the 6th International Conference on Multimodal Interfaces, ICMI '04, pages 289–296, New York, NY, USA, 2004. ACM. (Cited on pages 11 and 12.)

[Malik 2005] Shahzad Malik, Abhishek Ranjan and Ravin Balakrishnan. *Interacting with large displays from a distance with vision-tracked multi-finger gestural input*. In Proceedings of UIST '05, pages 43–52, New York, NY, USA, 2005. ACM. (Cited on page 16.)

[Marquardt 2011] Nicolai Marquardt, Johannes Kiemer, David Ledo, Sebastian Boring and Saul Greenberg. *Designing User-, Hand-, and Handpart-aware Tabletop Interactions with the TouchID Toolkit*. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '11, pages 21–30, New York, NY, USA, 2011. ACM. (Cited on page 58.)

[McNeill 1992] D. McNeill. *Hand and Mind: What Gestures Reveal about Thought*. Psychology/cognitive science. University of Chicago Press, 1992. (Cited on pages 1, 7, 16 and 17.)

[Mehta Nimish 1982] C. Mehta Nimish and University of Waterloo. School of Computer Science. *Flexible Machine Interface*. Canadian thesis. PhD. Thesis, Department of Electrical Engineering, University of Toronto, 1982. (Cited on page 7.)

[Mitchell 2003] G. Daryn Mitchell. *Orientation on Tabletop Displays*, 2003. (Cited on page 9.)

[Morris 2006] Meredith Ringel Morris, Anqi Huang, Andreas Paepcke and Terry Winograd. *Cooperative gestures: Multi-user gestural interactions for co-located groupware*. In Proceedings of CHI '06, pages 1201–1210, New York, NY, USA, 2006. ACM. (Cited on pages 9 and 16.)

[Morris 2010] Meredith Ringel Morris, Jacob O. Wobbrock and Andrew D. Wilson. *Understanding Users' Preferences for Surface Gestures*. In Proceedings of GI '10, pages 261–268, Toronto, Ont., Canada, Canada, 2010. Canadian Information Processing Society. (Cited on pages 1, 2, 13, 15, 16, 56 and 75.)

[Morris 2012] Meredith Ringel Morris. *Web on the Wall: Insights from a Multimodal Interaction Elicitation Study*. In Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces, ITS '12, pages 95–104, New York, NY, USA, 2012. ACM. (Cited on page 1.)

[Moscovich 2006] Tomer Moscovich and John F. Hughes. *Multi-finger cursor techniques*. In Proc. GI'06, pages 1–7, Toronto, Ont., Canada, Canada, 2006. (Cited on page 7.)

[Myers 1981] Cory S. Myers and Lawrence R. Rabiner. *Comparative Study of Several Dynamic Time-Warping Algorithms for Connected-Word Recognition*. The Bell System Technical Journal, vol. 60, no. 7, 1981. (Cited on page 25.)

[Myers 1990] Brad A. Myers, Dario A. Giuse, Roger B. Dannenberg, David S. Kosbie, Edward Pervin, Andrew Mickish, Brad Vander Zanden and Philippe Marchal. *Garnet: Comprehensive Support for Graphical, Highly Interactive User Interfaces*. Computer, vol. 23, no. 11, pages 71–85, November 1990. (Cited on page 28.)

[Myers 1997] Brad A. Myers, Richard G. McDaniel, Robert C. Miller, Alan S. Ferrency, Andrew Faulring, Bruce D. Kyle, Andrew Mickish, Alex Klimovitski and Patrick Doane. *The Amulet Environment: New Models for Effective User Interface Software Development*. IEEE Trans. Softw. Eng., vol. 23, no. 6, pages 347–365, June 1997. (Cited on page 27.)

[Nacenta 2009] Miguel A. Nacenta, Patrick Baudisch, Hrvoje Benko and Andy Wilson. *Separability of Spatial Manipulations in Multi-touch Interfaces*. In Proceedings of Graphics Interface 2009, GI '09, pages 175–182, Toronto, Ont., Canada, Canada, 2009. Canadian Information Processing Society. (Cited on pages 9, 121, 130 and 7.)

[Nacenta 2013] Miguel A. Nacenta, Yemliha Kamber, Yizhou Qiang and Per Ola Kristensson. *Memorability of pre-designed and user-defined gesture sets*. In Proceedings of CHI '13, pages 1099–1108, New York, NY, USA, 2013. ACM. (Cited on page 75.)

[Nielsen 2004a] M. Nielsen, M. Sterring, T.B. Moeslund and E. Granum. *A procedure for developing intuitive and ergonomic gesture interfaces for HCI*. In Proc. of GW '03. LNCS 2915, pages 409–420. Springer, 2004. (Cited on page 75.)

[Nielsen 2004b] Michael Nielsen, Moritz Störring, ThomasB. Moeslund and Erik Granum. *A Procedure for Developing Intuitive and Ergonomic Gesture Interfaces for HCI*. In Antonio Camurri and Gualtiero Volpe, editeurs, Gesture-Based Communication in Human-Computer Interaction, volume 2915 of *Lecture Notes in Computer Science*, pages 409–420. Springer Berlin Heidelberg, 2004. (Cited on pages 13 and 16.)

[North 2009] Chris North, Tim Dwyer, Bongshin Lee, Danyel Fisher, Petra Isenberg, George Robertson and Kori Inkpen. *Understanding Multi-touch Manipulation for Surface Computing*. In Proceedings of INTERACT '09, pages 236–249, Berlin, Heidelberg, 2009. Springer-Verlag. (Cited on page 10.)

[Odell 2004] Daniel L. Odell, Richard C. Davis, Andrew Smith and Paul K. Wright. *Toolglasses, Marking Menus, and Hotkeys: A Comparison of One and Two-handed Command Selection Techniques*. In Proceedings of Graphics Interface 2004, GI '04, pages 17–24, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004. Canadian Human-Computer Communications Society. (Cited on page 12.)

[Oh 2013] Uran Oh and Leah Findlater. *The challenges and potential of end-user gesture customization*. In Proceedings of CHI '13, pages 1129–1138, New York, NY, USA, 2013. ACM. (Cited on pages 17, 27, 36, 53, 55 and 99.)

[Owen 2005] Russell Owen, Gordon Kurtenbach, George Fitzmaurice, Thomas Baudel and Bill Buxton. *When It Gets More Difficult, Use Both Hands: Exploring Bimanual Curve Manipulation*. In Proceedings of Graphics Interface 2005, GI '05, pages 17–24, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. Canadian Human-Computer Communications Society. (Cited on page 12.)

[Piumsomboon 2013] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst and Andrew Cockburn. *User-defined gestures for augmented reality*, 2013. (Cited on page 19.)

[Plamondon 2000] Réjean Plamondon and Sargur N. Srihari. *On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 1, pages 63–84, January 2000. (Cited on page 25.)

[Rekik 2012] Yosra Rekik, Nicolas Roussel and Laurent Grisoni. *Mouvements Pseudo-rigides Pour Des Interactions Multi-doigts Plus Flexibles*. In Proceedings of the 2012 Conference on Ergonomie Et Interaction Homme-machine, Ergo'IHM '12, pages 241:241–241:244, New York, NY, USA, 2012. ACM. (Cited on pages 6 and 143.)

[Rekik 2013] Yosra Rekik, Laurent Grisoni and Nicolas Roussel. *Towards Many Gestures to One Command: A User Study for Tabletops*. In Human-Computer Interactionâ INTERACT 2013, volume 8118 of *Lecture Notes in Computer Science*, pages 246–263. Springer Berlin Heidelberg, 2013. (Cited on pages 4 and 143.)

[Rekik 2014a] Yosra Rekik, Radu-Daniel Vatavu and Laurent Grisoni. *Match-up & Conquer: A Two-step Technique for Recognizing Unconstrained Bimanual and Multi-finger Touch Input*. In Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, AVI '14, pages 201–208, New York, NY, USA, 2014. ACM. (Cited on pages 6 and 143.)

[Rekik 2014b] Yosra Rekik, Radu-Daniel Vatavu and Laurent Grisoni. *Understanding Users' Perceived Difficulty of Multi-Touch Gesture Articulation*. In Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, ICMI '14, New York, NY, USA, 2014. ACM. (Cited on pages 5 and 143.)

[Rekimoto 2002] Jun Rekimoto. *SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces*. In Proceedings of CHI '02, pages 113–120, New York, NY, USA, 2002. ACM. (Cited on pages 9 and 16.)

[Roudaut 2009] Anne Roudaut, Eric Lecolinet and Yves Guiard. *MicroRolls: expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb*. In Proceedings of CHI '09, pages 927–936, New York, NY, USA, 2009. ACM. (Cited on page 11.)

[Roy 2013] Quentin Roy, Sylvain Malacria, Yves Guiard, Eric Lecolinet and James Eagan. *Augmented Letters: Mnemonic Gesture-based Shortcuts*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, pages 2325–2328, New York, NY, USA, 2013. ACM. (Cited on page 11.)

[Rubine 1991a] Dean Rubine. *Specifying Gestures by Example*. SIGGRAPH Comput. Graph., vol. 25, no. 4, pages 329–337, July 1991. (Cited on pages 20, 86, 87 and 100.)

[Rubine 1991b] Dean Rubine. *Specifying gestures by example*. In Proc. of SIGGRAPH '91, pages 329–337. ACM Press, 1991. (Cited on pages 25 and 28.)

[Rubine 1992] Dean Harris Rubine. *The Automatic Recognition of Gestures*. PhD thesis, Pittsburgh, PA, USA, 1992. UMI Order No. GAX92-16029. (Cited on page 25.)

[Ruiz 2011] Jaime Ruiz, Yang Li and Edward Lank. *User-defined motion gestures for mobile interaction*. In Proceedings of CHI '11, pages 197–206, New York, NY, USA, 2011. ACM. (Cited on pages 1, 2, 16 and 19.)

[Rust 2014] Karen Rust, Meethu Malu, Lisa Anthony and Leah Findlater. *Understanding Childdefined Gestures and Children's Mental Models for Touchscreen Tabletop Interaction*. In Proceedings of the 2014 Conference on Interaction Design and Children, IDC '14, pages 201–204, New York, NY, USA, 2014. ACM. (Cited on page 16.)

[Salvador 2007] Stan Salvador and Philip Chan. *Toward Accurate Dynamic Time Warping in Linear Time and Space*. Intell. Data Anal., vol. 11, no. 5, pages 561–580, October 2007. (Cited on page 25.)

[Schieber 2004] Marc H. Schieber and Marco Santello. *Hand function: peripheral and central constraints on performance*. J Appl Physiol, vol. 96, no. 6, pages 2293–2300, June 2004. (Cited on page 46.)

[Schuler 1993] Douglas Schuler and Aki Namioka. *Participatory Design: Principles and Practices*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1993. (Cited on pages 16 and 34.)

[Segala 1995] Roberto Segala and Nancy Lynch. *Probabilistic simulations for probabilistic processes*. Nordic J. of Computing, vol. 2, no. 2, pages 250–273, June 1995. (Cited on page 46.)

[Seyed 2012] Teddy Seyed, Chris Burns, Mario Costa Sousa, Frank Maurer and Anthony Tang. *Eliciting Usable Gestures for Multi-display Environments*. In Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces, ITS '12, pages 41–50, New York, NY, USA, 2012. ACM. (Cited on page 16.)

[Sezgin 2005] Tevfik Metin Sezgin and Randall Davis. *HMM-based efficient sketch recognition*. In Proceedings of IUI '05, pages 281–283, New York, NY, USA, 2005. ACM. (Cited on page 25.)

[Shen 2004] Chia Shen, Frédéric D. Vernier, Clifton Forlines and Meredith Ringel. *DiamondSpin: An Extensible Toolkit for Around-the-table Interaction*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04, pages 167–174, New York, NY, USA, 2004. ACM. (Cited on page 9.)

[Sima 2008] C. Sima and E. Dougherty. *The peaking phenomenon in the presence of feature-selection*. Pattern Recognition Letters, vol. 29, pages 1667–1674, 2008. (Cited on page 107.)

[Sodhi 2012] Rajinder Sodhi, Hrvoje Benko and Andrew Wilson. *LightGuide: Projected Visualizations for Hand Movement Guidance*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, pages 179–188, New York, NY, USA, 2012. ACM. (Cited on page 15.)

[Spano 2012] Lucio Davide Spano, Antonio Cisternino and Fabio Paternò. *A compositional model for gesture definition*. In Proceedings of the 4th International Conference on Human-Centered Software Engineering, HCSE'12, pages 34–52, Berlin, Heidelberg, 2012. Springer-Verlag. (Cited on pages 28 and 49.)

[Streitz 1999] Norbert A. Streitz, Jörg Geissler, Torsten Holmer, Shin'ichi Konomi, Christian Müller-Tomfelde, Wolfgang Reischl, Petra Rexroth, Peter Seitz and Ralf Steinmetz. *i-LAND: An Interactive Landscape for Creativity and Innovation*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '99, pages 120–127, New York, NY, USA, 1999. ACM. (Cited on page 9.)

[Swigart 2005] Scott Swigart. *Easily write custom gesture recognizers for your Tablet PC applications*. `http://msdn.microsoft.com/en-us/library/aa480673.aspx`, 2005. (Cited on page 28.)

[Tan 2006] Pang-Ning Tan, Michael Steinbach and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley, 2006. (Cited on page 126.)

[Tandler 2001] Peter Tandler, Thorsten Prante, Christian Müller-Tomfelde, Norbert Streitz and Ralf Steinmetz. *Connectables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces*. In Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology, UIST '01, pages 11–20, New York, NY, USA, 2001. ACM. (Cited on page 9.)

[Tappert 1990] C. C. Tappert, C. Y. Suen and T. Wakahara. *The State of the Art in Online Handwriting Recognition*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 8, pages 787–808, August 1990. (Cited on page 25.)

[Thieffry 1981] S. Thieffry. *Hand gestures*. In The Hand (R. Tubiana, ed.), pages 488,492, Philadelphia, PA: Sanders, 1981. University of Chicago Press. (Cited on page 40.)

[Tse 2006] Edward Tse, Chia Shen, Saul Greenberg and Clifton Forlines. *Enabling interaction with single user applications through speech and gestures on a multi-user tabletop*. In Proceedings of AVI '06, pages 336–343, New York, NY, USA, 2006. ACM. (Cited on page 16.)

[Tse 2008] Edward Tse, Saul Greenberg, Chia Shen, Clifton Forlines and Ryo Kodama. *Exploring True Multi-user Multimodal Interaction over a Digital Table*. In Proceedings of the 7th ACM Conference on Designing Interactive Systems, DIS '08, pages 109–118, New York, NY, USA, 2008. ACM. (Cited on page 10.)

[Tu 2012] Huawei Tu, Xiangshi Ren and Shumin Zhai. *A Comparative Evaluation of Finger and Pen Stroke Gestures*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, pages 1287–1296, New York, NY, USA, 2012. ACM. (Cited on pages 22, 56 and 73.)

[Valdes 2014] Consuelo Valdes, Diana Eastman, Casey Grote, Shantanu Thatte, Orit Shaer, Ali Mazalek, Brygg Ullmer and Miriam K. Konkel. *Exploring the Design Space of Gestural Interaction with Active Tokens Through User-defined Gestures*. In Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems, CHI '14, pages 4107–4116, New York, NY, USA, 2014. ACM. (Cited on pages 16 and 19.)

[Vatavu 2011a] Radu-Daniel Vatavu. *The effect of sampling rate on the performance of template-based gesture recognizers*. In Proceedings of ICMI '11, pages 271–278, New York, NY, USA, 2011. ACM. (Cited on page 107.)

[Vatavu 2011b] Radu-Daniel Vatavu, Daniel Vogel, Géry Casiez and Laurent Grisoni. *Estimating the perceived difficulty of pen gestures*. In Proceedings of INTERACT '11, pages 89–106, Berlin, Heidelberg, 2011. Springer-Verlag. (Cited on pages 24, 36, 54, 75, 76, 80, 86, 87 and 99.)

[Vatavu 2012a] Radu-Daniel Vatavu. *1F: One Accessory Feature Design for Gesture Recognizers*. In Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces, IUI '12, pages 297–300, New York, NY, USA, 2012. ACM. (Cited on page 20.)

[Vatavu 2012b] Radu-Daniel Vatavu. *User-defined Gestures for Free-hand TV Control*. In Proceedings of the 10th European Conference on Interactive Tv and Video, EuroiTV '12, pages 45–48, New York, NY, USA, 2012. ACM. (Cited on pages 1 and 17.)

[Vatavu 2012c] Radu-Daniel Vatavu, Lisa Anthony and Jacob O. Wobbrock. *Gestures as point clouds: a $P recognizer for user interface prototypes*. In Proceedings of ICMI '12, pages 273–280, New York, NY, USA, 2012. ACM. (Cited on pages 6, 11, 25, 27, 49, 57, 105, 106, 107, 110, 111, 114 and 120.)

[Vatavu 2013a] Radu-Daniel Vatavu. *The Impact of Motion Dimensionality and Bit Cardinality on the Design of 3D Gesture Recognizers*. IJHCS, vol. 71, no. 4, pages 387–409, 2013. (Cited on page 107.)

[Vatavu 2013b] Radu-Daniel Vatavu, L. Anthony and J.O. Wobbrock. *Relative Accuracy Measures for Stroke Gestures*. In Proceedings of ICMI '13. ACM, 2013. (Cited on page 89.)

[Vatavu 2013c] Radu-Daniel Vatavu, Lisa Anthony and Jacob O. Wobbrock. *Relative Accuracy Measures for Stroke Gestures*. In Proceedings of the 15th ACM on International Conference on Multimodal Interaction, ICMI '13, pages 279–286, New York, NY, USA, 2013. ACM. (Cited on page 21.)

[Vatavu 2013d] Radu-Daniel Vatavu, Géry Casiez and Laurent Grisoni. *Small, Medium, or Large?: Estimating the User-perceived Scale of Stroke Gestures*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, pages 277–280, New York, NY, USA, 2013. ACM. (Cited on page 24.)

[Vatavu 2013e] R.D. Vatavu. *A Comparative Study of User-Defined Handheld vs. Freehand Gestures for Home Entertainment Environments*. Journal of Ambient Intelligence and Smart Environments, vol. 5, no. 2, pages 187–211, 2013. (Cited on page 1.)

[Vernier 2002] Frédéric Vernier, Neal Lesh and Chia Shen. *Visualization Techniques for Circular Tabletop Interfaces*. In Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '02, pages 257–265, New York, NY, USA, 2002. ACM. (Cited on page 9.)

[Wagner 2014] Julie Wagner, Eric Lecolinet and Ted Selker. *Multi-finger Chords for Hand-held Tablets: Recognizable and Memorable*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, pages 2883–2892, New York, NY, USA, 2014. ACM. (Cited on page 11.)

[Wang 2009a] Feng Wang, Xiang Cao, Xiangshi Ren and Pourang Irani. *Detecting and Leveraging Finger Orientation for Interaction with Direct-touch Surfaces*. In Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology, UIST '09, pages 23–32, New York, NY, USA, 2009. ACM. (Cited on page 11.)

[Wang 2009b] Feng Wang and Xiangshi Ren. *Empirical Evaluation for Finger Input Properties in Multi-touch Interaction*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, pages 1063–1072, New York, NY, USA, 2009. ACM. (Cited on page 11.)

[Webb 2002] Andrew Webb. *Statistical pattern recognition*. John Wiley & Sons, Ltd. cop., West Sussex, England, Hoboken, NJ, 2002. Réimpression : 2003. (Cited on page 109.)

[Wigdor 2011] Daniel Wigdor and Dennis Wixon. *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st édition, 2011. (Cited on page 13.)

[Willems 2009] Don Willems, Ralph Niels, Marcel van Gerven and Louis Vuurpijl. *Iconic and multi-stroke gesture recognition*. Pattern Recognition, vol. 42, no. 12, pages 3303 – 3312, 2009. (Cited on pages 20, 86 and 87.)

[Wilson 2008] Andrew D. Wilson, Shahram Izadi, Otmar Hilliges, Armando Garcia-Mendoza and David Kirk. *Bringing Physics to the Surface*. In Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology, UIST '08, pages 67–76, New York, NY, USA, 2008. ACM. (Cited on pages 10 and 11.)

[Wing 1996] Alan M. Wing, Patrick Haggard and J. Randall Flanagan. *Hand and brain: the neurophysiology and psychology of hand movements*. Academic Press, 1996. (Cited on page 56.)

[Wobbrock 2005] Jacob O. Wobbrock, Htet Htet Aung, Brandon Rothrock and Brad A. Myers. *Maximizing the guessability of symbolic input*. In CHI '05 Extended Abstracts on Human Factors in Computing Systems, CHI EA '05, pages 1869–1872, New York, NY, USA, 2005. ACM. (Cited on page 16.)

[Wobbrock 2007] Jacob O. Wobbrock, Andrew D. Wilson and Yang Li. *Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes*. In Proceedings of UIST '07, pages 159–168, New York, NY, USA, 2007. ACM. (Cited on pages 6, 11, 23, 25, 26, 27, 36, 54, 57, 105, 107 and 110.)

[Wobbrock 2009] Jacob O. Wobbrock, Meredith Ringel Morris and Andrew D. Wilson. *User-defined gestures for surface computing*. In Proceedings of CHI '09, pages 1083–1092, New York, NY, USA, 2009. ACM. (Cited on pages 1, 2, 13, 15, 16, 17, 18, 34, 37, 40, 41, 48, 56, 67, 75, 77, 99 and 100.)

[Wu 2003] Mike Wu and Ravin Balakrishnan. *Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays*. In Proceedings of UIST '03, pages 193–202, New York, NY, USA, 2003. ACM. (Cited on pages 2, 9, 12 and 18.)

[Wu 2006] Mike Wu, Chia Shen, Kathy Ryall, Clifton Forlines and Ravin Balakrishnan. *Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces*. In Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems, TABLETOP '06, pages 185–192, Washington, DC, USA, 2006. IEEE Computer Society. (Cited on pages 9, 10 and 19.)

[Zhai 2003] Shumin Zhai and Per-Ola Kristensson. *Shorthand Writing on Stylus Keyboard*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '03, pages 97–104, New York, NY, USA, 2003. ACM. (Cited on page 25.)

[Zhao 2004] Shengdong Zhao and Ravin Balakrishnan. *Simple vs. Compound Mark Hierarchical Marking Menus*. In Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology, UIST '04, pages 33–42, New York, NY, USA, 2004. ACM. (Cited on page 11.)

Appendices

A
Match-Up Pseudocode

We provide pseudocode for the Match-Up technique that will run at *every* timestamp t during the entire duration of articulation of the multi-touch gesture and deliver its representation as key strokes. A key stroke is represented by a cluster of points resulted from grouping together points that belong to similar strokes. POINT is a structure that defines a touch point with position coordinates (x, y), the two previous positions dp and ddp and identification id. POINTS is a list of points. CLUSTER is a structure that contains a list of $points$ and an id. CLUSTERS is a list of clusters.

MATCH-UP (POINTS $\mathscr{P}$, CLUSTERS *previous*)

clusters $\leftarrow$ **new** CLUSTERS
for *each* $p \in \mathscr{P}$ ***such as*** $dp \neq null$ **do**
    INSERT($clusters$, **new** CLUSTER(p))
while $|clusters| \geq 2$ **do**
    $(\mathscr{A}, \mathscr{B}) \leftarrow$ MOST-SIMILAR-CLUSTERS(*clusters*)
    if $(\mathscr{A}, \mathscr{B}) == null$ **then break**;
    $\mathscr{A}.points \leftarrow \mathscr{A}.points \cup \mathscr{B}.points$
    REMOVE($clusters$, $\mathscr{B}$)
MATCH-IDS(*clusters*, *previous*)
Return *clusters*

```
MOST-SIMILAR-CLUSTERS(CLUSTERS clusters)
ε_θ ← 30, ε_d ← 0.125 · INPUT-SIZE
θ_min ← ∞, (𝒜_min, ℬ_min) ← null
for each 𝒜 ∈ clusters do
    for each ℬ ∈ clusters do
        θ ← MINIMUM-ANGLE(𝒜, ℬ);
        δ ← MINIMUM-DISTANCE(𝒜, ℬ);
        if (θ ≤ ε_θ and δ ≤ ε_d) then
            if (θ < θ_min) then
                θ_min ← θ, 𝒜_min ← 𝒜, ℬ_min ← ℬ
```

```
MATCH-IDS(CLUSTERS current, CLUSTERS previous)
1  for each 𝒞 ∈ current do
2      matched ← false;
3      for each 𝒦 ∈ previous do
4          copy ← COPY-POINTS(𝒦.points)
5          for each p ∈ 𝒞.points such as ddp ≠ null do
6              matchedPt ← false;
7              for each q ∈ 𝒦.points do
8                  if (p_i.id == q.id) then
9                      REMOVE(copy, q)
10                     matchedPt ← true;
11                     break
12             if (not matchedPt) then continue 𝒦. /*Go to Line 3*/;
13         if (SIZE(copy) ≠ null) then continue 𝒦 ;
14         𝒞.id ← 𝒦.id
15         matched ← true
16         break
17     if (not matched) then 𝒞.id ← new id ;
18 Return current
```

```
MINIMUM-ANGLE(CLUSTER 𝒜, CLUSTER ℬ)
θ_min ← ∞
for each p ∈ 𝒜.points do
    for each q ∈ ℬ.points do
        a ← SCALAR-PRODUCT(p − dp, q − dq);
        b ← NORM(p − dp) · NORM(q − dq)
        θ ← ACOS(a/b);
        if (θ<θ_min) then θ_min ← θ ;
Return θ_min
```


MINIMUM-DISTANCE(CLUSTER $\mathscr{A}$, CLUSTER $\mathscr{B}$)

```
δ_min ← ∞
for each p ∈ 𝒜.points do
    for each q ∈ ℬ.points do
        δ ← EUCLIDEAN-DISTANCE(p, q)
        if (δ < δ_min) then δ_min ← δ ;
Return δ_min;
```

B
Cursor Variants Prototypes

In Section 7.3.3, we investigated the design of simple cursor enabled using two rigid movements. We believe that the specialization of interaction through separability of rigid movements is a powerful principle that can be used in different contexts and for more general purposes [Nacenta 2009]. We illustrate this claim by describing other variants of multi-finger cursors obtained following the same general design scheme, but simply specializing rigid movements in a slightly different manner. We emphasis that our primary goal is not guided by the specific cursor properties; consequently, we will not go through a detailed description of cursor properties, but we shall rather focus on the ease of design and feasibility issues.

Scalable cursor. Our basic cursor can be extended by adding an adjustable selection area around the vicinity of the cursor. This cursor is actually inspired by the work of Moscovich et al. [Moscovich 2006] which aim is to offer the user different degrees of selection precision [Kabbash 1995, Grossman 2005]. As in [Moscovich 2006], we endow the basic cursor with a dynamic user-controlled area defining an activation surface. This new feature can be enabled in a straightforward manner as follows. The size of the activation area is simply controlled by the scale movement done by fingers in the right virtual sub-surface (Figure B.1). Concurrently, selected objects can be translated and rotated using respectively the right and left rigid-movement exactly in the same way than previously. The novelty is that several orthogonal features of the cursor can be controlled in a coordinated manner using the left and right rigid-movement as illustrated in Table B.1. This is for instance to contrast with the cursor described in [Moscovich 2006], where an external device is used to both control the selection area, the cursor movement, and the manipulation of objects.

Shadow cursor. Our basic cursor can be extended to emulate contact-points in a remote area of the interactive surface and to manipulate distant objects using virtual direct touches. For this purpose, we propose to virtually move the contact-points detected in the left sub-

(a) Controlling the scalable cursor (b) Selecting objects (c) Manipulating selected objects

Figure B.1: Illustration of a rigid-movement based adjustable area cursor. The activation area is controlled by left hand induced rigid-movement.

	Left RM			Right RM		
	R	T	S	R	T	S
A. Cursor Manipulation						
Cursor position					√	
Cursor activation area						√
B. Object Manipulation						
Object selection		√				
Object position					√	
Object orientation	√					

Table B.1: Comprehensive matching of rigid-movements (RM for short) and cursor functionalities.

surface (Figure B.2). More precisely, the translation performed by the movement of fingers on the right-subsurface has the effect of translating the contact points being on the right side. In this way, we can move a set of virtual contact-points emulated by the fingers on the right sub-surface, *i.e.*, the real-contact points are shadowed on a remote and virtual space controlled by the right subsurface. Virtual contact-points can then be used to manipulate distant objects, as if they were directly touching users fingers. For instance, in the context of a large interactive surface, this enables to emulate a direct interaction style without moving around the table.

Customized cursors. Thanks to the the rigid-movement concept, the number of contact points has not constituted an issue when setting up the previously described techniques. In line with our first design guideline, the number of contact points can however serve as a secondary parameter. For instance, we can use this information in order to choose the type of cursors we want to activate. While a split movement is always used for cursor activation, the number of fingers can be additionally used to invoke a particular type of cursors. The user can of instance dynamically switch between one cursor to another by simply adapting the number of fingers he uses to generate the split event.

(a) Controlling the shadow cursor (b) Selecting objects (c) Manipulating selected objects

Figure B.2: Illustration of distant multi-touches matching the right hand rigid movement.

The number of fingers can also be used to control the precision of the designed cursors. Actually, in our prototype implementations, the amplification factor of the translation applied to cursors is controlled by the number of fingers. More precisely, we fix the velocity of the cursor (the gain) by using a simple linear function of the number of fingers being used in the translational movement. In our implementation, we consider that the more the user wants to move the cursor far away, the more he should add fingers to the rigid movement defined in the left sub-surface, *i.e.*, the velocity of the cursor can be adjusted dynamically at runtime by holding or removing fingers. Notice that we can similarly use the number of fingers to control the amplification factor of the rotation/translation applied to selected objects. Notice also that although other more sophisticated gain functions could be investigated, we do not address this issue in this thesis and we keep it open for future research.