USABILITY ANALYSIS IN LOCOMOTION INTERFACE FOR HUMAN COMPUTER INTERACTION SYSTEM DESIGN

Farzin Farhadi-Niaki

Thesis submitted to the University of Ottawa
in partial Fulfillment of the requirements for the
Doctorate in Philosophy degree in Electrical Engineering

Department of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

In the past decade and more than any time before, new technologies have been broadly applied in various fields of interaction between human and machine. Despite many functionality studies, yet, how such technologies should be evaluated within the context of human computer interaction research remains unclear. This research aims at proposing a mechanism to evaluate/predict the design of user interfaces with their interacting components. At the first level of analysis, an original concept extracts the usability results of components, such as effectiveness, efficiency, adjusted satisfaction, and overall acceptability, for comparison in the fields of interest. At the second level of analysis, another original concept defines new metrics based on the level of complexity in interactions between input modality and feedback of performing a task, in the field of classical solid mechanics. Having these results, a set of hypotheses is provided to test if some common satisfaction criteria can be predicted from their correlations with the components of performance, complexity, and overall acceptability. In the context of this research, three multimodal applications are implemented and experimentally tested to study the quality of interactions through the proposed hypotheses: a) full-body gestures vs. mouse/keyboard, in a Box game; b) arm/hand gestures vs. three-dimensional haptic controller, in a Slingshot game; and c) hand/finger gestures vs. mouse/keyboard, in a Race game. Their graphical user interfaces are designed to cover some extents of static/dynamic gestures, pulse/continuous touch-based controls, and discrete/analog tasks measured. They are quantified based on a new definition termed index of complexity which represents a concept of effort in the domain of locomotion interaction. Single/compound devices are also defined and studied to evaluate the effect of user's attention in multi-tasking interactions. The proposed method of investigation for usability is meant to assist human-computer interface developers to reach a proper overall acceptability, performance, and effort-based analyses prior to their final user interface design.

Acknowledgments

First and foremost, I would like to offer my sincerest gratitude to my supervisors:

Dr. Robert Laganière; I am so grateful for all your support, vision, wisdom, and mentorship. This work would have not been accomplished without your supervision.

Dr. Pierre Payeur; I truly appreciate your vast knowledge and experience that contributed very much to my academic growth, and your great attributes of an inspirational teacher, researcher, and human being.

Next, I would like to acknowledge the members of my defence committee, Dr. Ali Arya, Dr. Shi Cao, Dr. Hussein Al Osman, Dr. Emil Petriu, and the defence chair Dr. Beatriz Martin-Perez for their expertise that helped shape this work better.

In addition, I would like to thank the participants who contributed in our user experiments to test and evaluate our applications.

Last but not least, my special thanks go to my dear parents and brothers who granted me their unlimited love, faith, and support without which I would not be where I am today.

Finally, I dedicate this thesis to the dearest and the most valuable and significant person in my life, my son Arad, for his patience, encouragements, and brilliant ideas that supported me during my PhD program.

Table of Contents

List of Figures

List of Tables

Lexical Table

Acronym	**Definition**
HCI	Human-Computer Interaction
3D UI	Three Dimensional User Interface
GUI	Graphical User Interface
DOF	Degree Of Freedom
IoC	Index of Complexity
ISO	International Standards Organization
I/O	Input/Output
LoS	Line of Sight
TSP	Time-Space Pattern
SD	Shortest Distance
RM	Real Motion
NA	Number of Actions
NT	Number of Trials
TT	Total Time
SR	Stability Ratio
AR	Accuracy Ratio
AV	Activity Velocity
AF	Activity Frequency
R	Radius of object
E_s	Effectiveness
E_y	Efficiency
E	Easiness
F	non-Fatiguing
N	Naturalness
P	Pleasantness
M	Movability
$\widetilde{E}$	weighted Easiness
$\widetilde{F}$	weighted non-Fatiguing
$\widetilde{N}$	weighted Naturalness
$\widetilde{P}$	weighted Pleasantness
$\widetilde{M}$	weighted Movability
$\widetilde{E}_n$	normalized Easiness

$\widetilde{F}_n$	normalized non-Fatiguing
$\widetilde{N}_n$	normalized Naturalness
$\widetilde{P}_n$	normalized Pleasantness
$\widetilde{M}_n$	normalized Movability
$\widehat{E}$	adjusted Easiness
$\widehat{F}$	adjusted non-Fatiguing
$\widehat{N}$	adjusted Naturalness
$\widehat{P}$	adjusted Pleasantness
$\widehat{M}$	adjusted Movability
S_a	adjusted Satisfaction
OA	Overall Acceptability
$(DOF)_{Parametric}$	Parametric Degree Of Freedom
I	level of Intensity
RoS	Rate of Simplicity
IoS	Index of Similarity
SDK	Software Development Kit
API	Application Programming Interface
ppi	pixel per inch
LED	Light Emitting Diode
OpenNI	Open Natural Interaction
NiTE	Middleware from PrimeSense
OpenCV	Open-source Computer Vision
ANOVA	ANalysis Of VAriance
MATLAB	MATrix LABratory
MOG	Mixture Of Gaussian
RGB	Red, Green, Blue
YCbCr	Luminance; Chroma: Blue; Chroma: Red
LUV	Lightness (L) and chrominance (UV)
HSV	Hue, Saturation, Value
YUV	Luma (Y) and chrominance (UV)
CAMShift	Continuously Adaptive Mean Shift
COG	Centre Of Gravity
ASL	American Sign Language
UX	User eXperience

Chapter 1: Introduction

Human-Computer Interaction (HCI) studies the interaction between human and computing devices, where the information is transferred between the brain's neurons and silicon's bits. HCI essentially aims at improving this interaction by making computers more practical and responsive to the user's requests. The improvement on practicality means more efficient and natural to shuttle information by users, while the progress on responsiveness means more effective and pleasant feedback by computers [1]. In the long-run its goal is to propose systems that minimize difficulties between the human's cognitive model and the computer's ability to understand and respond properly. On the human side, communication is conducted by the natural and cognitive abilities. The relevant topics on the human side include design constraints, communication theory, cognitive psychology, and human satisfaction. On the computer side, it is bound only by the technologies and methods that can be invented [2]. The relevant techniques on the machine side are implemented in operating systems, multimedia frameworks, development tools, and programming languages.

Professional designers/researchers in HCI are generally concerned with realistic application to real-world problems, and are involved in developing novel design methodologies, testing with innovative hardware devices, prototyping latest software systems, investigating new patterns for communication, and developing models and theories of interaction. New interaction technologies are among the most active areas of research in this regard.

The substantial growth of computational capabilities in recent years, both for computer hardware and software, has provided a series of new HCI systems that promise to provide a more natural and intuitive way of interacting with computing devices. Motion detection and gesture recognition are among the most popular of these recent technologies. From mobile devices to game consoles, HCI systems show a rapid shift toward the use of vision-based methods to detect user motion and gestures, and replace traditional input methods such as mouse and keyboard with these new alternatives [3]. Within the game industry, this new trend of natural interfaces started with Nintendo Wii and its remote control [4], while Microsoft Kinect changed the face of natural interfaces by making 3D cameras available and affordable, a trend that was followed by similar products from other companies.

According to Merriam-Webster dictionary, *locomotion*, by definition, is:

> "*an act or power of moving from place to place*" [5].

In the substance of locomotion interface, human communicates to the machine through motion inputs, such as pushing buttons, moving a stylus, moving his/her limbs, and so on. An interface system then converts these input motions to some actions through the real or virtual medium, and provides feedback to the user. This feedback is fundamental to update and balance the mental model of interactions [6]. The recent progresses in the Virtual Reality (VR) practical interactions have built a new capacity for locomotion HCI [7].

To conclude this section, research in input and output are oriented around the two ends: 1) the hardware design and technical approach on the machinery side of communication; and 2) the perceptual capacities and common usage of organs constrained by the nature of human attention. Developers then seek to find the salient balance through which the two can naturally be linked and communicate. Fundamental research pursues theories and principles that inform us of the parameters of human cognitive and perceptual facilities, as well as models that can predict or interpret user satisfaction in computing tasks. In this path, the need for new modalities directs the advances to support different application domains. HCI researchers attempt to exert technological breakthroughs in order to improve or develop the capabilities of interfaces, and discover new problems. Even though these approaches are complementary, with their unique contributions to the field, the best findings seem to have a combination of these elements.

1.1. Problem Definition

Fundamental context of HCI is the attitude that human using a computer should come first. For this purpose, the system design should follow the users' preferences and capabilities for performing different tasks. People should not have to change the way they use a system in order to fit in with it. Instead, the system should be designed to match their requirements. Lately the research in HCI is showing a significant focus on creating interfaces that are more user-friendly, by applying natural communication and human skills in the user interface design [8]. Input/output techniques, interaction styles, and evaluation methods are the main challenging fields of research for HCI applications improvement.

In general, two major aspects of HCI systems need extensive study and analysis: functionality and usability. *Functionality* determines features of the device, and evaluates device performance, consistency, and robustness. *Usability* measures the extent to which a product can be taken advantage of by users to achieve specific goals with performance (effectiveness and efficiency) and satisfaction [9]. Functionality is often bound by machine hardware, software, and complex algorithms, while usability depends largely on another complex process: the human experience. Whereas many researchers have been working on natural motion-based interfaces from a functionality point of view [10-14], the focus of this research is on the usability side.

A common pattern observed for many new technologies is the rapid adoption of the "hot new technology", even though it may not be particularly appropriate for some applications. Humans are multimodal and a really "natural" user interface should provide an environment within which different modalities can be used for what they are appropriate. Multimodal interfaces, such as those using different input/output methods, are becoming more popular due to their efficiency and ability to adapt to user's needs [8,15,16]. Although research on multimodal interfaces has been progressive, plentiful questions have left unanswered [17]. Finally, to the best of our knowledge, a consistent and uniform HCI evaluation framework, based on the heuristic relations among the components of interactions, still needs more work, since it is a key asset for researchers and developers to study and predict the major elements of usability among various HCI systems. With the purpose of proposing quantitative measures, this thesis contributes to the formulation of such a consistent and uniform evaluation framework for HCI. In other words, a new evaluation framework is introduced to help in finding answers to the question of "how to represent the relationships between social and technical information in the meaning of usability".

1.2. Motivation of Research

Referring to the problem formulation from the previous section, the concern of UI evaluation is to study the quality of interaction between the user and the artifact. Therefore, the purpose of this research is mainly influenced from the prominences as follows.

- Consolidated and reliable usability analyses are essential.
- Testing factors in a usability study need to be chosen carefully.
- Making things measurable is vital aspect.
- Applying an iterative balance of both methods (human-based design and then technology-based evaluation) can be more promising to achieve a practical and useful application.
- Evaluating in what extent and where the subjective feedback is trustable. This will assist to set up the rules of thumb, showing the relationships among all involved usability factors of study, to be applied in the process of user interface design.
- Validating the rules of thumb based on the correlations among the dependent (sub-factors in satisfaction criteria) and independent (quantitative measures)

usability factors. This method of prediction will reduce the need for massive-questionnaires in the process of user interface evaluation.

- The focus of the research is on kinematic human gestures for two significant purposes: navigating (translation and rotation) and switching, which are found in many other applications such as robotics, education, rehabilitation, smart home, virtual/augmented reality, tele-operation, etc.

1.3. Research Objectives and Expected Contributions

Perceived affordance, introduced by Norman [18], describes a desirable property of a User Interface (UI) or the objects of it that leads users to perform the correct actions and reach their goals. Inspired from this introduction, overall acceptability is defined as the level of *overall-satisfaction* obtained from a survey. On the other hand, the value of usability factors, that include *effectiveness*, *efficiency*, and *satisfaction*, represents how well the users interact with the user interface. Finally, the level of effort to perform a task represents how simple an interaction has been designed based on the complexity of the input and feedback actions. This thesis studies a "demand pull" paradigm (human-driven) under a "technology push" approach (device-driven), where the "demand" is the range of human feedback at the point of interaction. With an available consistent framework, this guidance can also be embodied in automated tools to assist designers [19].

The diagram in Figure 1.1 provides an illustration of our research objectives where:

- observational factors of effectiveness and efficiency are formulated to be obtained from experiments;
- overall acceptability and satisfaction criteria are directly obtained from surveys;
- adjusted satisfaction is extracted from the involved satisfaction criteria;
- a metric for complexity of input/feedback actions is defined;
- some hypotheses are proposed to explain the correlations between technical and social factors, for the purpose of predicting some satisfaction criteria without requiring massive questionnaires in experiments; and

- hypotheses are statistically tested through our three HCI applications designed for an empirical investigation.

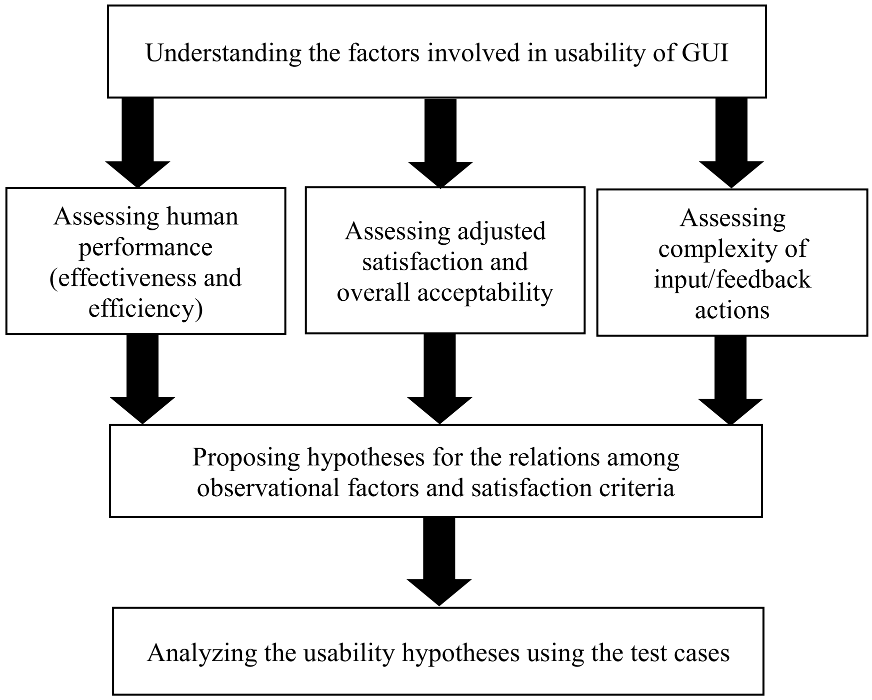


Figure 1.1. Diagram of research objectives.

This dissertation aims to present statistical comparisons of various input modalities and their effects on human performance/satisfaction factors, while interacting with 2D/3D graphical user interfaces (GUI). The focus of this study for input modalities is on vision-based kinematic human gestures, in the forms of full-body, arms, hands, and fingers motion capture. These systems will be compared to classical desktop controllers, in the forms of keyboard, mouse, and haptic 3D controller. Moreover, the diversion of attention/concentration is studied through the usage of different scenarios or devices. Furthermore, the effect of human cognition and ability, when performing various serial and parallel complex tasks in GUI, is among the studies of this dissertation.

Therefore, the main expected contributions of this research include:

a) Formulating some metrics to measure the parameters involved in human usability study, including effectiveness, efficiency, overall acceptability satisfaction, and complexity of input/feedback actions;

b) proposing formal hypotheses for usability evaluation to provide some prediction mechanisms for some social feedback, from the set of overall acceptability and observational factors, such as performance and complexity of interactive actions; and

c) designing three robust 2D/3D GUIs with fairly extensive physics engine and a range of multi-modal sensors that broadly cover most types of locomotion HCI systems, in order to perform three sets of user experiments and verify the proposed hypotheses (test cases).

1.4. Thesis Structure

In Chapter 2 some key literatures in usability studies and user interfaces in the field of HCI are reviewed. Chapter 3 explains usability factors and formulates effectiveness, efficiency, and satisfaction criteria. This chapter examines the components of locomotion in HCI systems, including input modality and feedback of task, and quantifies the actions based on the concept of Index of Complexity (IoC), which is introduced in that chapter. Chapter 4 studies the relations among usability factors of overall acceptability, effectiveness, efficiency, satisfaction criteria, and complexity metrics in UI design, and proposes some hypotheses which provide means for predicting some of these factors from others. Chapters 5 and 6 detail two case studies, of Box and Slingshot applications respectively, to test the proposed hypotheses. These chapters explain the design methodologies, natural gestures selection, gesture recognition modules along with their algorithms to control the UI, experiment processes, and experimental results for further analyses. Chapter 7 details gesture detection/recognition algorithms that are developed to build a third UI, as a Race application, considered for further usability analysis in Chapter 8. The latter deals with the experiment processes, evaluation methods, and experimental results of the third test case, where usability parameters are also analyzed to verify the proposed hypotheses. Finally, in Chapter 9 some concluding remarks are presented.

Chapter 2: Literature Review

Consolidated and reliable usability analyses are essential to improve the ongoing research in HCI, particularly for gesture-based input. Such studies have not yet been fulfilled as they deserve, to shed light on designing a practical interaction between human and machines, and determining the application domains where gesture-based input is more suitable.

In the context of this chapter, Section 2.1 reviews previous works in usability, including general and gesture-based usability. The concepts of complexity/effort and overall acceptability are also reviewed in this section. Section 2.2 focuses on user interfaces, including vision-based, gesture-based, and haptic applications in works of other researchers.

2.1. Usability Study

What happens in the cycle of product design and usability evaluation is a trade-off process that involves choosing which goals or constraints can be relaxed, so that others can be met. Goals are the purposes of a design that we are intending to produce, and constraints are the limitations on the design process generated by external factors.

2.1.1. General Usability

One of the popular standards for product design is the heuristic (*H*) evaluation by Nielsen [20] that builds upon the following considerations:

- *H1*: visibility of the system status to ensure that users are always informed about what is going on;
- *H2*: match between the system and the real world to communicate with words, phrases and concepts that users understand;
- *H3*: user control and freedom;
- *H4*: consistency and standards to enable users distinguish among different words, situations, or actions;
- *H5*: error prevention to avoid problems from happening in the first place;
- *H6*: recognition rather than recall to reduce the users' memory load;
- *H7*: flexibility and efficiency of use to improve the quality of frequent interactions;
- *H8*: aesthetic and minimalist design to exclude irrelevant or rarely needed information;
- *H9*: help users recognize, diagnose, and recover from errors through messages that announce the problem, and advise the solution; and
- *H10*: help and documentation provided with accessible and precise information.

According to Norman's emotional design model [21], there are three levels of processing in human brain, between sensory and motor blocks: *visceral*, *behavioral*, and *reflective*. Visceral design refers to the initial impact of the UI appearance (such as attractiveness, aesthetics, and visual appeal). Behavioral design refers to the total experience of the usage (such as pleasantness, effectiveness, and efficiency). Reflective design refers to the emotional reactions (such as personal satisfaction, memories, relevance of the information, prestige, and intellectualization). The behavioral and visceral are directly affected by the sensory block, and then connect us to the reflective which attempts to bias the behavioral level. The motor block demonstrates how we behave.

In a work by GhasemAghaei *et al.* [22], a framework, called MADE (Multimodal Affect Design and Evaluation), is proposed to address support for *affect* (emotional factors) in multimodal education systems. Their proposal suggests that a learning objective and learning strategies (cognitive, affective and multimodal) should precede system design, and then the system design should involve an HCI approach with both design and evaluation [23,24]. Two inspection techniques of *cognitive walkthrough* and *heuristic evaluation* are adapted in their framework, to explore the effectiveness of the design in some test cases. In the proposed *affective heuristic evaluation* [24], each of the aforementioned levels in Norman's emotional design model [21] is addressed based on Nielsen's heuristics [20] explained earlier: the heuristics of *H8* and *H10* leverage Norman's visceral model; the heuristics of *H1*, *H3*, *H5*, *H6*, *H7*, and *H9* leverage Norman's behavioral model; and the heuristics of *H2* and *H4* leverage Norman's reflective model. Finally, from the qualitative results in the test cases, by GhasemAghaei *et al.* [25-27], the affective heuristics are re-worded to improve the user interactions. Even though the proposed evaluation method is novel, a qualitative study is dependent on the evaluator's interpretation which can be biased and not as precise as a quantitative study.

Even standard user interfaces, with the benefit of consistency, are subject to being out of date, due to technological changes. They run the risk of becoming solely suitable for limited groups of users and tasks [28]. Therefore, the scope of international standards for HCI has been mainly set up on the principles that help to meet user and task needs in user interface design. These standards mostly fall into two categories. a) a "top-down" approach in reference to a usability objective, which is the appropriateness of a product for its intended purpose of use; and b) a product-oriented "bottom-up" view in reference to features of the interface and features of the design that make a system easier to use [29].

According to the usability framework proposed by Nielsen [20], the four basic ways of evaluating user interfaces are: *automatically* (programs run a user interface specification to compute the usability measures), *empirically* (real users test the interface to assess usability), *formally* (usability measures are calculated using exact models and formulas), and *informally* (rules of thumb are set based on the general skill and experience of the evaluators).

On the other hand, the usability framework by Koua *et al.* [30] bases the exploratory visualization and knowledge discovery process on four aspects as: *pattern discovery* (through similarity representations), *hypothesis generation* (from correlations and relationships exploration), *exploration in dataset* (how the data is distributed), and *detection of irregularities* in the data. Their measuring is either based on time (for observational factor of performance) which cannot be general enough to be counted as a common ground among different applications, or based on the survey responses (for satisfaction factors) which is not proposed under one cumulative metric to cover larger number of application types to be compared. Moreover, the iterative process of framework in [30] is fairly complex.

Usability is a property of the complete system. Some attributes of usability, in different studies, are listed as: effectiveness [9][31-37], satisfaction [9,38], usefulness [31,33,39], attitude [31,36,37], learnability [31,33][36-38][40-43], functional correctness [40], efficiency [9,38,40,41,44,45], remembrance [40], error tolerance [38,40], subjective pleasantness [40], easiness [35,42,44], useableness [39], initial and long-term performance [43], retainability [43], advanced feature usage [43], likeability [33], memorability [38], helpfulness [45], adaptability [45], affect [45], controllability [45], and flexibility [36,37].

MacLean [46] introduces the term *expressive capacity* in reference to usability studies, to portray the richness of a communication channel, including its dimensionality or continuity, amount of control, and easiness or comfort. To gauge an interface, expressive capacity includes *density* (information to be transferred), *controllability* (preciseness of transfer), *directness* (contrary to the encoded nature of the desired actions), *responsiveness* (the timely feedback to the user), and *emotiveness* (the quantity of emotion to be extended into the channel). By these measures, the author understands that a computer keyboard is an expressive interface on the first two items, but it does not fit on the third and fourth counts, while the fifth one is context-based. To answer the question “expressive for whom?”, the author gleans that it is also context-based, since some people, for instance, prefer typing while others may prefer to interact verbally or gesturally. Finally, the author comes to the conclusion that haptic interfaces are expressive, since the physical control of tasks feels very natural, needing both emotiveness and precision. She also concludes that due to lack of constraints in an ungrounded gestural interface, it might be very suitable for solely emotive control. Eventually, from her perspective, a keyboard is the best interface that provides the available range of actions with maximum controllability.

On the other hand, according to the terminology for usability in ISO 9241 parts 10 and 11 [9]:

> "*performance and satisfaction represent usability, whereas easy to use interface is a factor that affects suitability for task, individualization and learning, self-descriptiveness, controllability, conformation with user expectations, and error tolerance.*" [9]

According to ISO 9241-11, usability depends on the situation in which a product is used and the framework of use. This situation, as the context of use that may impact the usability factors, is made up of the users, tasks, equipment, and the physical and organizational environments [47].

From these considerations, it can be inferred that the user requirements, user profile, and context of use are the key for success in user interface design and usability evaluation. This conditions that the right usability factors and criteria are addressed in experiments. Loyal to this discussion, the testing factors in a usability study need to be chosen carefully, to extract as much related and suitable information required for the iterative process of a user-centric interface design. Among these factors, efficiency and effectiveness can represent density, controllability, and responsiveness; while the emotiveness and directness can be more precisely broken down into elementary satisfaction criteria, such as naturalness, easiness, non-fatiguing, flexibility, and pleasantness. Moreover, for the purpose of generality, the proposed frameworks and patterns need to freely support any choice of satisfaction criteria that can be applied by usability engineers who are in charge of evaluation.

With regard to the main testing usability factors in ISO 9241-11 [9], Jokela *et al.* [48] claim that effectiveness, efficiency, and satisfaction are fundamental parts of the usability definition, but they are not brought up in the descriptions of the activities. According to [48], it is expected to find instructions for determining effectiveness, efficiency and satisfaction in the description of the activity "*Specify the user and organizational requirements*" [9]. However, the issue is addressed in ISO 9241-11 [9], only with a fairly general statement:

> "*the specification of user and organizational requirements should … provide measurable criteria against which the merging design can be tested*" [9],

where the reference to these measures is missing [48]. Making things measurable is an essential aspect in development cultures where what is measured gets to be done [48]. Moreover, according to the analysis in [48], typical products have different users with various goals, where the levels of effectiveness, efficiency and satisfaction may vary between users and goals. The authors believe that there is a real need in practice to have some kind of systematic way to tackle this challenge of complexity.

Concerning the relationships among the most standardized and accepted usability factors, a usability study of academic digital libraries, by Jeng [49], is conducted from the perspectives of effectiveness, efficiency, satisfaction, and learnability. The developed model considers both quantifying elements such as accuracy rate, time, number of steps in a task, and the subjective criterion of satisfaction. The latter is further broken down into the areas of ease of use, organization of information, terminology and labeling, visual appearance, content, and mistake recovery, with each criterion being evaluated separately. The study examines the interlocking relationships among effectiveness, efficiency, and satisfaction. It finds that the relationships are statistically significant. The longer time is taken for a subject to perform a task, the more steps are involved. The greater number of steps is involved in completing a task, the lower satisfaction is expressed by the subject. The longer time spent on completing a task, the lower satisfaction is also expressed by the subject. Furthermore, the study establishes that demographic factors do not have a statistically significant impact on performance. However, different attitudes among ethnic groups are found in rating satisfaction.

From the source of our empirical studies, these relationships are highly application-based. Referring to the meaning of *user interface*, the concern of evaluation is to study the quality of interface/interaction between the user (social issues) and the artifact (technical issues). Thus, the factors of evaluation need to be chosen from both perspectives: how the user feels about the usefulness of the artifact (*satisfaction* in *visibility*, *naturalness*, *consistency*, *recognizability*, *aesthetic*, and *memorability*), and how the artifact can be technically designed to improve the user's feedback (*effectiveness* and *efficiency* in *error prevention*, *controllability*, and *efficiency of use*). Although, the goal of usability evaluation is a trade-off process, the major purpose of using computer in HCI systems is for what it is better than human: higher speed and less error. By this means, a suitable system receives a higher level of *overall acceptability* from the users when it covers their major demands based on both *satisfaction* and *performance*. However, there is a need for further research to evaluate these two main factors separately in different context-based spaces, where changing the quantity of the involved parameters in space may show a relationship with usability factors. For example, how the user's level of effort in doing a task can affect the easiness of the interaction.

Finally, the authors in [50] first highlight the limitations and complementarities of various standards, and then explain how a single model of usability evaluation can unify different models. The model is called Quality in Use Integrated Measurement (QUIM). Ten factors are included in the QUIM model, where each one correlates to a specific aspect of usability in an existing standard or model. These 10 factors are decomposed into a total of 26 sub-factors or measurable criteria that are furthered decomposed into 127 specific metrics.

Having included some of the fixed factors, as the main concern, to be tested during the usability evaluation does not necessarily address the fitness of all user-centric interfaces, since it must be context-based. For example, the factor of *safety* probably is not the major aspect to study in a game application, while *pleasantness* has more importance. Considering this, it may be more practical to choose a few main factors that offer the least overlap and the most generalization for usability studies, while some sub-factors can be included to be tested under the context of use; for instance, the criterion of *easiness* under the main factor of *satisfaction* in a smart-home application for elderly people. However, a consolidated pattern is required to link the significance level of the sub-factors to the final measurement of their main factors for further evaluation. The best usability evaluation usually comes from a right balance among standards of design, context of use, and flexibility of evaluation.

2.1.2. Complexity and Effort

It is challenging to obtain a full perspective of the nature of human cognition and to apply this knowledge. Therefore, reliable knowledge on cognition requires complimentary theories and replication of experimental results. In mental computation, complexity is mainly related to the memory workload, which in some cases is testable in learning curves, using the physical parameters. The physical parameters, in the forms of multi-dimensional physical units, define the actions. The complexity of mental computation is correlated to the complexity of action. For example, the mental computation is more complex in moving an object under a limited speed (a physical unit of time is added to the distance) than unconditionally (only distance).

As claimed by Mehta in [51], the study of cognitive human factors is concerned with mental processes, such as perception, memory, reasoning, and motor response in the interactions between human and system’s elements. In order to examine human behaviour, ideally physical and cognitive demands need to be considered together. Computer users have trouble learning and remembering information, due to the user's inability to form a mental schema (the big picture) of the information [51]. This problem can be solved in some points by applying the standards of UI design from Nielsen's heuristics, e.g., visibility of system status, and recognition rather than recall. However, more interdisciplinary studies are needed to make some common grounds (in the forms of human factors, context of use, task analysis, I/O device channels, etc.) between cognitive and locomotion HCIs.

According to Vallée-Tourangeau in [52], the affordances, offered by an alterable problem representation, facilitate performance. The simpler and less costly solutions, offered by the physical resources, make them more easily perceived and consequently help counteract mental set. The cognitive effects of some main factors, such as complexity and visibility, on quantified accuracy and efficiency are studied in this thesis.

In the previous section, we have reviewed the 10 elements for product design in the heuristic evaluation proposed by Nielsen [20]. However, there is one more crucial element that is either missed in the above list or merged into other elements: *complexity*. Considering that complexity is different from difficulty, the level of complexity in the concept of this study– which is different than the computational complexity in computer science– is not necessarily correlated to the quality of interaction, and is application-based. For example, a complex action clinically set for rehabilitation purpose can be more usable than a simple, but irrelevant, action. Having the main factors of usability (overall acceptability, effectiveness, efficiency, and adjusted satisfaction) evaluated in an experiment, the next step for redesigning a user interface is to find out which of the "10+1" elements (ten heuristic elements plus complexity) are related to the main factors of usability requiring improvement. For this purpose, some metrics are needed to quantify these elements. Although some elements, such as "aesthetic design", cannot be objectively quantified, others may be measured. An important question arises now: Are there any formal metrics to measure the complexity of HCI systems which affects the satisfaction of use? In this section, more literature is reviewed to show the significant effect of complexity in usability study, and investigate how it can be measured.

In a study by Schlick *et al.* [53] on mobile HCI, a quantitative complexity theory of HCI is validated through some user experiments. The measure, termed the Effective Measure Complexity, has three main advantages: 1) it relies on information-theoretic quantities connected with the concept of complexity; 2) it can be estimated efficiently from data; and 3) the estimates can be derived from observable interaction events. Multiple targets are searched on a marine simulator's display, by 30 participants, to study the effects of various platform motion and light conditions in the workplace. The results in [53] show that the complexity of the visual search is inexorably lower when the simulator is controlled with straining motion forces by users. Also, interaction complexity is greatly lowered when lighting is changed from daylight to twilight level. Finally, the authors in [53] conclude that the complexity measure seems to be credible for the measurable assessment of HCI.

According to [54], the task complexity is evaluated through use of quantitative methods as part of morphological analysis. This information is valuable in order to predict the mental effort required whilst carrying out the task and also to determine the critical characteristics of the task. Representing the physical effort, the weight of the task is a significant property of manual tasks, where it is easier to measure objectively. However, there is also a subjective element as a feeling of physical stress, dependent on the physical conditions of a user. Physical and mental efforts can influence one another.

In Rudolf Laban's Theory of Effort [55,56], the dynamic nature of actions and the relationship among movement, space and time are studied. This approach tries to assign the main features of user movement without zoning in on a type of movement particularly. The basic idea of Laban's theory is to consider regard effort as the movement's characteristics. From an engineering perspective it can be seen as a vector of components showing the qualities of a movement performance. This effort vector is seen as having a four dimensional space with the axes of space, time, weight, and flow. According to [57,58], out of the four Laban's dimensions, the main spotlight needs to be on two: space and time. These dimensions are chosen due to a) their successful implementation in previous analysis on human full-body movement; and b) their relevant display of key features for familiar human gestures. Space relates to the path and direction of a succession of motion movements. The space element is described as "flexible" or "direct" if the movement follows these directions evenly or along a straight trajectory respectively. Time, on the other hand, relates to a vacillating portrayal which allows the binary description of the time element in the effort space: an action can be "quick" or "sustained".

In a work by Boletsis [7], virtual reality (VR) locomotion is investigated through a systematic literature review of empirical studies in the latest advancements. The review analyzes the locomotion techniques and their interaction-related characteristics. According to [7], results show that the locomotion research has been mainly focusing on technological aspects; thus, overshadowing the evaluation of user experience. From this review, the author in [7] proposes four distinct VR locomotion types: motion-based, room scale-based, controller-based and teleportation-based locomotion. The movement in the first three types is continuous, while the last one is classified for non-continuous actions. He concludes that, there is a need for more user-centric research under comparative settings.

Finally, according to [7,59], the locomotion technique physiologically affects user perception. This perception refers to performance and objective measurement for time, distance, and speed in VR surroundings. Therefore, providing a mutuality to allow for the contrasting and analysis of these techniques is vital.

From the aforementioned literature, we infer that the physical efforts need to be measured, particularly in locomotion HCI systems, should they take an important role in complexity of systems. The mental efforts, associated with decision making, are prominent when a new action needs to be initiated. Therefore, a concept in the context of HCI, that represents both mental and physical efforts, is required to measure the complexity. This concept also needs to conventionally define the intensity of actions in the measure of complexity. The new concept is introduced in this thesis to address these characteristics of complexity: a) to be quantified under a simple model in different dimensions which define physical efforts; and b) to represent the transitions among different actions which affect mental efforts.

2.1.3. Gesture-based Usability

Since the focus of this study is partially on the effect of user effort on usability in locomotion HCI systems, this section studies some of the literature related to human gestures and vision-based applications.

3D-gestures, such as touch-less hand gestures and body movements are more and more used in human-computer interaction. Although gesture-based user interfaces have been broadly investigated, developing natural and ergonomic 3D-gesture interactions is still challenging. The usage of appropriate gestures, considering the physiology of human body and the users' goal, the investigation of suitable UI widget types and presentation, and the development process are some of the works in this field. The contrast between technology-based and human-based approaches is introduced by Nielsen *et al.* [60]. In technology-based solutions, gestures are first implemented under the constraint of the available technologies, and then evaluated. This method may be undesired from the user perspective; and consequently can be costly to modify, if practicable yet. In human-based methods, on the contrary, the gestures are proposed by users prior to the implementation. Users are asked to demonstrate gestures that should be implemented. However, this time, the selected gestures may not be realizable by current technology. Therefore, applying an iterative balance of both methods (human-based design and then technology-based evaluation) can be more promising to achieve a practical and useful application [61].

As proposed by Bomsdorf and Blum [62], gesture interaction is developed in three design concepts: presentation, gesture, and dialog. They present a tool, ProGesture, designed based on presentation-gesture-dialog space. Through their findings, they infer that the more range for design exists on each dimension, the larger space for finding a solution is available. The authors in [62] come to the conclusion that the presentation, more precisely the *perceived affordance* [18], considerably impacts the gestures that users will utilize.

In order to add a feeling of real interaction with objects, Frohlich *et al.* [63] use some physics libraries of simulated Newtonian forces to move, distort or bounce an object based on its type and how the user is approaching it. However, the semantics of gestures is not gathered in this system. With the hypothesis of having an object as a direct receiver of the gestures, Boulabiar *et al.* [64] organize the multi-touch gestures into four principal families (touch, move, rotate, and scale), that are detected by computing some properties as the number of fingers, speed and time. Prior to this work, Reisman *et al.* [65] factorized gestures into rotate, scale, and translate categories.

In another work, by Rouxel *et al.* [66], the dynamic morphology of the gesture is represented through analog gestures, in time-space pattern (TSP), with the features of speed, acceleration, direction, and angle. For each criterion, the assigned value is either continuous or discrete according to human capability. They also suggest that analog gestures are multi-touch motions when depicting relative evolutions of multiple trajectories is the objective. They conclude that drawing a direction by human hand is a continuous parameter, as it comes with fairly good accuracy. However, they assume that since its speed is not accurate, it can only be considered as zero, slow, normal and fast, and therefore is a discrete criterion.

As mentioned earlier, assuming robust technical systems, the usability aspect of gesture monitoring systems is the second parameter in need for an in-depth study. Therefore, there has been significant usability study in the field of vision-based modalities. In this regard, the combination of hands and line of sight (LoS), achieved through gaze detection, is considered by the authors in [67] as the interaction method. They conclude that this method can lessen the fatigue compared to a one-hand pointing interaction.

Cabral *et al.* [68] discuss numerous usability issues associated with the use of gestures as an input mode. A 2D computer vision based gesture recognition system is introduced by the authors to be used for interaction in some virtual reality environments. Their results illustrate that it is more time consuming, as well as more fatiguing to complete simple pointing tasks than using a mouse. However, according to their findings, several advantages are revealed by the use of gestures as a substitute in multimodal interfaces. These include immediate access to computing resources using a natural and intuitive way, and that balances properly to joint applications, where gestures can be used infrequently.

Villaroman *et al.* [69] suggest using Kinect in classroom through examples that demonstrate how Kinect-assisted instruction can be utilized to accomplish certain learning results in Human Computer Interaction (HCI) courses. Moreover, the authors have confirmed that OpenNI, in addition to its accompanying libraries, are adequate and beneficial in enabling Kinect-assisted learning activities. For students, Kinect and OpenNI offer a hands-on experience with its gesture-based, natural user interaction technology.

The adaptation of three free hand techniques for menu selection is put forward by Bailly *et al.* [70]: Linear menu, Marking menu and Finger-Count menu. The experiment illustrates that Finger-Count is more mentally demanding than other techniques.

In a study on 3D applications using Kinect, Kang *et al.* [71] introduce a control method that naturally regulates the application with the use of distance information and joints' location information. Furthermore, the gesture recognition rate is more successful, as well as the use of the proposed gestures in the 3D application, which is 27% quicker than a mouse.

Code Space, introduced by Bragdon *et al.* [72], is a system that combines touch and air gesture hybrid interactions to jointly carry small developer group meetings. This method enables access, control and sharing of information, through several different devices, combined with multi-touch screen, mobile touch devices, and Microsoft Kinect sensors. In a formative study, professional developers are positive about the interaction design, and most feel that pointing with hands or devices, and forming hand postures, are socially acceptable.

According to Bhuiyan and Picking [73], a gesture-based user interface application, named Open Gesture, is available for standard tasks, for instance making telephone calls, operating the television, and executing mathematical calculations. This prototype uses a television interface to carry out various tasks by using simple hand gestures. Based on a usability evaluation, the authors in [73] recommend that this technology can improve the lives of the elderly and disabled users by creating more independence while some challenges still remain to be overcome.

An experimental study by Ebert *et al.* [74], on touch-free navigation through radiological images, measures the response period and the practicality of the system compared to the mouse/keyboard control. The image creation time using gestures is 1.4 time longer than using mouse/keyboard. However, it does remove the risk of infection, for both patients and staff.

Having some related works in the field of usability study reviewed so far, the next section will discuss about user interfaces. This topic is important for designers to create functional, effective, simple, and user friendly applications that will be of great support in achieving the objectives of user experiences in usability studies.

2.2. User Interfaces

Among the first hand gesture detectors, mechanical devices have been used to capture information from a hand gesture [75]. One example of this early technology includes data glove devices, which collect the information generated from the movement of the fingers and transmit it to a computer system [76,77].

2.2.1. Vision-based

Over the past ten years, the performance of computer hardware has become significantly enhanced while units have steadily decreased in price. This improvement in technology has resulted in the gradual replacement of data glove devices by vision-based hand gesture technology. Vision-based technology does not require users to wear a device, making their gestures more natural because there are no limitations in the movements of the hand. It is also very user-friendly, which is essential in any human-computer interaction. Given that vision is one of the core physical media, vision-based technology is more desirable than wearable devices, such as the data glove device, in hand gesture recognition systems [78].

2.2.2. Gesture

Humans, and many other living organisms, have been employing motions of limbs for expression. The broad definitions for gestures in biology and sociology allow researchers to describe the gestures proprietarily [79]. Kendon [80] classifies a gesture as: gesticulation (impulsive movements of hands/arms during speech), language-like gestures (replace words/phrases), pantomime-like gestures (depict objects/actions), emblems (common gestures), and sign languages (sets of gestures/postures defining linguistic communication systems). From gesticulation to sign languages, the association with speech decreases and the influence of social parameters increases.

When interacting with computers, gestures can be utilized through vision-based techniques. Hand movements and poses such as pointing, grabbing, and moving, are extremely intuitive and content rich, both spatiotemporally and perceptually. They are therefore perfect means for interaction purposes [17]. For eventual gesture utilization in an HCI system, modeling, analysis, and recognition need to be designed precisely [81].

Although static arm/hand gestures, also called hand postures, have been the main focus in the gesture recognition field, recently researchers are showing more interests in incorporating dynamic gestures in their study, due to a broader domain of hand's dynamic actions comparing to hand postures [82].

The objective of research in hand gesture recognition is to develop ways in which the human hand can be utilized as an interface for human-computer interaction. The shape, position and/or movement of the hand are parameters that are analyzed by vision-based hand gesture recognition systems. Specifically, hand gestures can be described by four main characteristics including hand configuration, palm orientation, hand position and hand movement. The flex angles of the fingers and the orientation of the palm are used to model static hand gestures, whereas hand trajectories and orientation are also needed to model dynamic hand gestures. Therefore, to accurately model dynamic hand gestures, the interpretation of dynamic gestures based on hand movement, shape and position must be appropriate. A series of hand gestures, that in their entirety bear some meaning, are defined as continuous gestures.

2.2.3. Haptic

Haptics denotes the human's sense of touch for feeling or manipulating a virtual object. Haptics has been supported by a collaboration of various disciplines such as psychophysics, neuroscience, biomechanics, robot design, modeling and simulation, and software engineering. A wide range of applications have emerged and span many fields such as product design, education, video games, graphic arts, medical training, and rehabilitation [83].

A system, capable of recognizing human gestures and providing haptic feedback, is considered a major step towards a more natural and viable multimodal system. For example, in an augmented reality system to play table tennis, it would be ideal if the users could provide the system with controlling commands using either speech or gesture and feel force feedback when they hit the ball using the racquet. In such systems, human gestures play a critical role, which needs to be preferably detected, classified, and interpreted through computer vision and pattern recognition means in order to avoid "non-natural" sensors.

With regard to the main forms of feedback, haptic devices can be classified into three groups [82]: ground referenced force feedback (e.g., Phantom device [84]), body referenced force feedback (e.g., CyberGrasp [85]), and tactile feedback (e.g., CyberTouch [85]).

Considering the continuous contact of the user with an haptic device in most haptic systems, the user's perception of the virtual environment is hindered, and it is also impossible for the user to feel a new tactile sensation. The latter drawback hampers the integration of tactile and force feedbacks in one haptic device [17,86,87]. Moreover, most haptic devices have a very limited workspace [88,89]. Therefore, constant contact in such a limited space impairs to incorporate rich interaction elements in an extended virtual environment.

2.3. Summary

The traditional HCI, such as using mouse/keyboard, presents a narrow variety of actions to the user, while also its touch-based control becomes more difficult on smaller devices. In addition, in some applications, employing conventional controllers makes the interactions more cumbersome, whereas using direct sensing of human hand gesture is naturally more suitable. Recent advances in computer vision, and more specifically hand modeling, tracking, and gesture recognition have made this new modality possible. However, gesture recognition software remained a challenging issue. Noise, inconsistent lighting, items in the background, multiple subjects, distinct features, and equipment limitations are some of the main constraints associated with image-based gesture recognition [90]. To achieve the required accuracy, robust computer vision methods are highly needed.

In order to evaluate what can be achieved in creating synthetic experiences, technology development and quantitative investigations are indispensable. Having this combination, preferably in a multimodal system, would help to assign the right balance of input methods to different output tasks in order to build an interactive system between human and machine as much natural and effective as possible. The research presented in this thesis aims at developing a consistent framework, which conforms the evaluation and prediction mechanisms for usability study of user interfaces in locomotion HCI systems.

Chapter 3: Usability Factors in Locomotion Interface

A solid, comprehensive, and uniform framework is the key for a better analysis of experimental results. Therefore, in order to evaluate the usability of locomotion HCI applications designed, a number of metrics are proposed in this chapter. They are grouped into four categories: 1) human performance, 2) human satisfaction (users' feedback), 3) overall acceptability, and 4) effort-based design. The first two factors are sourced from ISO 9241-11 [9], whereas the last two are proposed by this research. Human performance represents the observational factors such as effectiveness and efficiency, obtained from the user experiment. Human satisfaction compiles the users' opinion about some satisfaction criteria, obtained from a questionnaire after the user experiment. Overall acceptability is also a perceptual result that is the users' opinion about their overall satisfaction, again obtained from a questionnaire filled after the user experiment. Finally, the effort-based design represents the level of complexity, obtained from the actions in input modality and output task executed during user experiments.

3.1. Human Performance

This section proposes two indexes as measures of effectiveness and efficiency, which are, we believe, the core metrics in the final analysis of UI involving a moving object in an analog task, or a still object in a discrete task.

As reviewed in Chapter 2, for formal experimental evaluation, physical responses of users, in the forms of duration, number of total versus successful trials, object deviation, etc., are needed to measure the two major factors of performance:

- ***Effectiveness:*** how precise a given task can be accomplished?
- ***Efficiency:*** how fast a given task can be accomplished?

As reviewed in the related works, Boletsis [7] has proposed four locomotion types of motion-based, room scale-based, controller-based and teleportation-based locomotion. For the purpose of uniformity, this thesis has logically re-contextualized the HCI locomotion types in three types of ***distance-based*****,** ***action-based***, and ***mixed*** (of the first two), as they are generally found in various applications in the field of locomotion HCI. This classification is important and prerequisite to measuring the major factors of effectiveness and efficiency, since they are affected differently under such application-based categories. Distance-based interactions involve a displacement over a certain distance while remaining on a predefined trajectory (e.g., balance of movement in Parkinson disease). Action-based interactions are related to performing a predefined movement (e.g., sign language). Finally, mixed interactions involve a combination of distance-based and action-based interaction features (e.g., driving a car using a manual transmission system).

As mentioned, a distance-based type is considered when only analog movements are involved. In this case, effectiveness is commensurate with the stability of the motion while efficiency is measured with motion velocity. On the other hand, an action-based type is considered when only discrete movements are involved. In this case, effectiveness is commensurate with the accuracy of the performed movement while efficiency is measured from the frequency of action. Finally, a mixed type is considered when both analog and discrete movements are involved. In this case, depending on the dominant effect of stability versus accuracy, effectiveness can be considered as distance-based or action-based respectively. Similarly, depending on the dominant effect of velocity versus frequency, efficiency can be considered as distance-based or action-based respectively. For example, when moving a virtual object in a narrow path, in which it has to be discretely resized and continuously passed through the path, the dominant effect on effectiveness comes from accuracy, as it mainly affects the activity to be completed or not. In the same example, the dominant effect on efficiency comes from velocity, as it mainly affects the duration of activity to be completed.

Let us suppose an object needs to be moved/ relocated in a 2D/3D path, as shown in Figure 3.1. A metric to quantify the stability of movement or accuracy of the action, in the path is needed. The travel distance, defined as the displacement (shortest distance, *SD*), is the length of the straight line between the start point and the end point in an analog distance-based motion. In a similar way, in an action-based interaction for jumping from the start point to the end point, the travel distance is replaced with a discrete definition, that is the minimum number of actions (*NA*) required to relocate the virtual object. For example, one gesture can be used for designating the object to be moved and one gesture to specify the destination, *NA* would be 2 in this case. Furthermore, in an analog distance-based scenario, *RM* is the real motion of moving object to complete the task in a total time of *TT*. In an action-based scenario, however, the analog real motion in the distance-based interaction is replaced with the number of trials (*NT*), as the actual number of discrete actions including errors, taken by the user to complete the task.

The main equations proposed to estimate effectiveness and efficiency are developed in the following subsections.

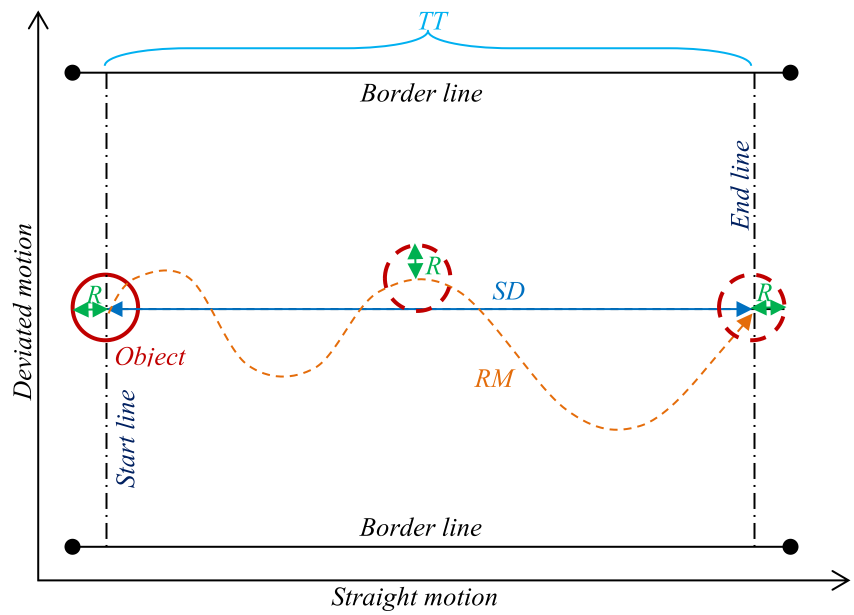


Figure 3.1. A moving object and the elements involved in a locomotion interaction.

3.1.1. Effectiveness

The Stability Ratio (*SR*) can be considered as a measure of *effectiveness* in a task that involves distances. It can be similarly represented as Accuracy Ratio (*AR*) in a task that involves actions or a mixture of distances and actions. They are defined with the following equations:

$$SR = \frac{SD}{RM} \qquad (Distance - based) \tag{3.1}$$

$$AR = \frac{NA}{NT} \qquad (Action - based) \tag{3.2}$$

where *SD* is the shortest distance to complete a distance-based activity, *RM* is the actual length of motion to complete a distance-based activity, *NA* is the number of actions as the minimum number of steps expected to complete an action-based activity, and *NT* is the number of trials as the actual number of steps the user takes to complete an action-based activity. From aforementioned, the metric of effectiveness (E_s) is defined with the following equations:

$$E_s = \begin{cases} SR & (Distance - based) \quad (3.3) \\ AR & (Action - based) \quad (3.4) \end{cases}$$

3.1.2. Efficiency

The Activity Velocity (AV, in *distance/second*) of the object is proposed as a measure of *efficiency* to complete a distance-based task involving a displacement over *SD* during a period *TT*. It is transformed in an Activity Frequency (*AF*, in *trial/second*) to complete an action-based task involving a minimum number of steps expected, as *NA*, during a period TT. The AV and *AF* are defined using the following equations:

$$AV = \frac{SD}{TT} \qquad (Distance - based) \tag{3.5}$$

$$AF = \frac{NA}{TT} \qquad (Action - based) \tag{3.6}$$

On the other hand, the distance involved in the metric of velocity and the size of objects in the interface has two flaws when it comes to comparing results among different applications involving distance measurements: 1) distance may have different units in various applications; 2) the size of an object to be manipulated by the action, that can be defined via its radius (*R*), may be different in various applications, while this needs to be unified not to influence the assessment of efficiency. For this reason, the metric of efficiency (E_y) is defined by the following equation:

$$E_y = \begin{cases} \frac{AV}{R} & (Distance - based) \quad (3.7) \\ AF & (Action - based) \quad (3.8) \end{cases}$$

The physical interpretation of the E_y metric is that the size of the manipulated virtual object is used as the unit metric to measure displacement.

3.2. Human Satisfaction

According to ISO 9241-11 [9], *satisfaction* is freedom from discomfort, and positive attitude to the use of the product. In a usability study protocol, it is usual that after each test, participants are asked to fill a questionnaire. The questionnaire evaluates how the users rate, from 1 to 7 (low to high), each interaction/activity. In the proposed frameworks, the same strategy is used to measure soft satisfaction parameters, as proposed in our previous work [8], and defined in Table 3.1 as *rate of satisfaction criteria*. Depending on the application design, any other satisfaction criteria, such as level of stress or concentration required, can be included.

After evaluating the interactions based on the satisfaction criteria, the participants are also asked to rate the level of importance for each of the five satisfaction criteria, from 1 to 7. The more important satisfaction factors, in the opinion of the user, get the higher rates. These extra parameters are named as *weighted satisfaction criteria* as listed in Table 3.1.

Table 3.1. Satisfaction factors considered in a GUI experiment.

Human factor		Value	Description
Rate of satisfaction criteria (X)	*Easiness (E)*	$1\ to\ 7$, answered by users	How easy the activity is
	Fatigue (F)	$1\ to\ 7$, answered by users	How non-fatiguing the activity is
	Naturalness (N)	$1\ to\ 7$, answered by users	How natural/intuitive the activity is
	Pleasantness (P)	$1\ to\ 7$, answered by users	How pleasant the activity is
	Mobility (M)	$1\ to\ 7$, answered by users	Ability to move freely during the activity
Weighted satisfaction criteria ($\tilde{X}$)	*Easiness ($\tilde{E}$)*	$1\ to\ 7$, answered by users	How important easiness is while interacting with the UI
	Fatigue ($\tilde{F}$)	$1\ to\ 7$, answered by users	How important fatigue is while interacting with the UI
	Naturalness ($\tilde{N}$)	$1\ to\ 7$, answered by users	How important naturalness is while interacting with the UI
	Pleasantness ($\tilde{P}$)	$1\ to\ 7$, answered by users	How important pleasantness is while interacting with the UI
	Mobility ($\tilde{M}$)	$1\ to\ 7$, answered by users	How important mobility is while interacting with the UI

3.2.1. Data Normalization

In order to refine the results and acquire an unbiased set of coefficients, the *weighted satisfaction criteria* are normalized per user through the following equation:

$$\tilde{X}_n = \frac{\tilde{X}}{\underset{1 \le i \le N_x}{\operatorname{avg}} \ \tilde{X}_i} \tag{3.9}$$

where $\tilde{X}_n$ are the *normalized weighted satisfaction criteria* ($\tilde{E}_n$, $\tilde{F}_n$, $\tilde{N}_n$, $\tilde{P}_n$, and $\tilde{M}_n$) per user, and $\tilde{X}$ are the *weighted satisfaction criteria* ($\tilde{E}$, $\tilde{F}$, $\tilde{N}$, $\tilde{P}$, and $\tilde{M}$) rated by each user, with $i \in$ total number of satisfaction criteria, which is $Nx = 5$ in the experiments of this research.

3.2.2. Adjusted Satisfaction Criteria

For the next step, it is required to adjust the users' level of satisfaction on different criteria (direct feedback) with the normalized weights of their importance based on the users' opinions. Therefore, the *adjusted satisfaction criteria* (indirect feedback, $\hat{X}$) are produced per user ($\hat{E}$, $\hat{F}$, $\hat{N}$, $\hat{P}$, and $\hat{M}$) through the following equation:

$$\hat{X} = X \times \tilde{X}_n \tag{3.10}$$

where X are the *rate of satisfaction criteria* (direct feedback) per user (E, F, N, P, and M), and $\tilde{X}_n$ are the *normalized weighted satisfaction criteria*, as explained before.

3.2.3. Adjusted Satisfaction

Finally, a new cumulative rate (adjusted *satisfaction*, S_a), is defined per user, as an average of *adjusted satisfaction criteria*, through the following equation ($i \in$ total number of satisfaction criteria, with $Nx = 5$ in the experiments of this thesis):

$$S_a = \underset{1 \le i \le Nx}{\text{avg}} \hat{X}_i \tag{3.11}$$

3.3. Overall Acceptability

Overall acceptability (*OA*) means "the overall quality of being accepted". Therefore, it properly represents the meaning of overall satisfaction, as how the user satisfactorily perceives (and gives a final score) about the interaction with the user interface. This new term, proposed in the field of HCI, includes the effects of effectiveness, efficiency, and all other involved human satisfaction factors. To explain the overall acceptability in the field of social life by example, an adventurous trip may be more satisfactory overall, even if it takes longer or is more dangerous than a regular trip. This can be due to the fact that some factors may take significant roles to prefer this trip over the other. Therefore, all quality factors need to be evaluated under this final perceptual feedback to find those leading factors among different experiences.

In the field of HCI, the overall acceptability is measured by the users' direct feedback (in the range of 1 to 7, as the lowest to the highest) through the questionnaire after the experiment. In other words, the adjusted satisfaction (S_a) is calculated merely based on the human satisfaction factors of study, while the overall acceptability (OA) is the users' final feedback (overall satisfaction) as the most significant user-centric usability scale, based on all possible factors that affect the quality of interactions.

3.4. Effort-based Design

In the previous sections of this chapter, some technical and social usability factors, that model the general view of user interfaces, were proposed and quantified. Even though they are useful to form a big picture of the design in relation to the capability of the system, they do not address some important issues in the UI design. Many usability factors are influenced by the level of effort that the user needs to invest for performing a task. Therefore, a good design in HCI systems needs to consider how the complexities of lower-level elements in an interaction, such as the input modality and feedback of task, affect the quality of interaction. For this purpose, a complementary framework is proposed in this section to analyze the usability of design from the perspective of effort and complexity in the user-centric evaluation.

3.4.1. Locomotion Effort Model

As shown in Figure 3.2, an HCI system can be represented by its two counterparts, *human* and *machine*, each made of three components [91,92]. The internal states of each interact in a closed-loop system through *control-display* (the machine interface) and *motor-sensory* behavior (the human interface).

The terms *input* and *output* typically relate to the machine. Input provides the physically sensed information about the environment to machine. Output emits the electronically processed information of machine to the environment. The two worlds of digital and human senses are connected through input and output bridge [1].

Input devices are controlled or manipulated by human outputs. By convention, human outputs are provided through our limbs. *Output devices* emit or transfer the information processed in the user interface to the human inputs (such as eyes, ears, and skin) [92].

Interaction takes place at the middle-line between the human and machine blocks, through the output channel and the input channel in order to perform the tasks.

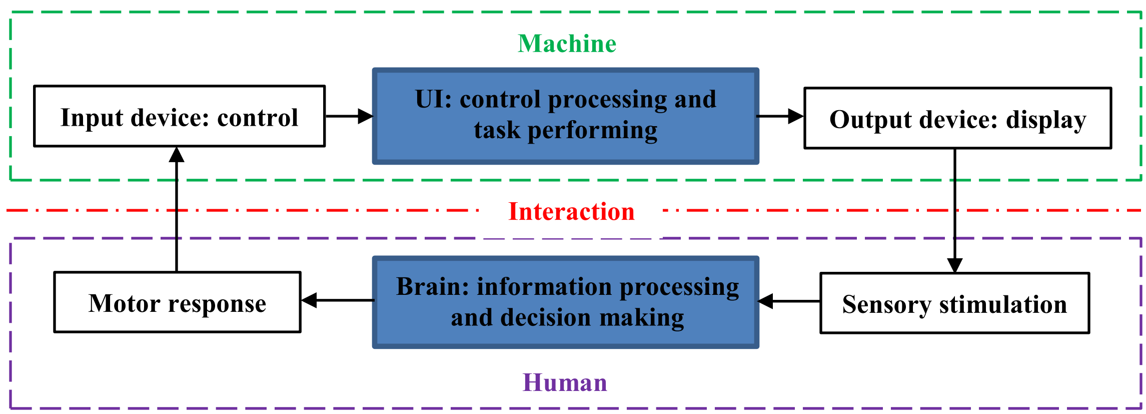


Figure 3.2. The general scheme of an HCI system.

Inspired from the general scheme of HCI systems (Figure 3.2), the components of Locomotion Effort Model are described as follows: 1) *modality* represents the actions performed at "motor response" in the human block; 2) *feedback* represents the actions sensed at "sensory stimulation" in the human block; 3) *I/O device* represents the total dimensions of input and output devices at "control" and "display" channels (mediums for communication or the passage of information) in the machine block; and 4) *overall acceptability* represents the amplitude of usability factors obtained from the user experiment of performing a task. This latter corresponds to the "interaction" middle line between the human and machine blocks, which is in fact affected by the relations between the input modality and feedback in human block. Therefore, this model can be simplified by considering only the two factors of modality and feedback affecting some usability factors and consequently the overall acceptability of interaction.

The elements of modality, feedback of task, I/O device, and interaction are divided into different types, as detailed in Figures 3.3 to 3.6 respectively. Particularly, for I/O device, the quality of interaction may also be impacted when the user's attention is divided among multiple receivers/transmitters. Therefore, the I/O device parameter is subdivided in four types: 1) single input, single output; 2) single input, compound output; 3) compound input, single output; and 4) compound input, compound output. The single input case corresponds to a single control device that receives the user's actions, while the compound input case collects the user's actions through multiple controls. Similarly, the single output case corresponds to a production medium that solely stimulates the user's sensory input. The signal in a compound output case, however, is combined with some other ambient signals that cause the diversion of attention from the user's main task. With this definition, a desktop monitor can be categorized as a single output. However, a large-screen display (a screen with extended field of view) can be considered as a compound output, since the user may sense its signal perturbed with the presentation of other information in the peripheral vision.

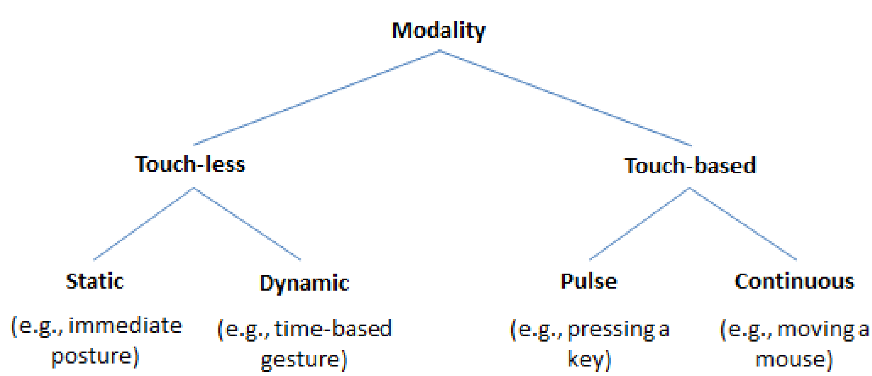


Figure 3.3. Types of modality.

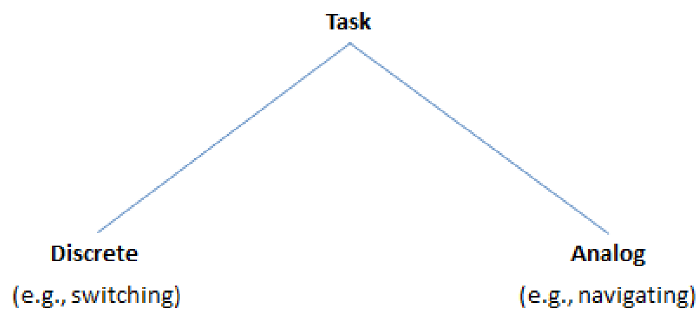


Figure 3.4. Types of task.

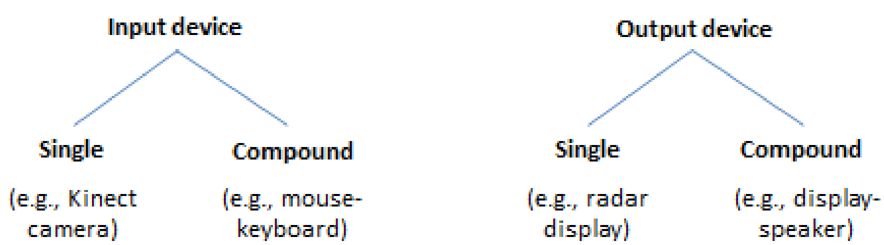


Figure 3.5. Types of I/O device.

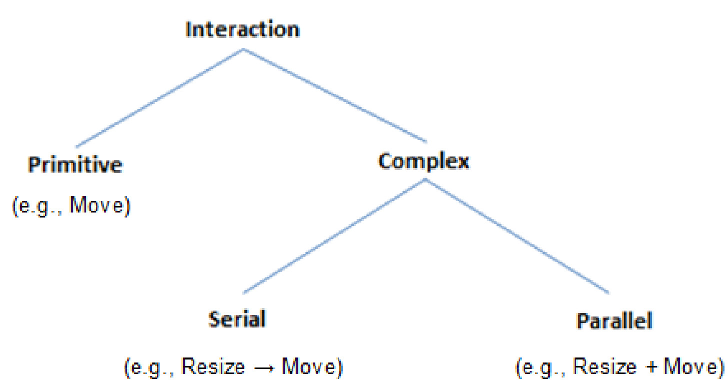


Figure 3.6. Types of interaction.

To conclude this section, employing a multitude of perceptual channels, through multimodal interfaces, humanizes interactions with computers. Concerning the modality-related psychology theories, we select different modality conditions (vision-based input, haptic control, keyboard and/or mouse, and output devices with different field of view). We also study the cognitive effect of different modalities on performance, using a high-load information presentation scenario (complex interactions, including serial and parallel feedback of tasks).

3.4.2. Complexity: A Metric for Measuring Effort

From the reviewed literatures in Chapter 2, complexity is defined as the level of effort that the user needs to invest for performing a task with the elements of input modality and output feedback. Therefore, the number of components in the actions of modality/feedback, and the relationships among those components need to be measured.

Punching and grabbing are two similar actions with regard to the extension of the arm. However the timing and intensity of the motions are very different [93]. Taking this fact into consideration, some terms are defined as follows.

Actions are the procedures to follow for doing something to achieve the desired aims, such as moving or rotating the hand. ***Interaction*** is the corresponding set of actions of input modalities and feedback required to perform a task, such as moving a joystick to drag a virtual object. Complex interactions (also named ***activity***) can be in the forms of ***serial*** (an activity performing a series of tasks, one after another) or ***parallel*** (an activity performing a combination of tasks, all at the same time).

3.4.2.1. Index of Complexity

Any action has a given number of degrees of freedom (DOF). These refer to the number of possible joint angles, length of links, velocity, acceleration, external forces, movement distances, etc. The task is controlled by the input modality in interaction with the receiving feedback. Table 5.1 enlists some common kinematic quantities, in the form of *parametric DOF*, that are used for further analysis of actions in modality and feedback levels. This table includes most of the actions that can happen in a locomotion interaction.

An Index of Complexity (*IoC*) is proposed in this section as a metric of action analysis. It measures the complexity of actions (level of effort) in modality and feedback, and is obtained from the following equation:

$$IoC = \sum_{i=1}^{S} (I_i + P_i) \tag{3.12}$$

where P_i is the number of parametric DOF, $(DOF)_{Parametric}$, for any involved basic-quantities (as shown in Table 3.2), I_i is the level of intensity for any basic-quantities (e.g., the action of running has a higher level of velocity compared to walking), and S is the number of basic-quantities involved in a modality or feedback action. This equation is explained in more details as follows.

Table 3.2. DOF: an effort-based metric for the input/output actions.

Basic-quantity		Dimension	Unit	$(DOF)_{Parametric}$
Unit	*Length*	n = 1	L	(one unit of ‘L’) = 1
	Radian	n = 1	r	(one unit of ‘r’) = 1
	Time	n = 1	T	(one unit of ‘T’) = 1
	Mass	n = 1	M	(one unit of ‘M’) = 1
	Action	n = 1	A	(one unit of ‘A’) = 1
Static/Discrete action		n = number of actions	A	n × (one unit of ‘A’) = n
Frequency		n = number of cycles in a second	T^{-1}	n × (one unit of ‘T’) = n
Translation		n = 1, 2, 3	L	n × (one unit of ‘L’) = n
Rotation		n = 1, 2, 3	r	n × (one unit of ‘r’) = n
Velocity		n = 1, 2, 3	LT^{-1}	(n × (one unit of ‘L’)) + (one unit of ‘T’) = n + 1
Angular velocity		n = 1, 2, 3	rT^{-1}	(n × (one unit of ‘r’)) + (one unit of ‘T’) = n + 1
Acceleration		n = 1, 2, 3	LT^{-2}	(n × (one unit of ‘L’)) + (two units of ‘T’) = n + 2
Angular acceleration		n = 1, 2, 3	rT^{-2}	(n × (one unit of ‘r’)) + (two units of ‘T’) = n + 2
Impulse		n = 1, 2	MLT^{-1}	(one unit of ‘M’) + (n × (one unit of ‘L’)) + (one unit of ‘T’) = 1 + n + 1 = n + 2
Force/Weight		n = 1, 2, 3	MLT^{-2}	(one unit of ‘M’) + (n × (one unit of ‘L’)) + (two units of ‘T’) = 1 + n + 2 = n + 3

In order to better discriminate between different actions, the level of intensity, *I*, is associated to three levels (low = 0, medium = 1, high = 2) based on the ratio of the *designed intensity* to the *maximum possible intensity* in a basic-quantity. Each range of the result, set to [0 to 1/3], [1/3 to 2/3], and [2/3 to 1] respectively, represents its own level of intensity (0, 1, and 2). For instance, the maximum velocity of an impulse motion of hand is about 3.5 m/s [94]. On the other hand, the velocity threshold for recognizing the simple action of "push" (basic-quantity of impulse) is set to 2 m/s in the UI design [8][15]. Having the maximum possible intensity with the value of 3.5, and the designed intensity with the value of 2, then the ratio will be 0.57. Since this value is within the second level of intensity range [$\frac{1}{3} < 0.57 < \frac{2}{3}$], then the level of intensity will be medium ($I = 1$).

Having the level of intensity (*I*) explained above for an action of "push", the *IoC* of input and feedback for the task of Grab & Hit, as an example in a Slingshot application [8], is quantified as follows.

The procedure of dynamic input actions consists the following sequences: showing the right-hand palm (as an activation action in 1 dimension, with $P_1 = 1$ for one unit of action), moving hand (as a trajectory control in 3 dimensions, with $P_2 = 3$ for three units of distance), drawing a virtual velocity-controlled planar circle (as a grabbing control in 2 dimensions, with $P_3 = 3$ for 2 units of distance and one unit of time in velocity), moving hand (as a dragging control in 3 dimensions, with $P_4 = 3$ for three units of distance), and pushing hand (as a dropping control in 1 dimension, with $P_5 = 2$ for one unit of distance and one unit of time in velocity) which has the level of intensity ($I_5 = 1$) as explained in the previous paragraph. Inserting the P_i and I_i from above into Equation 3.12 will calculate the final IoC of input with the value of 13.

Through a similar procedure, the feedback actions consist the following sequences: showing the pointer (as the center of palm in 3 dimensions, with $P_1 = 3$ for three units of distance), getting pointer into the collision area of object (as a distance control in 3 dimensions, with $P_2 = 3$ for three units of distance), selecting object (as a grabbing action in 1 dimension, with $P_3 = 1$ for one unit of action), translation of object (as a dragging control in 3 dimensions, with $P_4 = 3$ for three units of distance), deselecting object (as a dropping action in 1 dimension, with $P_5 = 1$ for one unit of action), shooting object (as a velocity control in 1 dimension, with $P_6 = 2$ for one unit of distance and one unit of time in velocity), getting object into the collision area of target (as a distance control in 3 dimensions, with $P_7 = 3$ for three units of distance), and hitting/passing the target (as a scoring action in 1 dimension, with $P_8 = 1$ for one unit of action). Inserting the P_i and I_i (nothing in this case) from above into Equation 3.12 will calculate the final IoC of feedback with the value of 17.

3.4.2.2. Simplicity and Similarity

As explained in the beginning of this chapter, human satisfaction is one of the main factors to be evaluated in usability study. Depending on the type of application, it includes some human factors. Among these factors, *easiness* and *naturalness* are popular in the venue of user feedback. By definition, easiness (“freedom from difficulty or hardship or effort”) and naturalness (“the likeness of a representation to the thing represented”) correlate to *simplicity* and *similarity* respectively. In other words, simpler and similar interactions most likely cause performing of the tasks to be felt easier and more natural respectively. Since the two factors of simplicity and similarity are used in kinematic fields, they can be objectively measured based on the complexity of modality and feedback actions involved in locomotion HCI. In the context of locomotion HCI, modality action is what human needs to do to perform a designed task (motor response block). Feedback action is what happens in the application layer, in the meaning of motion analysis, to be sensed by human (sensory stimulation block).

Being inspired by the definition of easiness, *Rate of simplicity* (*RoS*), as presented in the equation below, is intuitively the ratio of the complexity of actions in the feedback of task over the corresponding complexity of actions in modality. This new factor represents the level of simplicity to perform a task using a modality. In real life, the lower number of actions to do a task makes it easier (e.g., showing the emblem of "thumb up" with one action, compared to 9 times key striking to text "good job!"). In other words, applying a less complex modality to do a task (higher RoS) means that the interaction is simpler compared to a more complex modality.

$$RoS = \frac{IoC_{feedback}}{IoC_{modality}} \tag{3.13}$$

Similarly, being inspired by the definition of naturalness, *Index of similarity* (*IoS*), as presented in the equation below, intuitively defines how close the two complexities in modality and feedback actions are. Also, in real life, the closer nature of actions in modality and feedback represents a more natural interaction. In the application layer, breaking down the number of actions in modality and interpreting them to the corresponding actions in feedback can take a main role as a natural interaction (e.g., using a steering-wheel to turn a virtual car on a curve, compared to pressing L/R for left/right). In other words, a higher IoS of interaction means more similar/natural input and output actions are involved (the negative sign is inserted to direct the values in a positive correlation to the meaning of naturalness).

$$IoS = -\left|IoC_{feedback} - IoC_{modality}\right| \tag{3.14}$$

In order to understand these two new metrics better, the following example is presented.

As shown in Figure 3.7, let us suppose that there are four alternatives of modality with the IoC values of $M_1 = 5$, $M_2 = 8$, $M_3 = 15$, and $M_4 = 20$, for one feedback of task with IoC value of F = 10. Lower IoC of input actions ($IoC_{modality}$) for a specific IoC of feedback ($IoC_{feedback}$) increases the RoS, as depicted in Figure 3.7, which means producing higher simplicity (less IoC of input actions makes it easier to complete a task). On the other hand, the best natural interaction is when IoC of input actions ($IoC_{modality}$) and the IoC of feedback ($IoC_{feedback}$) have the same value. Therefore, farther IoC of input actions on the left/right side of feedback IoC decreases the IoS and naturalness consequently.

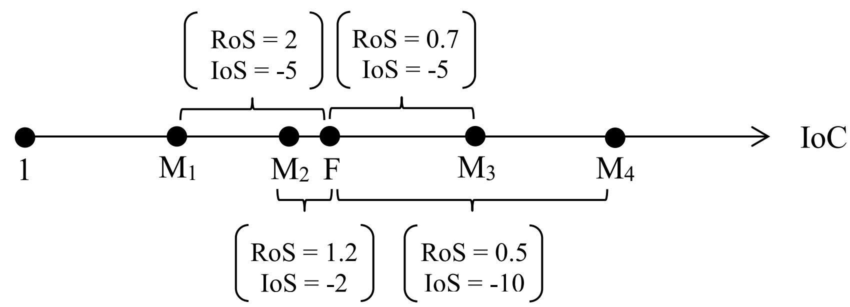


Figure 3.7. An example for simplicity and similarity.

3.5. Discussion

In order to make the HCI locomotion types more uniform, three categories have been proposed in this thesis: depending on the dominant effects of stability/accuracy and velocity/frequency, the effectiveness and efficiency are listed under either action-based or distance-based, while a mixed interaction includes a combination of both types for effectiveness and efficiency.

This chapter proposed different metrics required to measure the main usability factors of performance (effectiveness and efficiency), satisfaction (including the satisfaction criteria and adjusted satisfaction), overall acceptability, and effort-based design in the interactions of locomotion HCI systems. Compared to the other models such as ISO 9241-11, introducing the above-mentioned factors in a new model supports the usability evaluations in a more user-centric approach, where the factors of performance and satisfaction are evaluated with regard to the added factor of overall acceptability. Moreover, the proposed effort-based model provides a new solution to measure the physical or mental complexity, where the latter happens in a transition between two interactions and is included in this new model. After collecting measurements in a given HCI test scenario, the equations presented in this chapter are applied to quantify a number of usability factors, in order to consequently verify the hypotheses proposed in the next chapter.

This representation characterizes a UI in terms of the nature of available interactions. In other words, the proposed model examines the correlation for goodness of fit, in order to find the proper elements of interaction and related usability factors. Finding the correlations between the technical and social usability factors leads us to a reliable formalism that facilitates the selection mechanism for UI design. Eventually, creating a large database of user interfaces from a variety of HCI systems can offer a rich source of information to support the evaluation and prediction of usability by developers, prior to the UI design in HCI systems.

Chapter 4: Usability Framework

Robust modeling approaches are key to predict usability problems and subtle aspects of the user experience [95]. A proper framework to support usability studies should follow the logical flow from raw data collection, information analysis, and conclusion on actions to be taken. In order to fully take advantage of usability parameters introduced in Chapter 3, this chapter studies the relations among the factors of effectiveness, efficiency, simplicity, similarity, and overall acceptability with some satisfaction criteria such as easiness, non-fatiguing, naturalness, pleasantness, and movability of interactions. This formal usability evaluation framework provides an affordance-based selection mechanism on the uniform results of the main usability factors to facilitate the analysis of locomotion HCI systems before, during and after development process. The model involved in this framework represents simplifications of the actual phenomena involved in HCI by forming the quantitative basis for evaluation. It aims to make it easier for designers to predict and evaluate alternative designs while preserving HCI intrinsic characteristics.

A general model of usability is proposed in [17], where acceptability and ergonomics have been considered as the main factors affecting the usability. It is expanded by additional sub-factors, such as learnability, ease of use, fit to task/user, efficiency of use, satisfaction, etc. for each of the main usability factors. According to Nielsen [96], usability is about learnability, efficiency, memorability, errors, and satisfaction. However, as discussed in section 2.1.1, the definition of usability from ISO 9241-11 [9] which states that usability is:

> "*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use*" [9]

– is becoming the main reference. This definition of usability shows that one needs to determine the following outcomes in an iterative user-centric design: 1) users of the system; 2) goals of users; 3) environments of use; and 4) measures of effectiveness, efficiency and satisfaction [97].

4.1. Design Considerations

According to Holzinger *et al.* [98], the principal motivations to design a consolidated model and a form of structured representation for usability analysis are to:

- "*reduce the costs of usability testing by providing a common basis for understanding and comparing various usability metrics;*
- *complement more subjective, expert-based evaluation of usability;*
- *provide a basis for clearer communication about usability measurements between software developers and usability experts; and*
- *promote usability measurement practices that are more accessible to software developers who may not have strong backgrounds in usability engineering*" [98].

A coherent model should support the exploration of the relations among sets of factors, criteria, and metrics. It should also be flexible and generic enough for different types of applications. Having a guideline for interpretation of the collected data can be very useful to make informed decisions about how to evaluate the usability of particular applications in a specific context. Since the ISO 9241-11 standard is globally recognized, we selected the basic ISO 9241-11 standard (with usability factors of effectiveness, efficiency, and satisfaction) as the baseline for the proposed usability evaluation model, plus two new contexts of evaluation: overall acceptability and complexity in interactive design.

Finding the significant affecting factors can address where the software developers need to work on to improve the product design. From the view of product design and its standard in Nielsen's heuristic evaluation [20], among the mentioned predictors, effectiveness is related to some evaluation criteria such as "error prevention", while efficiency is related to some others such as "flexibility and efficiency of use" [50]. On the other hand, complexity is a standalone metric of design which is set and known before the experiment. This factor is also related to some evaluation criteria such as "aesthetic and minimalist design" or "match between system and real world". Finally, overall acceptability, as a pure user-centric factor, can be estimated even before performing a user experiment. This can be done through preparatory surveys, in an early stage of the product design based on the users' main needs and desires. Consequently, the main users' requests, as the source of their overall acceptability, address what evaluation criteria need to be worked on.

From aforementioned, the significant affecting factor is an interface that links where the software developers need to work on to the final goal of improving a specific satisfaction criterion. For instance, if improving the error prevention leads to a more effective design, and if more effective design leads to an easier usage of the product, then it is inferred that improving the error prevention leads to an easier usage of the product. Therefore, in the process of iterative design, the starting points of focus can be properly indicated in order to predict an improvement on a social feedback of users (as the opinions of participants in a survey about the quality of the content).

The next section proposes a set of usability hypotheses to predict some social feedback of users based on the technical results. Having such relations well defined should greatly facilitate usability evaluations with excluding some challenging and time consuming parts of social feedback through the surveys in the user-centric experiments.

4.2. Correlation, Causality, and Validation Process

Studying correlation and causality allows for systems to be targeted with desired consequences. Therefore, that is the base of the validation process on the proposed hypotheses presented in the next section. For a better understanding, more details are provided as follows.

To begin, we need to know what *correlation* and *causation* are, and how they are different. Correlation statistically measures the strength and direction of a relationship among variables. However, it does not automatically describe causal relationships among the variables. Causation, on the other hand, points out that one variable is the cause of the change in the values of another (cause and effect). Correlation-related questions (such as "is there a relationship between the level of education and health of people?") discover if there is a correlation between the independent variable and the dependent variable. When such a correlation exists in an observational study, then further research is required to investigate the causal relationship between the variables [99]. Since establishing causality does not have a statistical method generally, the validation process needs to be using either case-based methods or based on hypothesized arguments. The latter is the method of research in this dissertation [100].

In the validation process using the hypothesis method, there are two main terms of *null hypothesis* and *alternative hypothesis* that need to be stated first. Simply put, a null hypothesis (H_0) states that there is no relationship between two variables (what researchers try to disprove); whereas an alternative hypothesis (H_A) states that there is a statistically significant relationship between the two variables (what researchers try to prove). If the null hypothesis is not rejected, then the alternative hypothesis is rejected; and vice versa.

After collecting data, a certain amount of variation is expected to be caused by chance. However, there are some types of *statistical tests* to verify whether a particular pattern can define the cause for an effect. The predictions in these tests are supported by the *probability* levels in the results. For example, by using the 5% as the level of *significance* (probability equal to or less than 5 in 100 comparisons), the level of *confidence* can be indicated. Such a high level of confidence means that the data provide the evidence (not proof) that there is a cause for an effect, and it is not by chance. In other words, when there is strong association between variables, it is tended to assume that this is due to causality.

In order to verify a hypothesis, stating the relationship between multiple independent variables and a dependent variable, the validation process is as follows [101-106].

a) Correlation-test: finding the meaningful correlations (*weak*: $0.25 < |r| \leq 0.45$; *moderate*: $0.45 < |r| \leq 0.75$; and *strong*: $0.75 < |r| \leq 1$) between the testing variable (Y) and the corresponding 'i' number of independent variables (X_i) of the hypothesis;
b) F-test: performing a multiple linear regression analysis on the X_i with meaningful correlations to find a significant F-value ($F < 0.05$), which means there is less than $(F \times 100)\ \%$ chance that the null hypothesis is true; and
c) T-test: if the F-test, in the previous step, shows a value less than 5% (which means the null hypothesis is rejected, and alternative hypothesis is confirmed), the P-values of each independent variable show the significant effects of them on the dependent variable ($p < 0.05$ means a significant effect).

This is where the analysis ends and a discussion, based on the samples of experiment, starts to interpret the statistical findings in practice. The last steps of the process (regression equation) validate the conservatively proposed hypotheses. However, the first step (strong correlations) still can be considered remarkably for general, and even though non-significant, predictions.

4.3. Proposed Hypotheses

As explained in Chapter 3, *overall satisfaction* is termed as *overall acceptability*, while *adjusted satisfaction* is measured as a combination of the involved satisfaction criteria such as easiness, pleasantness, naturalness, etc. In addition, it has been explained that effectiveness, efficiency, and adjusted satisfaction are the main usability factors affecting overall acceptability in a user-centric design. With this reminder, the base of hypotheses, proposed in this section, is explained intuitively as follows.

It is intuitively perceived that an interaction is easier when it is simpler, performed faster, and satisfied more overall. It also feels less fatiguing when it is simpler and more satisfactory overall. On the other hand, an interaction intuitively feels natural when it is performed in a similar way of doing that in the real world. In other words, the feeling of naturalness is improved when the related actions of input and output are correspondent, and the overall satisfaction is also gained. Moreover, an interaction is perceived to be more pleasant when it is performed with less errors and more overall satisfaction. When it comes to the factor of movability in performing an interaction, there are various parameters that may affect it. The parameters such as diversion of attention, range of workspace, and cognitive perception can be named as some of them. Since these parameters affect the overall satisfaction, it is inferred that an interaction feels intuitively more flexible when it is more satisfactory overall. Finally, it is intuitively perceived that an interaction is more satisfied cumulatively when it feels better overall and has a higher performance (more effective and efficient).

From aforementioned, this dissertation studies some applications in the field of locomotion HCI, with the interactions designed under various settings of input modality, feedback of task, and output device, in order to verify the intuitive hypotheses of H1 to H6 as follows.

- **H1: Easiness**
  a) **$H1_0$ (null hypothesis):** Easiness is unrelated to usability factors of overall acceptability, efficiency, and simplicity.
  b) **$H1_A$ (alternative hypothesis):** Easiness is related to at least one of the usability factors of overall acceptability, efficiency, and simplicity.
- **H2: Non-fatiguing**
  a) **$H2_0$ (null hypothesis):** Non-fatiguing is unrelated to the usability factors of overall acceptability and simplicity.
  b) **$H2_A$ (alternative hypothesis):** Non-fatiguing is related to at least one of the usability factors of overall acceptability and simplicity.

- **H3: Naturalness**
  a) **$H3_0$ (null hypothesis):** Naturalness is unrelated to the usability factors of overall acceptability and similarity.
  b) **$H3_A$ (alternative hypothesis):** Naturalness is related to at least one of the usability factors of overall acceptability and similarity.
- **H4: Pleasantness**
  a) **$H4_0$ (null hypothesis):** Pleasantness is unrelated to the usability factors of overall acceptability and effectiveness.
  b) **$H4_A$ (alternative hypothesis):** Pleasantness is related to at least one of the usability factors of overall acceptability and effectiveness.
- **H5: Movability**
  a) **$H5_0$ (null hypothesis):** Movability is unrelated to the usability factor of overall acceptability.
  b) **$H5_A$ (alternative hypothesis):** Movability is related to the usability factor of overall acceptability.
- **H6: Adjusted satisfaction**
  a) **$H6_0$ (null hypothesis):** Adjusted satisfaction is unrelated to the usability factors of overall acceptability, effectiveness, and efficiency.
  b) **$H6_A$ (alternative hypothesis):** Adjusted satisfaction is related to at least one of the usability factors of overall acceptability, efficiency, and efficiency.

The purpose of these hypotheses is to facilitate the usability evaluations, based on the possible correlations among some social feedback and observational results. Since all hypotheses have a controlling independent variable of overall acceptability in common, if it is confirmed that the flow of overall acceptability is similar to the ranking of social factors (specially for the critical values with significant differences), then it is inferred that the user opinion in such primitive surveys is reliable to predict overall acceptability. The aforementioned ranking, as explained in Section 3.2, is the average values of *weighted satisfaction criteria* ($\tilde{X}$) obtained from the survey. Consequently, their other opinions about the dependency of satisfaction criteria on the observational controlling factors can be also valid. From this end, further adjustments in the design, concerned to the controlling independent variables included in the hypotheses, may need to improve the dependent variables.

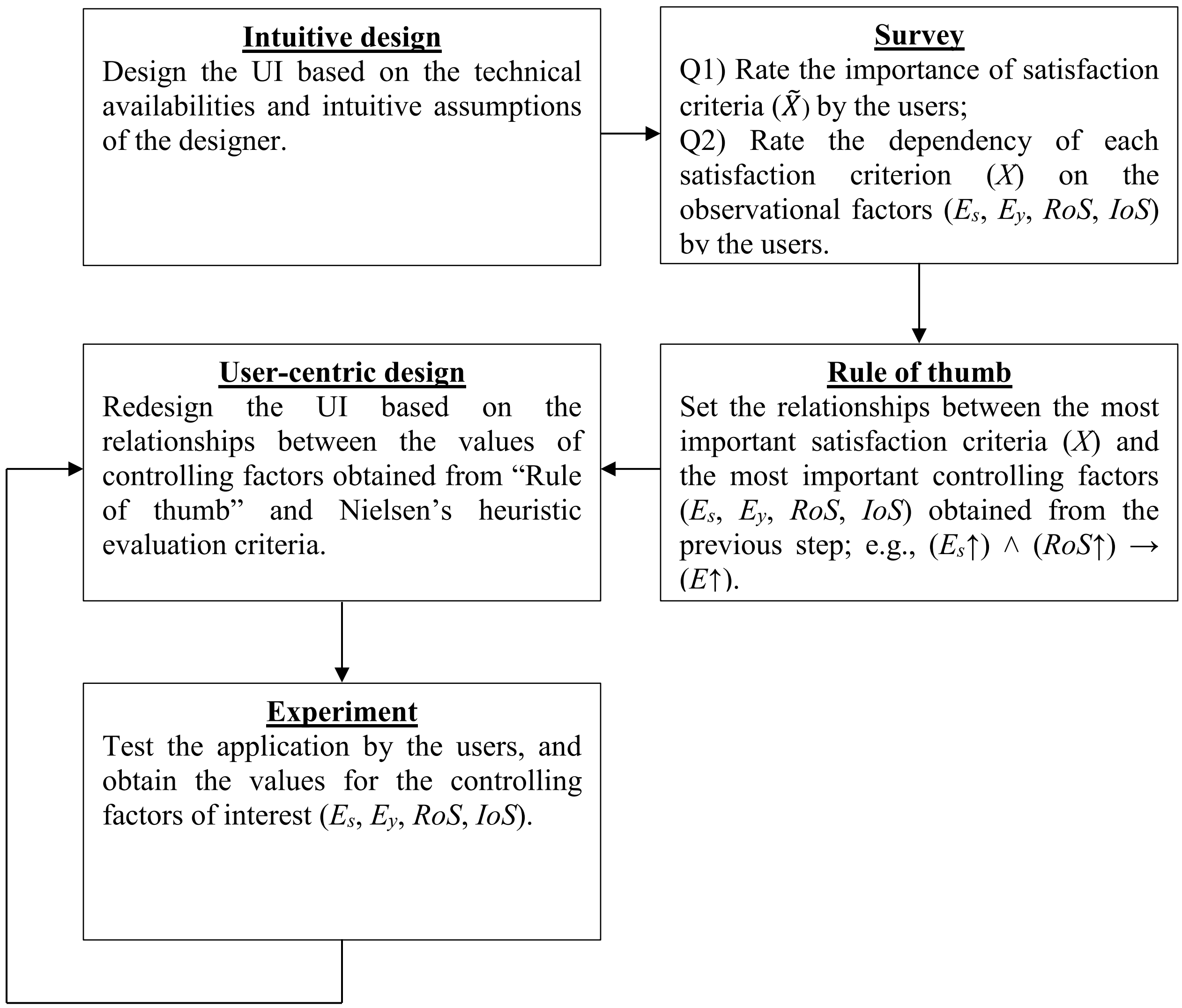


Figure 4.1. Scheme of usability model for the UI design.

Figure 4.1 shows a simplified scheme of usability model for the UI design, proposed by this thesis. The proposed hypotheses have been defined intuitively, by the author of this dissertation. However, for a legitimate process of iterative design, they need to be defined by the users of the usability study, as shown in Figure 4.1.

4.4. Overview of the Test Cases

In this thesis, graphical user interfaces (GUI) have been designed in the context of games. A generic GUI provides two ways of interacting with a system; these are: navigating and switching. The former deals with the analog changes (distance-based) such as translation and rotation of the controlled objects in the scene. The latter provides discrete manipulations (action-based) required for triggering actions such as selecting an object or switching a key on/off. Having these two interactive components, such GUI can represent many applications in various fields such as game, robotics, education, rehabilitation, smart home, virtual/augmented reality, tele-operation, transportation, etc.

The usability studies performed in this thesis analyze the impact of different input modalities and output feedback on different human satisfaction's criteria (such as easiness, non-fatiguing, naturalness, pleasantness, and movability), some observational factors (such as effectiveness and efficiency), and the complexity factors (such as simplicity and similarity). In order to perform a comprehensive study, this research analyzes three applications, on which usability studies have been conducted (Table 4.1).

Table 4.1. The three applications (test cases) designed for usability study.

<table>
<tr><th>Application</th><th>Activity</th><th>Activity repetition</th><th>Associated task</th><th>Observational/ human factor to evaluate</th><th>Input modality</th><th>Experiment</th></tr>
<tr><td rowspan="2">1- Box game</td><td>Step 1: Grab the box by a combination of actions</td><td>2</td><td>- Look (aim)
- Move (navigate)
- Jump (hit)</td><td rowspan="4">- Time (efficiency)
- Precision (effectiveness)
- Easiness
- Fatigue
- Naturalness
- Pleasantness
- Movability
- Overall Acceptability
- Adjusted Satisfaction</td><td rowspan="2">Full body
vs.
Keyboard/Mouse</td><td rowspan="2">Participants: 20</td></tr>
<tr><td>Step 2: Grab the boxes by reaching towards</td><td>5</td><td>- Grab (move & hit)</td></tr>
<tr><td>2- Slingshot game</td><td>Hit the target by shooting at (grab & hit)</td><td>5</td><td>- Navigate (move)
- Select (grab)
- Drag (pull)
- Deselect (release)</td><td>Hand
vs.
Fingers
vs.
Haptic device</td><td>Participants: 20</td></tr>
<tr><td>3- Race game</td><td>Control an object on a path under various scenarios (activation, city-road, road-break, narrow-road, bridge, cruise-control, free-way, and deactivation)</td><td>1</td><td>- Select (pick)
- Limited-speed Move (drag)
- Jump (switch)
- Parallel Resize & Move
- Serial Resize & Move
- Rotate
- Shoot and Guide (look)
- Deselect (release)</td><td>Fingers/Hand
vs.
Keyboard/Mouse</td><td>Participants: 20</td></tr>
</table>

In the first application (Box game), the impact of vision-based full-body gestures on usability factors is compared with the impact of traditional keyboard/mouse desktop controllers. Because full-body gestures are used here, which necessitates a distant camera to cover the whole size, shape, and location of the limbs, a Kinect depth camera has been used to interact with the application. Large-screen display and desktop monitor have been also included as output devices in this study.

In the second application (Slingshot game), the impacts of vision-based static and dynamic hand gestures on usability factors are compared with the impact of an haptic 3D desktop controller which applies the force feedback to the UI design. A Kinect depth camera is also used here to interact with the application, since it provides a precise 3D tracking of objects.

Having the findings of the first two applications, a third multimodal application (Race game) has been conceived that includes a physics engine simulating factors such as speed and acceleration. In this case, the usability of vision-based hand/finger gestures with a cheaper hardware (a single web camera) is compared with the usability of desktop controls (mouse and keyboard).

Through the implementation of these three applications and conducting their user experiments, the next chapters aim to show how the proposed usability model can be applied in practice.

4.5. Discussion

Concerning the nature of human cognition, there are two crucial issues [107]:

- *Ecological validity* where results apply outside the laboratory, requiring the methods, materials, and study to approximate the examined real-world task.
- *Generalizability of the results* where results can be applied outside of specific context of study, requiring thorough control of experiments with generalizable principles of human cognition and performance.

Optimal study should utilize ecologically valid methods and produce generalizable results (e.g., depending on the type of experiment and context of use, any human factors can still be included in the final evaluation of adjusted satisfaction). The practice of cognitive ergonomics included in our UI designs and user experiments are as follows.

- Number of objects/information to be kept in mind (simple/complex)
- Visual information and disruptions (desktop/large screen)
- Visibility (good/poor)
- Overlapping modalities (mouse & KB)
- Task switching (serial/parallel)

Finally, from the same people conducting the experiment, the correlation data are under different components of modality, task, and I/O devices separately. However, they can be grouped by averaging their values (e.g., comparing all interactions with desktop display vs. large-screen). Including I/O devices in the list of context in use is for the purpose of further analysis on the cognitive behavior of interactions (e.g., factors of disruption and overlapping controls in compound devices).

Chapter 5: Case Study of the Box Game

In this chapter, the quality of interactions is studied, while users perform some common tasks in a 3D Box game, using two input methods: full-body gesture and traditional keyboard/mouse. It reports on experiments with three different variables: input modality (full-body gesture vs. traditional mouse/keyboard), output modality (large projection screen vs. traditional desktop screen) and the task to be performed (various actions common in games such as looking, moving, jumping, and grabbing). Microsoft Kinect is used as the 3D sensor for gesture detection. The objective is to verify our hypotheses proposed in Chapter 4. For this purpose, some experiments are carried out for further analyses on different evaluation criteria related to performance, satisfaction, complexity, and overall acceptability.

5.1. User Interface General Design

The proposed system has been designed under the following settings:

Operating System: 32-bit Windows 7 SP 1; Camera: Microsoft Kinect depth camera located at about two meters from the user for the best results (based on Kinect's technical characteristics); Development tools: Microsoft XNA and Kinect SDK; Displays: a 1080p projector with a screen size of 92 inches located at about seven meters from the user used for the large screen, while a 24 inches LED monitor located at less than a meter from the user used for the small display (Figure 2.1).

Figure 5.1. The system's layout for the Box game.

The experiments went through a study consisting of two steps: Step 1(Figure 5.2-top row) included a series of tasks of different types and their related gestures. Step 2 (Figure 5.2-bottom row) was designed based on the results of the first step in order to optimize the use of the gesture-based option, i.e. the tasks that seemed not suitable for gesture-based input were removed and the remaining ones for gesture input were redesigned.

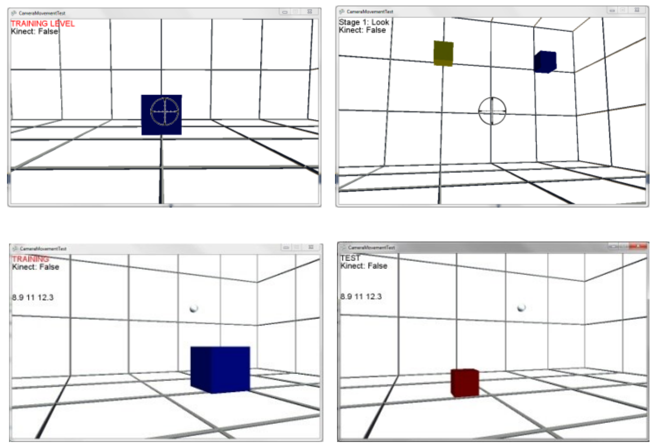


Figure 5.2. User interfaces for Step 1 (up), and Step 2 (down) in the Box game.

Step 1 included a combination of three basic tasks in a test game: look, move, and jump actions. The user has to use either the keyboard or the limb (arm/waist/legs) gestures to hit virtual boxes by performing look/move/jump actions, in the 3D environment. There were 2 target boxes in each level that the user had to hit.

Step 2 included only one single task that revolved around selecting targets in a 3D environment. The 3D environment used either keyboard/mouse or Kinect input to allow the user to move a cursor in 3D space to select target boxes by colliding the small 3D spherical cursor with the target boxes. Once one of the boxes was collided with, another would appear. There were 5 target boxes in total that the user had to hit.

5.1.1. Vision-Based Module

For each of the two steps corresponding to the tasks, body gestures were used. Gesture definition was done by considering common gestures used in Kinect games. For Step 1, this included the following three gestures (Figure 5.3-a, -b, -c):

- *Look:* The user controls the Look by moving the position of their right hand. Moving it left and right will pan the camera left and right and moving it up and down will tilt the camera up and down. The Look gesture was created using the Windows SDK where the position of the users hand joint returned by the Kinect was subtracted from the position of the users shoulder joint in order that the distance of the users hand could be found in the X, Y and Z directions. These distances were then mapped to the camera in the 3D environment so that the farther the user moved their hand away from their shoulder, the faster the camera would move in that given direction.

- *Move:* The user controls the Move gesture by leaning. Leaning left and right will move the camera in the left and right directions and leaning forward and backward will move it forwards and backwards. The Move gesture was created using the Windows SDK where the angle is found between the position of the user's chest and a perfectly vertical straight initial line. This enables the degree to which they are tilted away from the vertical in the X and Z directions to be returned. This angle was then mapped to the camera in the 3D environment where if the angle was between a certain range (5 and 45 degrees) the camera would move in that direction at a speed that factored in how far the user was tilted.
- *Jump:* The user controls the Jump gesture by physically jumping up. The gesture was defined so that it would have to obtain a certain upward velocity in order for the jump to be triggered. The Jump gesture was created using the Windows SDK where the velocity of the user's waist joint in the upwards Y direction is found through taking the distance it has moved over a period of one second in order to find out how fast it is moving per second. This velocity is then mapped to the camera in the 3D environment where if the upwards Y velocity of the person's waist is above a threshold it triggers the jump action in the camera.

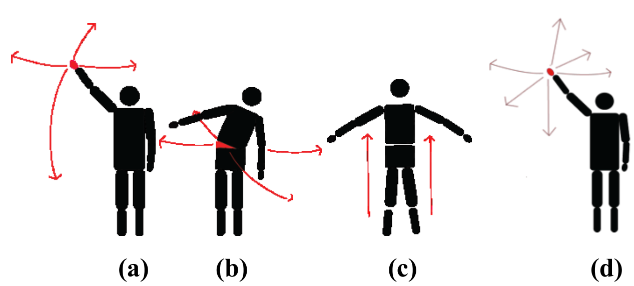


Figure 5.3. Gestures; a) Look by Arm (2D), b) Move by Waist (2D), c) Jump by Legs (1D), and d) Grab by Arm (3D).

For Step 2, only one gesture was defined that would combine the look and move actions into a 3D movement (Figure 5.3-d):

- *Grab (3D Movement):* The user controls the 3D cursor by moving the position of their right hand. This is similar to the Look task in Step 1 but performed in 3D. The position of the right hand is tracked and compared to the location of the right shoulder to figure out the distance the user is reaching. This information is stored as a three dimensional vector where the distance is broken down into its XYZ components. Each distance component is stored as a number from 0 to 1 (zero being the hand in the same place as the shoulder, one being the hand at its maximum reach distance).

5.1.2. Touch-based Module

The keyboard, used in this experiment, was the standard Qwerty keyboard. Instead of being assigned to bodily gestures on the Kinect, the various movements were assigned to the mouse and keyboard keys.

- *Look:* To look the user presses the up and down arrow keys to tilt the camera up and down, and the left and right arrow keys to pan it left and right.
- *Move:* To move the user uses the 'W' key to move forward, the 'S' key to move backwards, the 'A' key to move left, and the 'D' key to move right.
- *Jump:* To jump the user presses the Spacebar.
- *Grab (3D Movement):* The Up and Down arrow keys control the Z position of the cursor moving it in and out. The X and Y position of the cursor was controlled by the mouse (in Step 2).

5.2. User Experiments and Analysis

For Step 1, there were five main stages created as follows (Figure 5.2-up):

- *Training Level* - This was the practice level and was not part of the tests.

- *Look* - The first simple task in where the user was tasked with only having to look (align the crosshair with) at the two different boxes to complete the test.
- *Move* - The second simple task in where the user was tasked with only having to move and collide the camera with the two different boxes to complete the test.
- *Jump* - The third simple task in where the user was tasked with only having to jump that is properly executing the jump command so that the avatar would jump to hit the two different boxes.
- *Combination (Look + Move + Jump)* - The fourth task is a complex task in where the user was tasked with having to perform all the three basic tasks at the same time to collide with the elevated two different boxes through different movements.

For Step 2, there were two main stages created as follows (Figure 5.2-down):

- *Training Level* - This was the practice level and was not part of the tests. In this level, there was a blue box that the user could use as reference as they tried out the controls.
- *Target Selection (Grab)* - This was the test stage where the 3D cursor movement was performed. The camera was in a fixed position and the user was tasked with having to move the small spherical 3D cursor and collide it with the target boxes. The red target boxes would appear one at the time, each one appearing after the previous one was selected. They would appear within the view/interaction space in front of the camera and were programmed in a fixed sequence and with a fixed position so that every user took the exact same test.

20 participants (10 male, 10 female), in the age range of 21 to 44 years old (with average of 29), all university students and right-handed (except one), took part in the study. A stopwatch was used to track how much time it takes to complete each task. Ethics approval was secured for participant surveys. The participants first read the experiment instructions and were given introductions to the tasks they were to complete during the trial. Appendix A gives more details about this experiment.

5.2.1. Results

The average values of *weighted satisfaction criteria* ($\tilde{X}$), as explained in Section 3.2, show the participants' preferences, for the Box game, in the order (from high to low) of pleasantness (6.8), non-fatiguing (6.7), easiness (6.4), movability (5.9), and naturalness (5.1). Figure 5.4 illustrates the orders.

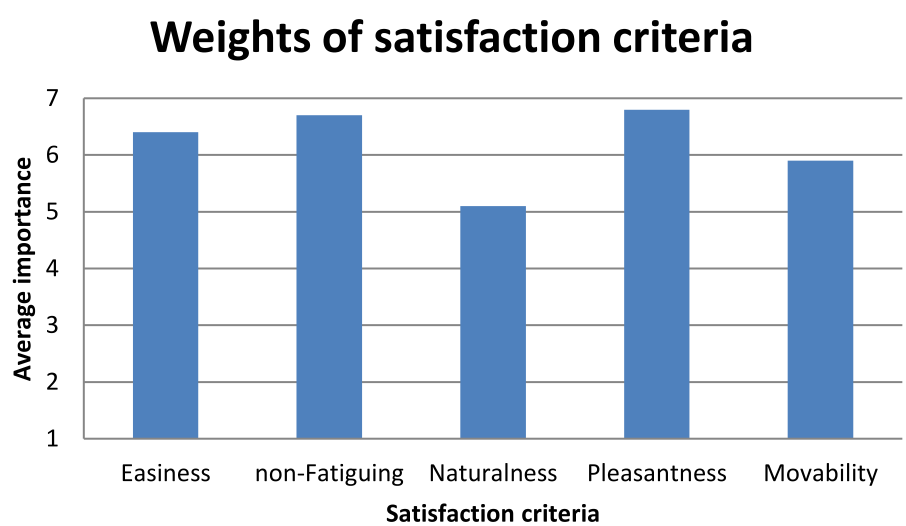


Figure 5.4. Weighted satisfaction criteria for the box game.

Applying the experiment data into the formulas explained in Chapter 3, Table A.5 enlists the measurements of all required elements to acquire the performance factors for the tasks of combination (Step 1) and grab (Step 2). Also, following the same method explained in Chapter 3 and the details of experiment, Table 5.1 presents the IoC for the Box application.

Table 5.1. Index of Complexity (IoC) for all actions in the Box application.

		Feedback of task	Touch-less modality	Touch-based modality
Step 1	***Action***	**Parallel Find (Combination)**	**Full body (3D)**	**Keyboard**
	Basic-quantities	Camera_velocity (3D) + Camera_angularVelocity (2D) + Target_collision (3D) + Target_action (1D)	Arm_move (2D) + Waist_move (2D) + Legs_jump (1D)	KB_button (5D)
	IoC	$((3 \times L) + T^{-1}) + ((2 \times r) + T^{-1}) + (3 \times L) + (A) = \mathbf{11}$	$(2 \times L) + (2 \times L) + (LT^{-1}) = \mathbf{6}$	$(5 \times L) = \mathbf{5}$
Step 2	***Action***	**Grab/Hit**	**Arm (3D)**	**Mouse & Keyboard**
	Basic-quantities	Pointer_translation (3D) + Target_collision (3D) + Target_action (1D)	Arm_move (3D)	KB_button (2D) + Mouse_move (2D)
	IoC	$(3 \times L) + (3 \times L) + (A) = \mathbf{7}$	$(3 \times L) = \mathbf{3}$	$(2 \times A) + (2 \times L) = \mathbf{4}$

Inserting the information acquired from Tables A.5 and 5.1 into the equations proposed in Chapter 3, effectiveness (E_s), efficiency (E_y), simplicity (RoS), and similarity (IoS) are calculated and listed in Table 5.2. This table also presents the measurements for the overall acceptability (OA), easiness ($\hat{E}$), non-fatiguing ($\hat{F}$), naturalness ($\hat{N}$), pleasantness ($\hat{P}$), movability ($\hat{M}$), and adjusted satisfaction (S_a) as explained in Chapter 3.

Table 5.2. Measurements of usability elements for distance-based interactions in the Box game.

			Usability Factors										
			Independent variables					Dependent variables					
			OA	E_s	E_y	***RoS***	***IoS***	S_a	$\hat{E}$	$\hat{F}$	$\hat{N}$	$\hat{P}$	$\hat{M}$
Context in use	***1***	***[Distance | Combination | Gesture | Desktop]***	4.5	0.4	0.7	1.8	-5	4.5	4.6	4.5	5.3	4.7	3.5
	2	***[Distance | Grab | Gesture | Desktop]***	5.0	0.4	0.3	2.3	-4	5.1	4.9	5.2	5.1	5.0	5.3
	3	***[Distance | Grab | Mouse/KB | Large-screen]***	5.1	0.5	0.5	1.7	-3	5.1	5.1	5.2	5.7	5.2	4.3
	4	***[Distance | Combination | Gesture | Large-screen]***	5.1	0.4	0.8	1.8	-5	5.2	5.0	5.1	6.3	5.3	4.3
	5	***[Distance | Grab | Gesture | Large-screen]***	5.7	0.5	0.4	2.3	-4	5.6	5.5	5.7	5.2	5.4	6.2
	6	***[Distance | Grab | Mouse/KB | Desktop]***	5.7	0.6	0.5	1.7	-3	5.2	5.2	5.8	6.2	5.7	3.3
	7	***[Distance | Combination | KB | Desktop]***	6.2	0.6	1.3	2.2	-6	6.3	6.7	6.7	5.3	5.9	6.9
	8	***[Distance | Combination | KB | Large-screen]***	6.2	0.6	1.2	2.2	-6	6.2	6.6	6.4	5.5	5.7	6.8

5.2.2. Analyses

In this section, the data provided in Table 5.2, for distance-based interactions, are analyzed in the Box game. The analyses follow the process explained in Section 4.2.

Easiness: Among the usability factors of interest for this hypothesis, *OA*, E_y, and *RoS* have meaningful correlations of 0.922, 0.794, and 0.514 respectively with $\hat{E}$ (Figure 5.5). In other words, $\hat{E}$ has a strong positive correlation with *OA* and E_y, while there is a moderate positive correlation with *RoS*. A multiple linear regression (Table A.6) shows a significant F-value (0.001), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, easiness is related to at least one of the usability factors of overall acceptability, efficiency, and simplicity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-values of 0.005 and 0.01 on *OA* and E_y respectively, which means they are significantly related to easiness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 1 and 4 in Table 5.2 shows that a gestural interaction on a large-screen display causes less error (due to better visibility) and consequently makes it easier than the same interaction on a desktop.

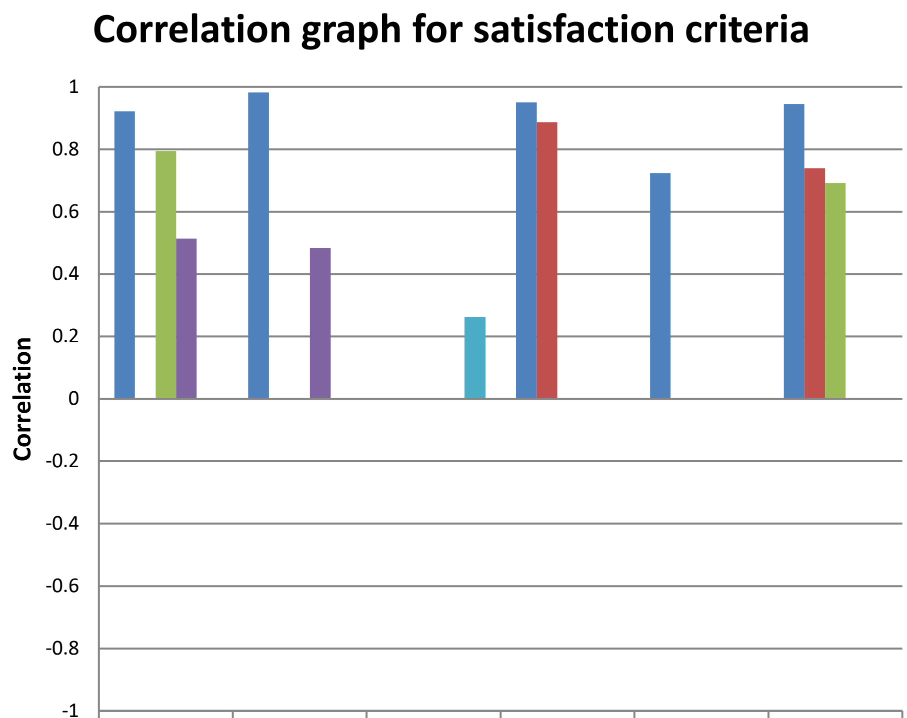
Non-fatiguing: Among the usability factors of interest for this hypothesis, *OA* and *RoS* have meaningful correlations of 0.982 and 0.484 respectively with $\hat{F}$ (Figure 5.5). In other words, $\hat{F}$ has a strong positive correlation with *OA*, while there is a moderate positive correlation with *RoS*. A multiple linear regression (Table A.7) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, non-fatiguing is related to at least one of the usability factors of overall acceptability and simplicity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.000 only on *OA*, which means it is significantly related to non-fatiguing. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 4 and 5 in Table 5.2 shows that a task of grabbing using gestures causes higher overall acceptability (due to better user control and freedom) and consequently makes it less fatiguing than the task of combination using the same modality.

Naturalness: Among the usability factors of interest for this hypothesis, only *IoS* has a weak positive correlation of 0.262 with $\widehat{N}$ (Figure 5.5). Lacking of a meaningful correlation on *OA* shows that the users have low preference on naturalness to be considered in the design of this application. A multiple linear regression (Table A.8) shows no significant F-value, which means the null hypothesis cannot be rejected and the alternative hypothesis is not accepted. Therefore, non-fatiguing is not related to the usability factors of overall acceptability and similarity. This is due to the non-meaningful and weak correlations of naturalness with overall acceptability and similarity respectively.

Pleasantness: Among the usability factors of interest for this hypothesis, *OA* and E_s have strong positive correlations of 0.950 and 0.887 respectively with $\widehat{P}$ (Figure 5.5). A multiple linear regression (Table A.9) shows a significant F-value (0.002), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, pleasantness is related to at least one of the usability factors of overall acceptability and effectiveness. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.043 only on *OA*, which means it is significantly related to pleasantness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 3 and 5 in Table 5.2 shows that a task of grabbing using gestures causes higher overall acceptability (due to better match between system and real world) and consequently makes it more pleasant than the same task using mouse and keyboard.

Movability: Among the usability factors of interest for this hypothesis, *OA* has a moderate positive correlation of 0.724 with $\widehat{M}$ (Figure 5.5). A multiple linear regression (Table A.10) shows a significant F-value (0.042), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, movability is related to the usability factor of overall acceptability. The positive correlation shows the direction of this relationship. Further analysis of T-test shows the significant P-value of 0.042 on *OA*, which means it is significantly related to movability. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 4 and 5 in Table 5.2 shows that a task of grabbing using gestures causes higher overall acceptability (due to better user control and freedom) and consequently makes it more flexible than the task of combination using the same modality.

Adjusted satisfaction: Among the usability factors of interest for this hypothesis, *OA*, E_s, and E_y have meaningful correlations of 0.945, 0.739, and 0.692 with S_a (Figure 5.5). In other words, S_a has a strong positive correlation with *OA*, while there are moderate positive correlations with E_s and E_y. A multiple linear regression (Table A.11) shows a significant F-value (0.001), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, adjusted satisfaction is related to at least one of the usability factors of overall acceptability, effectiveness, and efficiency. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.004 only on *OA*, which means it is significantly related to adjusted satisfaction. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 1 and 7 in Table 5.2 shows that an interaction using mouse/keyboard on a desktop display causes higher overall acceptability (due to better visibility) and consequently makes it more satisfactory than the doing it using gestures.



	Easiness	non-Fatiguing	Naturalness	Pleasantness	Movability	Adjusted Satisfaction
OA	0.922	0.982	N.M.C.	0.95	0.724	0.945
Es	N.M.C.	N.M.C.	N.M.C.	0.887	N.M.C.	0.739
Ey	0.794	N.M.C.	N.M.C.	N.M.C.	N.M.C.	0.692
RoS	0.514	0.484	N.M.C.	N.M.C.	N.M.C.	N.M.C.
IoS	N.M.C.	N.M.C.	0.262	N.M.C.	N.M.C.	N.M.C.

Figure 5.5. Correlations of usability factors of hypotheses with satisfaction criteria in the distance-based Box game (N.M.C. stands for "no meaningful correlation", with an absolute value of less than 0.25).

5.3. Discussion

Gamers' skills widely ranged from novice to expert. As a result, it would be valid to conclude that the findings of this study are applicable for practical purposes and not just tech-savvy users. In addition, the wide age span of the participants presents that the findings are general and age-independent.

In general, the full-body gestures have lower performance and overall popularity than the touch-based modalities. In addition, the combination of gesture-based input with large-screen is more acceptable for users than the same input with desktop-screen. This was intuitively expected due to a positive impact of movability where users have more space to interact.

The study shows that the correlation values of overall acceptability for satisfaction criteria are in a similar flow as of the average values of *weighted satisfaction criteria* (as demonstrated in Figure 5.4). This correlation is especially important for the critical values with significant differences. For example, naturalness has no meaningful positive correlation with overall acceptability, where it is also listed, by the users in their preferences for this application, as the least important satisfaction criterion. Therefore, the user's judgment is a reliable source for the UI design. This consequence is very useful for a user-centric design, which shows that the UI designers should trust the user's perception even for configuring the relationships between the observational controlling factors and the satisfaction criteria. The tuning process follows the loop illustrated in Figure 4.1. The final analysis of hypotheses is summarized in Table 5.3.

Table 5.3. Hypothesis evaluation in the distance-based Box application.

	Box (distance-based)	
	Significant positive correlation	**Hypothesis verification**
$\hat{\mathbf{E}}$	*OA*, E_y	✓
$\hat{\mathbf{F}}$	*OA*	✓
$\hat{\mathbf{N}}$	—	✗
$\hat{\mathbf{P}}$	*OA*	✓
$\hat{\mathbf{M}}$	*OA*	✓
$\mathbf{S_a}$	*OA*	✓

Further analysis, separately for gesture-based and touch-based input modalities, on the correlations of satisfaction criteria with the controlling factors, shows the following results: In gesture-based, effectiveness has more effect on easiness, non-fatiguing, pleasantness, movability, and adjusted satisfaction, while efficiency takes the main role on the mentioned satisfaction criteria in touch-based. This difference can be due to the distance-based design of this application, where more stable and faster performance reach out of gesture- and touch-based controls respectively. In other words, this is according to the nature of the two different input modalities, where gestures outperform the distance-based interactions more precisely, and desktop controls are faster due to the familiarity of usage. Comparing naturalness in the two types of modality shows that similarity is the main controlling factor only for the touch-based input. Lack of dependency on similarity for gesture-based can be due to the higher complexity level of cognition in this type of input, where other factors such as familiarity of usage and matching with real world may also be involved.

Eventually, through written feedback, most participants proposed “just or mostly touch-based” as their preferred combination of using gestures and/or touch-based modalities to interact on the game application in Step 1. They chose “equal use of gestures and touch-based” in Step 2, which is in positive correlation with the findings of this study. The activity in Step 1 covered a wider range of tasks and was not focused on those that could potentially be suitable for gestures and body motion. Compared to this, Step 2 included an activity that could benefit from free body movement as input. This can potentially explain the above results which show a change toward preferring gesture-based input from Step 1 to Step 2. It can justify the approach of starting with various tasks and then narrowing down and refining, in order to find those that are suitable for a particular modality.

Chapter 6: Case Study of the Slingshot Game

In this chapter, the quality of interactions are studied, while users perform a task of Grab & Hit in a 3D Slingshot game, using three input methods of static posture, dynamic gesture, and haptic control. Microsoft Kinect is used as the 3D sensor for gesture detection. The objective is to verify our hypothesis proposed in Chapter 4. For this purpose, some experiments are carried out for further analyses on different evaluation criteria in the fields of performance, satisfaction, complexity, and overall acceptability.

6.1. User Interface General Design

The proposed system has been designed under the following settings:

Operating System: 32-bit Windows 7 SP 1; Camera: Microsoft Kinect depth camera located at about two meters from the user for the best results (based on Kinect's technical characteristics); Development tools: Microsoft XNA and Kinect 360; Displays: a 1080p projector with a screen size of 92 inches located at about seven meters from the user was used for the large screen. Proper algorithms were designed to detect the proposed hand/finger gestures using the Kinect sensor and some existing API's (OpenNI [108], NiTE [108], and OpenCV [109]). A Falcon device was used for haptic controls (Figure 6.1).

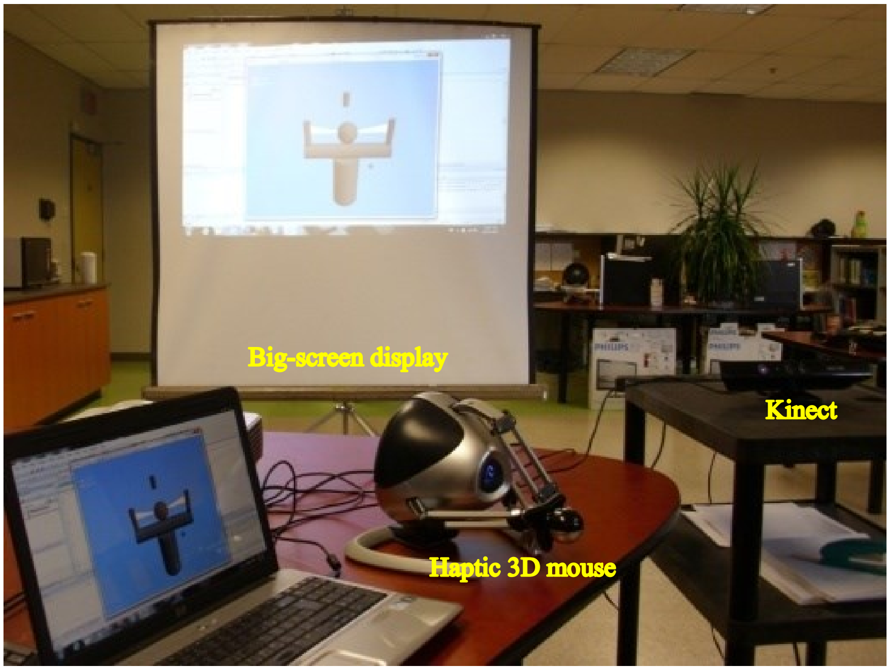


Figure 6.1. The system's layout for the Slingshot game.

For the user interface a Slingshot game has been designed, where the design is kept as simple and minimalistic as possible, with neutral colors to reduce user error or bias (Figure 6.2). An option for changing the camera view is also included that users can use when they wish. The same virtual environment is controlled by the three modalities (static postures, dynamic gestures, and haptic controls) independently.

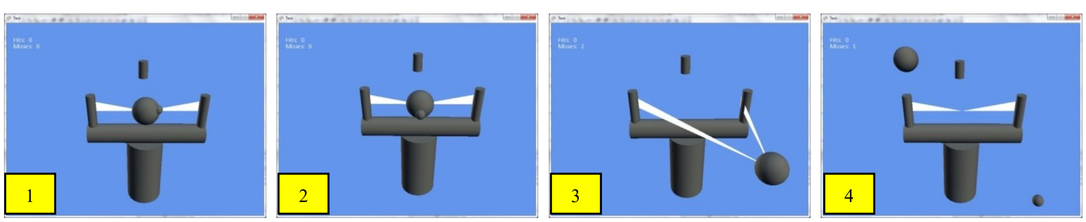


Figure 6.2. User interface for the Slingshot game.

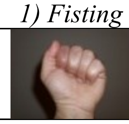
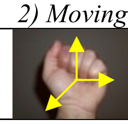
6.1.1. Vision-Based Module

Using the Kinect unit, as a commonly used vision-based input device, enables us to identify the depth of every single pixel in the frame. In addition, using OpenNI and NiTE has helped to secure the system with a higher stability and efficiency, and to develop a capable algorithm to recognize the hand and finger gestures.

6.1.1.1. Static Gestures

Considering the natural gestures to represent the tasks in interaction with the designed interface, the selected static postures definition is presented in Table 6.1.

Table 6.1. Design for hand postures in the Slingshot game.

Action	*1) Fisting*	*2) Moving*	*3) Palming*
Static fisting & palming			
Related task	*Grab*	*Pull*	*Release*

In order to recognize these gestures, as shown in Figures 6.3 and 6.4, it first needs to detect the fingertips through: i) depth thresholding, ii) contour extraction, and iii) assuming vertices of convex hull to be fingertips if their interior angles of hull corners are small enough (T_θ, obtained from the following equation, is the angle spanned by the predecessor edge and successor edge incident to vertex facing to the inside of contour polygon).

$$\boldsymbol{T_\theta} = \cos^{-1}\frac{(\boldsymbol{v_x} \times \boldsymbol{v'_x}) + (\boldsymbol{v_y} \times \boldsymbol{v'_y})}{\|\boldsymbol{v}\| \times \|\boldsymbol{v'}\|} \tag{6.1}$$

where $v_{x,y}$= $|$successor of corner index (x, y) – corner index $(x, y)|$, and $v'_{x,y}$= $|$predecessor of corner index (x, y) – corner index $(x, y)|$. Moreover, the depth map is automatically cropped to remove the wrist out of the frame in order to improve accuracy.

Following the above process of fingers detection, the system will recognize fist (no fingers) and palm (five fingers), and interact with the user interface through Algorithm 1.

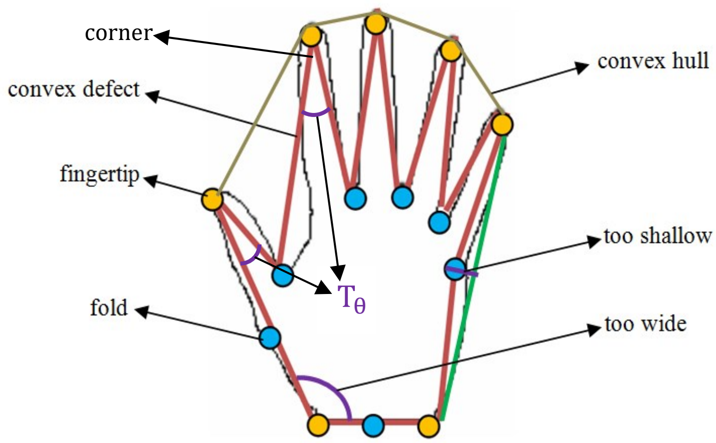


Figure 6.3. T_θ in the hands contour.

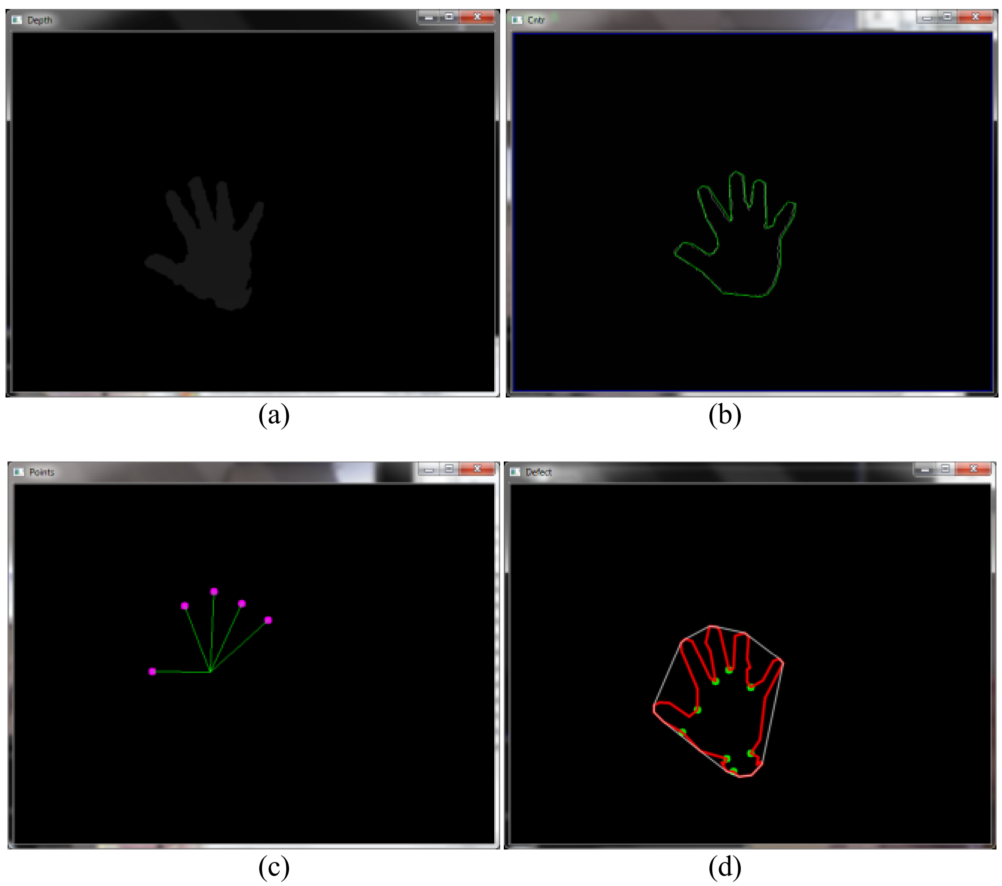


Figure 6.4. Images in the hand and finger detection processes: (a) depth thresholding, (b) contour extraction, (c) fingertips detection, and (d) depth points.

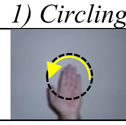
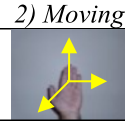
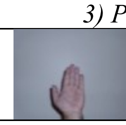
Algorithm 1. The algorithm for gesture recognition.

```
if (an initial Wave gesture happens) then
  the pointer appears (session starts)
  while the pointer follows the hand position (session updates)
    if (a grabbing gesture Fist/Circle is detected in the object area) then
      grab the object (save (x,y,z) based on distance deviation)
        if (a releasing gesture Palm/Push is detected in the object area) then
          release the object (transfer data)
```

6.1.1.2. Dynamic Gestures

During the controlling sessions of OpenNI and using the built-in gestures in NiTE, the system will recognize the dynamic gestures of Circle and Push (shown in Table 6.2), and interact with the user interface through the similar algorithm designed for finger gesture recognition (replacing Fist and Palm gestures by Circle and Push in Algorithm 1).

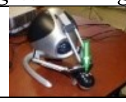
Table 6.2. Design for hand gestures in the Slingshot game.

Action	*1) Circling*	*2) Moving*	*3) Pushing*	
Dynamic circling & pushing				
Related task	*Grab*	*Pull*	*Release*	

6.1.2. Haptic Module

The haptic events controller was designed traditionally, as shown in Table 6.3. There are three dimensions of movement in this device: right/left and up/down like a mouse, but also forwards/backwards unlike a mouse. As the user moves the grip in the three dimensions, the Falcon's software keeps track of where the grip is moved and creates forces that a user can feel. For more control, there are four buttons on top of the spherical grip.

Table 6.3. Design for haptic controller.

Action	*1) Pressing & Holding the button*	*2) Moving the grip*	*3) Releasing the button*
Haptic controls			
Related task	*Grab*	*Pull*	*Release*

6.2. User Experiments and Analysis

The experimental evaluation starts with a training session of about 15 minutes (five minutes on each device) until the participants feel comfortable to start the test session. The participants try the following four primitive tasks to get used to the application in order to run the test session precisely:

a) Navigate: *Move the pointer* (tool) around
b) Select: *Grab the ball*
c) Drag: *Pull the ball* (elastic band) and point towards the pin
d) Deselect: *Release the ball* towards the pin

Finally, the time between any hit occurrences for the first 10 successful shots is recorded since the beginning of training session, in order to study the learning curves.

The second part of the evaluation is a test session, during which the user tries five shots using the haptic 3D mouse, static postures, and dynamic gestures to complete a later satisfaction questionnaire. An experiment observer keeps a record of the total time (during the 5 shots) and the scores (number of hits out of five shots) from the start to the end point. Appendix B gives more details about this experiment.

This study has been conducted using 20 participants (11 males and 9 females), all students from Ottawa region's grade school and universities. They ranged in the age of 15 to 45 years old (with average of 27), and 17 participants were right-handed. Ethics approval was secured for participant surveys.

6.2.1. Results

The average values of *weighted satisfaction criteria* ($\tilde{X}$), as explained in Section 3.2, show the participants' preferences, for the Slingshot game, in the order (from high to low) of pleasantness (6.7), non-fatiguing (6.5), easiness (6.5), movability (6.1), and naturalness (5.3). Figure 6.5 illustrates the orders.

Figure 6.5. Weighted satisfaction criteria for the Slingshot game.

Applying the experiment data into the formulas explained in Chapter 3, Table B.4 enlists the measurements of all required elements to acquire the performance factors for the tasks of Grab & Hit. Also, following the same method explained in Chapter 3 and the details of experiment, Table 6.4 presents the IoC for the Slingshot application.

Table 6.4. Index of Complexity (IoC) for all actions in the Slingshot application.

	Feedback of task	**Touch-less modality**		**Touch-based modality**
Action	**Grab & Hit (Drag & Drop)**	**Static**	**Dynamic**	**Haptic 3D mouse**
Basic-quantities	Pointer_translation (3D) + Object_collision (3D) + Object_action (1D) + Object_translation (3D) + Object_action (1D) + Object_velocity (1D) + Target_collision (3D) + Target_action (1D)	RH_palm (1D) + RH_move (3D) + RH_fist (1D)) + RH_move (3D) + RH_palm (1D)	RH_palm (1D) + RH_move (3D) + RH_circleVelocity (2D) + RH_move (3D) + RH_push (1D) with medium velocity (I = 1)	Haptic_move (3D) + Haptic_buttonHold (1D) + Haptic_force (3D) + Haptic_buttonRelease (1D)
IoC	$(3 \times L) + (3 \times L) + (A) + (3 \times L) + (A) + (L + T^{-1}) + (3 \times L) + (A) = \mathbf{17}$	$(A) + (3 \times L) + (A) + (3 \times L) + (A) = \mathbf{9}$	$(A) + (3 \times L) + ((2 \times r) + T^{-1}) + (3 \times L) + (I + L + T^{-1}) = \mathbf{13}$	$(3 \times L) + (A) + (M + (3 \times L) + T^{-2}) + (A) = \mathbf{11}$

Inserting the information acquired from Tables B.4 and 6.4 into the equations proposed in Chapter 3, effectiveness (E_s), efficiency (E_y), simplicity (*RoS*), and similarity (*IoS*) are calculated and listed in Table 6.5. This table also presents the measurements for the overall acceptability (*OA*), easiness ($\hat{E}$), non-fatiguing ($\hat{F}$), naturalness ($\hat{N}$), pleasantness ($\hat{P}$), movability ($\hat{M}$), and adjusted satisfaction (S_a) as explained in Chapter 3.

Table 6.5. Measurements of usability elements for mixed interactions in the Slingshot game.

			Usability Factors										
			Independent variables					Dependent variables					
			OA	E_s	E_y	***RoS***	***IoS***	S_a	$\hat{E}$	$\hat{F}$	$\hat{N}$	$\hat{P}$	$\hat{M}$
Context in use	*1*	***[Mixed | Grab & Hit | Dynamic | Kinect]***	5.4	0.8	4.4	1.3	-4	5.3	5.0	4.7	5.9	6.1	4.8
	2	***[Mixed | Grab & Hit | Haptic | Falcon]***	5.5	0.8	4.8	1.6	-6	6.1	5.9	6.0	6.1	6.0	6.5
	3	***[Mixed | Grab & Hit | Static | Kinect]***	6.3	1.0	5.9	1.9	-8	6.5	6.7	6.7	5.6	6.7	6.8

6.2.2. Analyses

In this section, the data provided in Table 6.5, for mixed interactions, are analyzed in the Slingshot game. The analyses follow the process explained in Section 4.2. However, no more regression analysis has been done, since the number of samples is low. Therefore, the hypotheses are verified by comparing the results individually, instead of statistically.

Easiness: Among the usability factors of interest for this hypothesis, *OA*, E_y, and *RoS* have strong positive correlations of 0.897, 0.956, and 0.999 respectively with $\hat{E}$ (Figure 6.6). The comparisons on all experiment results confirm the alternative hypothesis, meaning easiness is related to at least one of the usability factors of overall acceptability, efficiency, and simplicity. The positive correlations show the directions of these relationships.

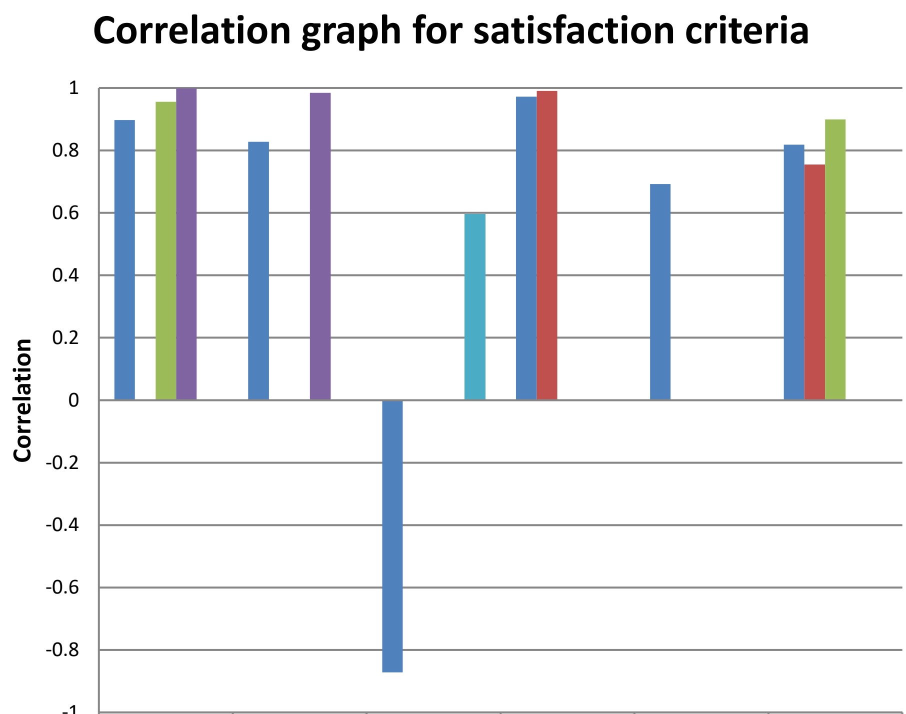
Non-fatiguing: Among the usability factors of interest for this hypothesis, *OA* and *RoS* have strong positive correlations of 0.828 and 0.985 respectively with $\hat{F}$ (Figure 6.6). The comparisons on all experiment results confirm the alternative hypothesis, meaning non-fatiguing is related to at least one of the usability factors of overall acceptability and simplicity. The positive correlations show the directions of these relationships.

Naturalness: Among the usability factors of interest for this hypothesis, *OA* and *IoS* have meaningful correlations of -0.872 and 0.596 respectively with $\widehat{N}$ (Figure 6.6). In other words, $\widehat{N}$ has a strong negative correlation with *OA*, while there is a moderate positive correlation with *IoS*. The negative correlation on *OA* shows that the users have no preference on naturalness to be considered in the design of this application. Since the hypothesis verification is merely concerned with the positive attitude of users about testing factors, the negative correlation is interpreted as "no meaningful positive correlation". The comparisons on all experiment results reject the alternative hypothesis, meaning naturalness is unrelated to the usability factors of overall acceptability and similarity. This is due to a non-meaningful positive correlation of naturalness with overall acceptability; even though the part of similarity in the hypothesis has a moderate positive correlation to naturalness.

Pleasantness: Among the usability factors of interest for this hypothesis, *OA* and E_s have strong positive correlations of 0.972 and 0.991 respectively with $\widehat{P}$ (Figure 6.6). The comparisons on all experiment results confirm the alternative hypothesis, meaning pleasantness is related to at least one of the usability factors of overall acceptability and effectiveness. The positive correlations show the directions of these relationships.

Movability: Among the usability factors of interest for this hypothesis, *OA* has a moderate positive correlation of 0.692 with $\widehat{M}$ (Figure 6.6). The comparisons on all experiment results confirm the alternative hypothesis, meaning movability is related to the usability factor of overall acceptability. The positive correlation shows the direction of this relationship.

Adjusted satisfaction: Among the usability factors of interest for this hypothesis, *OA*, E_s, and E_y have strong positive correlations of 0.818, 0.755, and 0.899 respectively with S_a (Figure 6.6). The comparisons on all experiment results confirm the alternative hypothesis, meaning adjusted satisfaction is related to at least one of the usability factors of overall acceptability, effectiveness, and efficiency. The positive correlations show the directions of these relationships.



	Easiness	non-Fatiguing	Naturalness	Pleasantness	Movability	Adjusted Satisfaction
OA	0.897	0.828	-0.872	0.972	0.692	0.818
Es	N.M.C.	N.M.C.	N.M.C.	0.991	N.M.C.	0.755
Ey	0.956	N.M.C.	N.M.C.	N.M.C.	N.M.C.	0.899
RoS	0.999	0.985	N.M.C.	N.M.C.	N.M.C.	N.M.C.
IoS	N.M.C.	N.M.C.	0.596	N.M.C.	N.M.C.	N.M.C.

Figure 6.6. Correlations of usability factors of hypotheses with satisfaction criteria in the mixed-based Slingshot game (N.M.C. stands for "no meaningful correlation", with an absolute value of less than 0.25).

6.3. Discussion

Similar to the experiment for the Box application, the gamers' skills and age span of the participants widely ranged in the Slingshot experiment. As a result, it would be valid to conclude that the findings of this study are general.

According to the provided analyses, it can be summarized that static postures are shown to be more superior (on all usability factors, except naturalness) than the two other input modalities. It is faster and less fatiguing to perform static postures due to fewer movements involved in the process compared to the two others. Moreover, the trade-off between the continuous attachment and force feedback in the haptic 3D mouse still shows a more natural interaction compared to the static postures, while the ergonomic design and force feedback increase fatigue and error rate. Since most of the human factors showed to be superior for static postures compared to the other modalities, it is valid to expect higher rates for the overall acceptability feedback as well.

On the other hand, a study of effort-based usability shows that decreasing complexity of modality increases the level of performance/satisfaction especially on static modality, which means a lower number of actions with lower IoC is needed to improve the quality of interaction in Slingshot application.

Similar to the findings from the Box application in the previous chapter, the correlation values of overall acceptability for satisfaction criteria are in a similar flow as of the average values of *weighted satisfaction criteria* (as demonstrated in Figure 6.5). For example, as for the critical values with significant differences, naturalness shows no meaningful positive correlation with overall acceptability, where it is also ranked, by the users, as the least important satisfaction factor. Therefore, we can reliably count on the user's judgment for configuring the relationships between the observational controlling factors and the satisfaction criteria. The next step is the tuning process as illustrated in the loop section of Figure 4.1. The final analysis of hypotheses is summarized in Table 6.6.

Table 6.6. Hypothesis evaluation in the mixed-based Slingshot application.

	Slingshot (mixed-based)	
	Significant positive correlation	**Hypothesis verification**
$\hat{\mathbf{E}}$	*N/A*	✓
$\hat{\mathbf{F}}$	*N/A*	✓
$\hat{\mathbf{N}}$	*N/A*	✗
$\hat{\mathbf{P}}$	*N/A*	✓
$\hat{\mathbf{M}}$	*N/A*	✓
$\mathbf{S_a}$	*N/A*	✓

Further study for learning curves shows that it is faster to learn the interactions using static postures, then dynamic gestures, and lastly haptic controls. Finally, the direct feedback from participants demonstrates that most of them suggest "equal use of haptic 3D mouse and gestures" and "mostly gestures" as their preferred combination of modalities.

Chapter 7: Design of the Race Game

Within the two previous chapters, first a full-body gestures system was compared to a traditional control system composed of keyboard and mouse to evaluate their relations with the tasks of different complexities in a 3D Box game. Then two arm/hand gestures systems (static and dynamic) were compared to each other and to a 3D haptic system to evaluate their relations with a limited level of discrete and analog tasks in a 3D Slingshot game. The findings of the aforementioned studies necessitate a further research in which hand gestures, as experimentally proved to be easier and more natural than the full-body gestures in a typical HCI system, need to be elaborately designed with a high level of discrete and analog tasks in a user interface simulated as practical as possible. For this purpose, a hand/finger recognition system has been designed in an application (Race game) to perform the third experiment of this research, in which its relations with a broad range of discrete to analog tasks are studied. For the concern of hardware availability, a single web camera has been used in the vision-based system design, along with a combination of mouse and keyboard for desktop controls. Due to more depth in the technical contribution of the UI design and gesture recognition process, this chapter solely explains the details of design. In the next chapter, some experiments are carried out to analyze the usability.

7.1. User Interface General Design

The proposed system has been designed under the following settings: Operating System: 32-bit Windows 7 SP 1; Camera: Logitech B525 HD webcam located at about one meter from the user for the best results (based on camera's technical characteristics); Development tools: Allegro 4, C++, and OpenCV 2.4.6 in Microsoft Visual Studio 10; Displays: two 24 inches LED monitors located at a meter from the user (Figure 7.1).

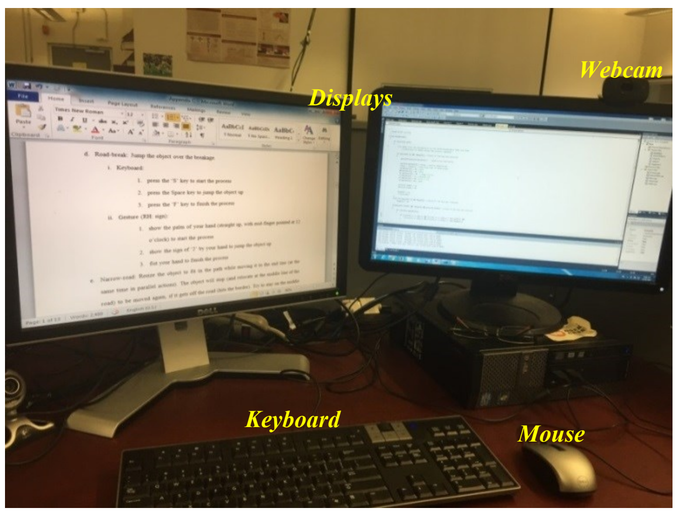


Figure 7.1. The system's layout for the Race game.

For the purpose of generality, a race game (Figure 7.2) is designed as simple and minimalistic as possible with neutral colors to reduce user error or bias, and with a vast range of tasks under some physics engine to cover the common actions in a game design. The GUI has various terrains (activities) including: activation (select the object), city-road (move under a limited speed), road-break (jump/switch to the other side), narrow-road (resize to fit and move at the same time, in parallel actions), bridge (resize to fit then move, in serial actions), cruise-control (rotate on the road turns with a fixed speed), free-way (shoot and control the direction with a fixed acceleration), deactivation (deselect the object). During the process of activation, the pointer (yellow circle), which follows the right-hand's trajectory, should collide the object (red box) in order to select it. If the user speeds up on city-road more than 1.2 times of the allowed speed or touches the borders on city-road/ narrow-road/ bridge/ cruise-control, the object will stop and relocate at the middle of the road due to an error, and the user will need to pick it again. If the object touches the borders on free-way, then that point would show the effective distance of the shooting task. The time, speed, stability, effective distance, and number of errors in various actions are automatically conserved by the program, for further analyses.

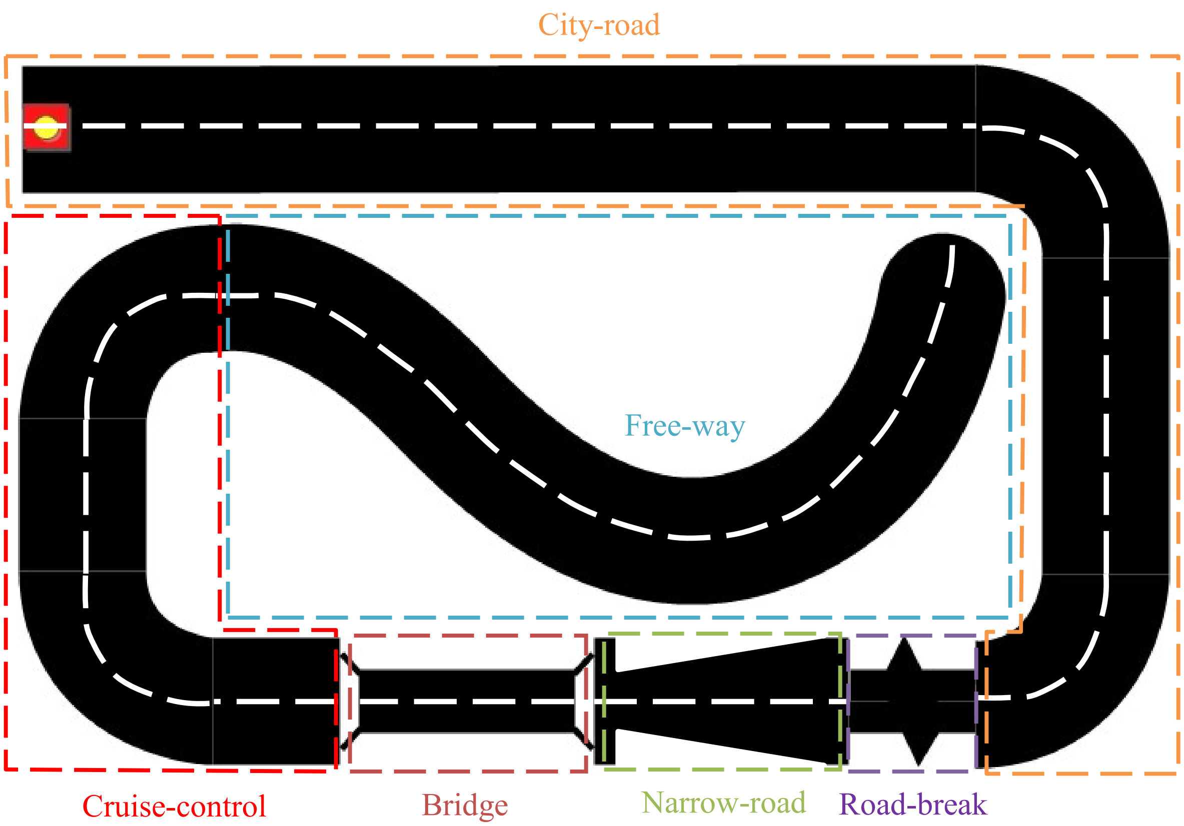
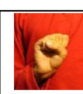


Figure 7.2. User interface and its sections in the Race game.

7.1.1. Vision-Based Module

For each stage, hand/finger gestures are defined corresponding to the actions to be performed. The variety of actions has been addressed by considering left/right hand gestures (Table 7.1). Moreover, by applying the proposed algorithm of Clutch-Gear-Rest the number of possible actions is increased. This algorithm is a controlling process using start-action-neutral setting, where palm gesture of hand represents start/clutch, various gestures of hand represent actions, and fist gesture of hand represents neutral/rest. Even though the proposed active controls could be simplified into less numbers, these complex combinations of active gestures have been preferred for the purpose of technical generality and expandability.

Table 7.1. Design for hand gestures.

Activity	Task	Action						
A) Set/Reset	*Connect pointer*	 Neutral	Move	 Start				
B) Activation	*Select*	 Start	Move	 Select				
C) City-road	*Limited speed*	 Start	Drag					
D) Road-break	*Jump*	 Start	Move	 Jump				
E) Narrow-road	*Resize + Move*	 Start	Move	   Resize & Drag				
F) Bridge	*Resize*	 Start	Move	   Resize			 Next	
	Move							Drag
G) Cruise-control	*Rotate*	 Start	Move	 Rotate upward		 Neutral	 Rotate rightward	
H) Free-way	*Shoot*	 Start	Move	 Shoot downward		 Shoot upward		
I) Deactivation	*Deselect*	 Start	Move	 Deselect				

Gesture recognition module: A graphical representation of the proposed algorithm for finger gesture recognition is shown in Figure 7.3. Figures 7.4 and 7.5 depict the extended diagram of the entire processes. Analysis of these stages follows.

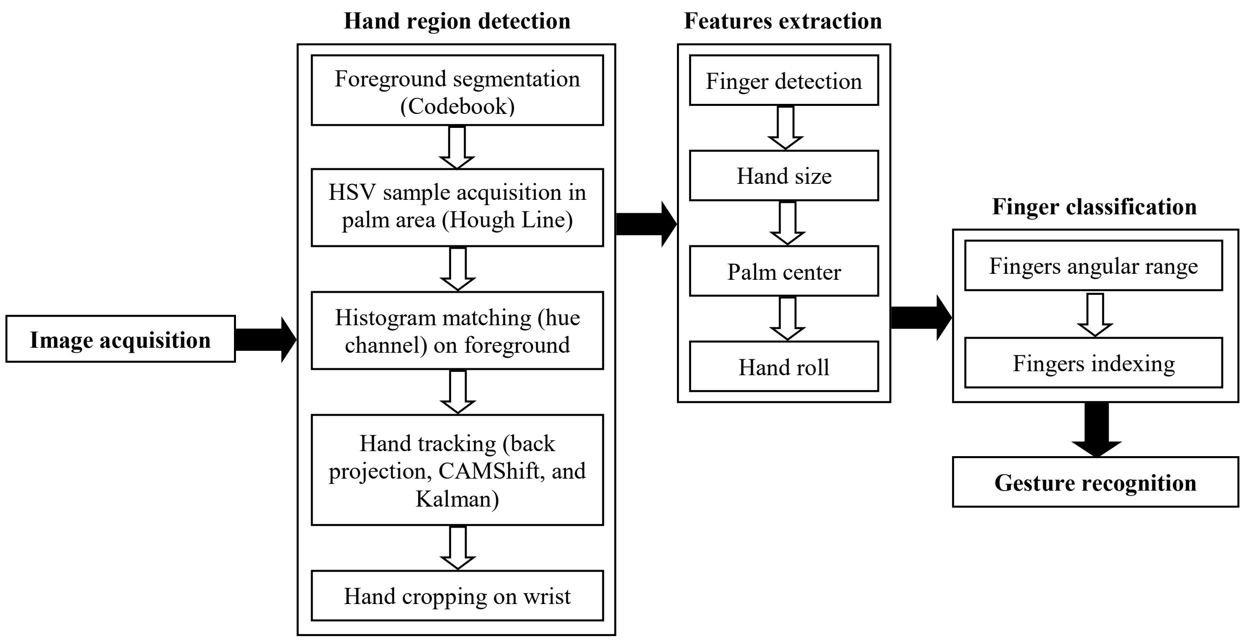


Figure 7.3. Scheme of the proposed algorithm for hand/fingers gesture recognition.

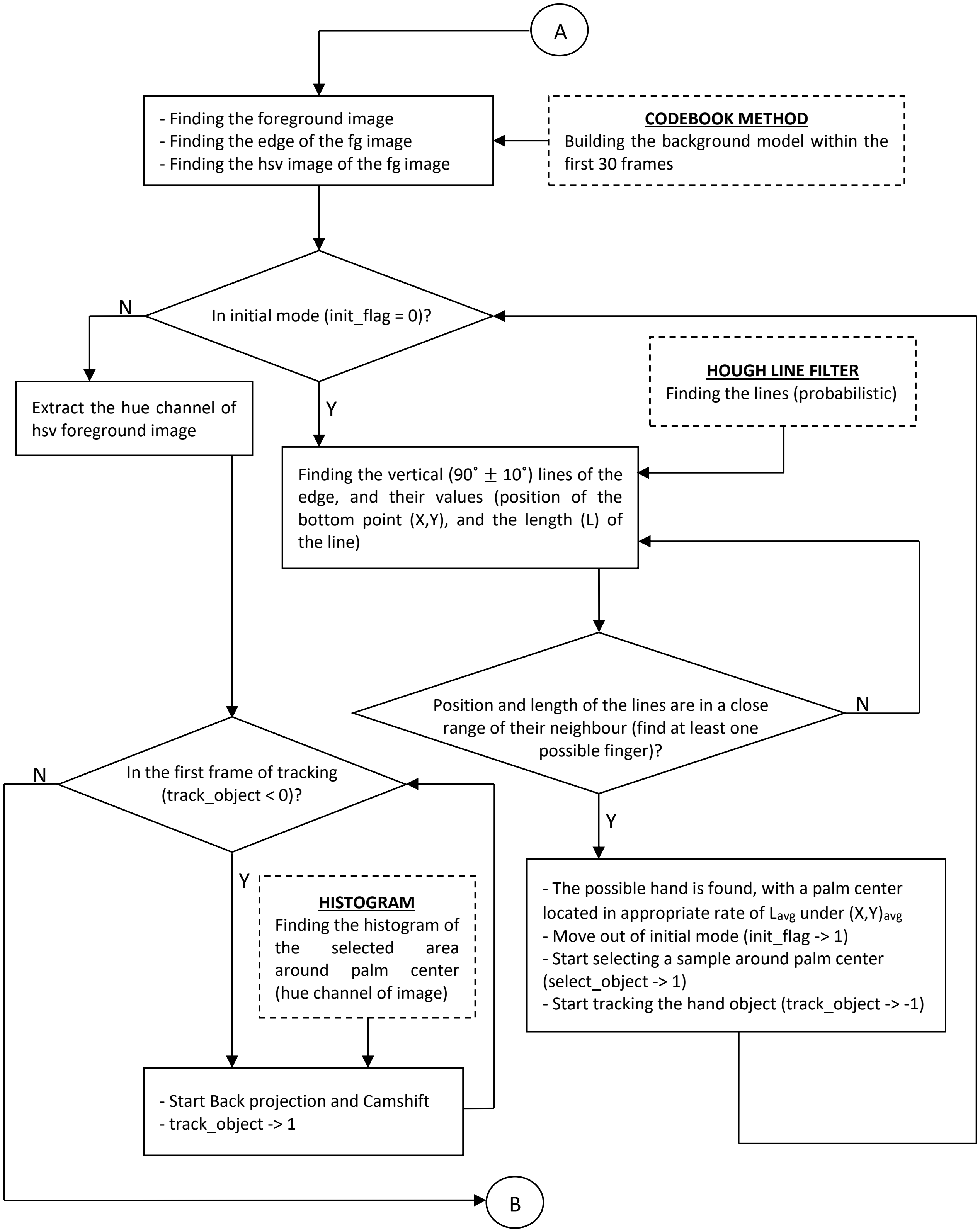


Figure 7.4. An extended diagram of the proposed algorithm for gesture recognition in the Race game (part 1).

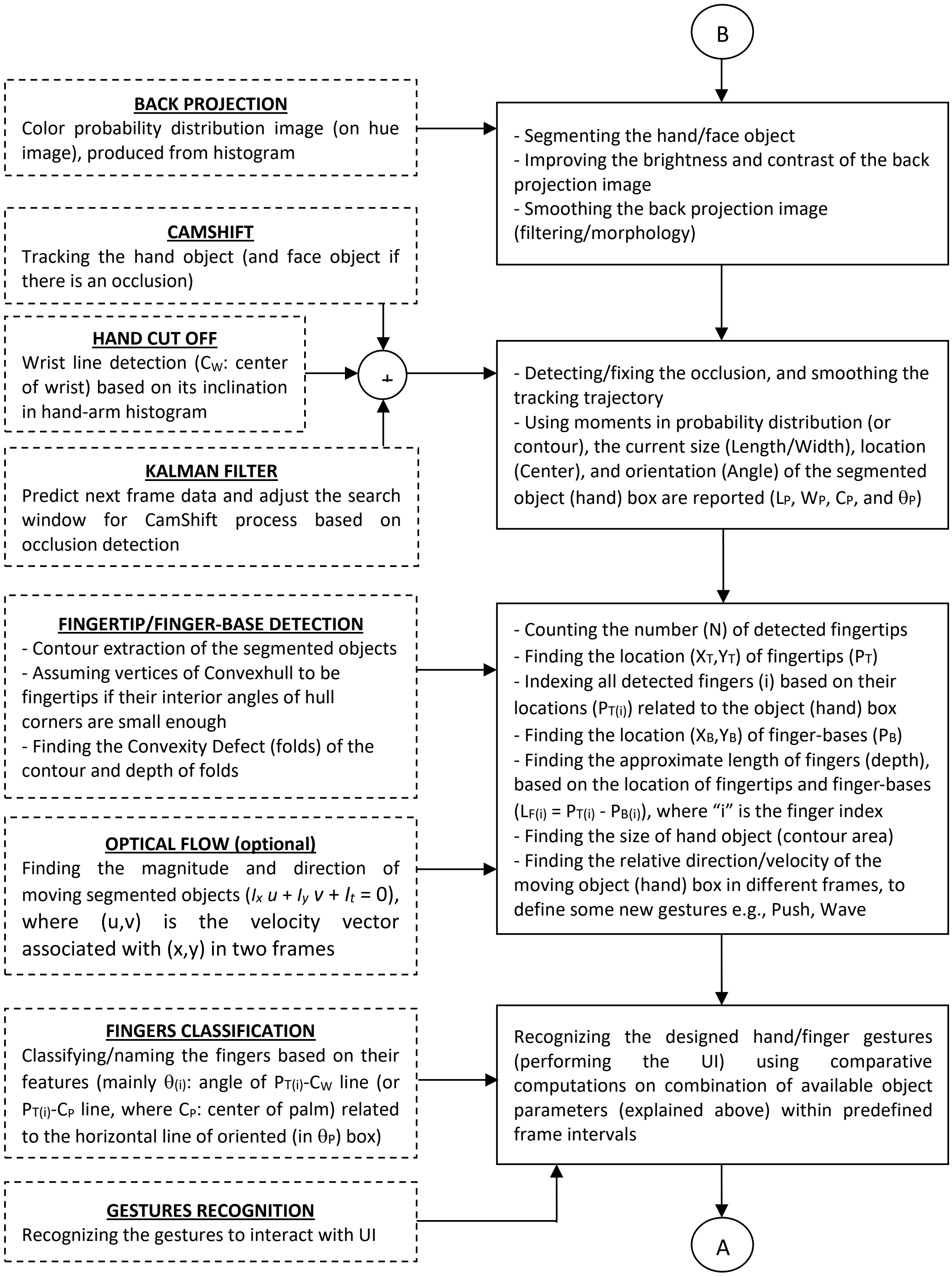


Figure 7.5. An extended diagram of the proposed algorithm for gesture recognition in the Race game (part 2).

Hand region detection: As shown in Figure 7.3 for the block of Hand Region Detection, the first step is to segment the moving foreground. This output then needs to be scanned in order to filter out only the skin area of the foreground image. The hand object through moving skin area is the subject of the next segmentation. A further segmentation is needed to crop the upper hand area from its wrist line. Having the hand image cropped from the previous step, it needs to be tracked continuously through a set of images in a real-time video. The masks produced in each level are the input for its next level, where the final mask is the desired hand object to be used for the process of features extraction.

Foreground segmentation: The first step of hand region detection is foreground/background segmentation. Acquired frames from a low-cost web camera contain a lot of noise, which in effect causes the values of static points in the background to differ substantially between two successive frames. Considering the noise caused by the values of static points and the complexity of the algorithm for background segmentation, it is preferable to use the Codebook [110] over some other methods such as MOG (Mixture of Gaussian) [110,111]. The Codebook method is based on setting thresholds, which dictates whether given pixel belongs to the background model or not. These thresholds can also be interactively configured in order to get a properly segmented foreground. The background model is computed in the YUV color space [110,112] from 30 subsequent frames, which contain no hand (or no tracking object, in general). Incorrectly detected artifacts in the segmented foreground mask are removed by morphological operations [113].

HSV sampling of the palm: There have been different methods for object (hand) segmentation, among which skin filtering can be named. This approach either needs an initial intervention by the user to feed a color sample area of the object, or is restricted by a preset skin color to be matched with the hand in the scene. For the first case, the system suffers from a non-automatic process. For the second case there can be false positive results for objects with similar color characteristics to skin, while also may fail to match a true hand with different color than the one provided as a searching criterion. Therefore, as a contribution of this research, a method has been designed to automatically detect the hands in the scene, using Hough Line filter.

This method (as shown in a part of Figure 7.4) looks, in a usual pose of a raised hand, for the vertical lines (90˚ ± 15˚) of the multiple edges (detected with a vertical Sobel filtering), and keeps their values which are the bottom point position (x_i, y_i) and length (l_i) of the i^{th} line in a 3D vector with n members (number of vertical lines). Through the following constraint, it checks if the members of vector, which are in a close proximity of their neighbours ($\Delta x_{min} < | x_i - x_{i+1} | < \Delta x_{max}$ and $\Delta y_{min} < | y_i - y_{i+1} | < \Delta y_{max}$), have similar length ($| l_i - l_{i+1} | < \Delta l_{max}$ and $l_{min} < l_i < l_{max}$). These ranges have been experimentally set (based on the initial depth of hand from the camera which in the present case is between 40cm to 70cm, for the best accuracy) as follows: Δx_{min} = 15pxl, Δx_{max} = 50pxl, Δy_{min} = 15pxl, Δy_{max} = 30pxl, Δl_{max} = 50pxl, l_{min} = 50pxl, and l_{max} = 80pxl. If these conditions are met at least four times (parallel fingers with their edge lines), then it is plausible that a hand is found. At this point, the average length (l_{avg}) and the average position (x_{avg}, y_{avg}) of the vector members were calculated which passed the constraint. This approach finds the fingers' lines (if any exists) in a raised hand posed in a practical range of tolerance for its blob relative to the vertical line (about 60˚ of hand rotation), and then takes an HSV sample of the palm with a small window at an appropriate location of (x_{avg}, $y_{avg} + l_{avg}$) under the detected fingers.

Color filter and skin region detection: Any object in the scene is extracted based on a histogram matching of a skin color sample in the hand area (after detecting a possible hand). In other words, it is based on HSV color space in which the skin area of image would be filtered using the histogram of the chromaticity (hue channel) value in the search window. Moreover, the color filter segments the image according to the ranges of saturation and lightness values calculated from the obtained skin color sample. A setting of the saturation boundary values is necessary since the low values of saturation leads to inaccurate corresponding hue values [91]. Back projection shows how well the pixels of image fit their distribution in a histogram model. Therefore, the final step is histogram back projection which is a probability map obtained after a back projection operation of the color model (typically the histogram of the search color that is to be isolated).

Morphological operations: Applying morphological operators removes small regions. This facilitates to properly select the regions of a possible hand (including any object in the scene with a similar color histogram to the search window). Two basic artifacts of Dilation and Erosion are the spine for the morphological operations [3]. Dilation or erosion, respectively, adds or removes pixels on/from the boundaries of objects in an image. Maximum or minimum value of the pixels in the neighborhood is the output of dilation or erosion respectively. By using the minimum value for dilation and the maximum value for erosion, the issue of homogeneous border is avoided [109].

Hand tracking: In this part, the goal is to separate and track the hand object in a video sequence or in real time, even under short occlusions, using a combination of histogram back projection, CAMShift, and Kalman estimator algorithms. Mean-Shift is a tracking approach which detects patterns of a probability distribution (initially obtained by the histogram back projection) through a recursive procedure converging to the nearest stationary point. CAMShift (Continuously Adaptive Mean-Shift) algorithm cleverly exploits Mean-Shift by changing the size of initial search window until it arrives to a convergence. At this end, applying the Connected Component method, the hand object is isolated from other parts of the image and is tracked [109]. Figure C.2 depicts the CAMShift algorithm.

The fingertips, which will be obtained from the next steps, are not stable due to noise and occlusion. On one hand, the fingers may be occluded, which causes the position error. On the other hand, since fingers movement is continuous and predictable by having the history information of fingertip position, further process on the initial fingertips is needed to acquire a stable result. This problem can be solved by using an effective Kalman filter. Kalman filter algorithm consists of three important values: measured value, estimated value and correct value. The measured value is the initial fingertip position, while the speed of moving fingers predicts the estimated value. Combining the measured and estimated values result in the correct value for the accurate fingertips' positions from the previous few frames [109]. Figure C.3 shows the cycle in discrete Kalman filter.

Wrist end detection and hand cropping: According to the histogram generating functions in [114], the wrist end is detected from the binary silhouette matrix of the image. Section C.3.1 explains the details of finding the wrist based on inclination in hand-arm histogram where wrist end is located on the valley (since wrist is the narrowest part of a hand). After the initial setup for center of wrist (*Wrist-Center*) in this step, further optimization takes place to adjust an optimum location of the real *Wrist-Center*.

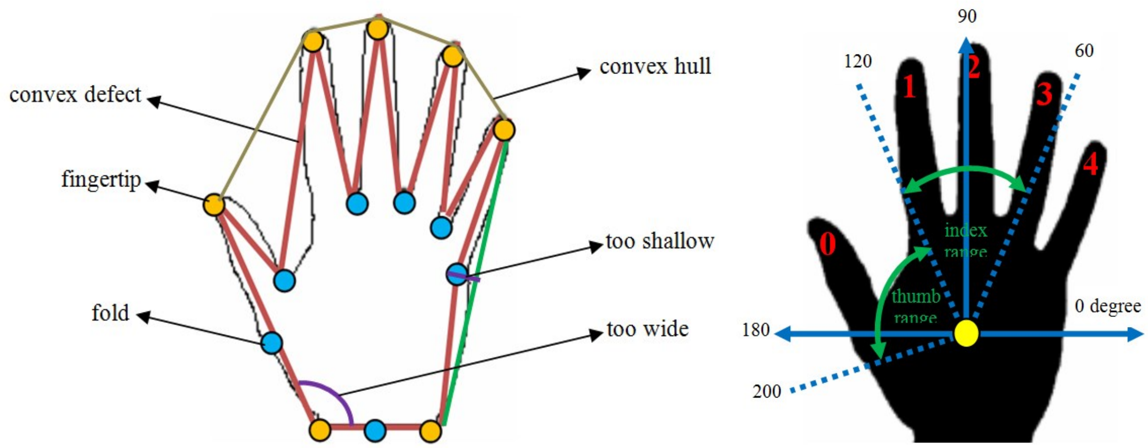
Palm center and hand roll (features extraction): Palm center and hand roll (rotation around the z axis) are obtained from the moments of the probability distribution image during the course of hand tracking algorithm, which is invariant to translation, scale, and orientation of the hand blob, due to the usage of the second order central moments and Hu set of invariant moment for the hand's blob (details are explained in Section C.3.2).

Fingers detection (features extraction): The finger detection module proceeds on the extracted hand contour to identify digits. It is based on convexity detection (as shown in Figure 7.6-a). The aim is to locate and count the convex and concave points of the hand shape. The key steps in this approach are: 1) approximating the hand contours with a polygon, 2) finding the convex hull on the polygon, 3) detection of convex and concave points, 4) filtering the convex and concave points, and 5) identifying digits [115]. In order to tune the output of approximation to a polygon, it considers an average of the successor in current convexity defect and the predecessor in next defect, where they meet on the convex hull. This technique avoids dual fingertips and also makes the motion of all found fingertips smoother over the video frames. The curvature of all boundary points, using Cosine-Law, finds five fingertips and four inter-finger points (folds), as shown in the following equation [13]:

$$a_i = (x_i - x_{i-1}, y_i - y_{i-1}) \, , b_i = (x_i - x_{i+1}, y_i - y_{i+1}) \, , \cos \theta_i = \frac{a_i . b_i}{|a_i| \, |b_i|} \tag{7.1}$$

In order to filter out the defects that are unlikely to be fingertips, it discards: 1) the fold points with shallow defect depths, 2) the tip coordinates with too large angle between their neighboring fold points, and 3) applying a distance condition (as explained in the following formula), in order to remove the outliers of falsely detected fingers on the edge of fist (Figure 7.6-a). The distance from fingertip to COG (*fingertip-to-COG*) must be bigger than 75% of the longest detected fingertip to COG (*fingertip-to-COG$_{max}$*), as shown in Figure 7.7 and the following equation:

$$(\text{fingertip} - \text{to} - \text{COG}) \; > \; 0.75 \times (\text{fingertip} - \text{to} - \text{COGmax}) \tag{7.2}$$



a) b)

Figure 7.6. a) Finding the fingertips/folds, filtering shallow depths of folds and wide angles of fingertips, and b) sorting and naming the fingertips based on angle ranges of fingers.

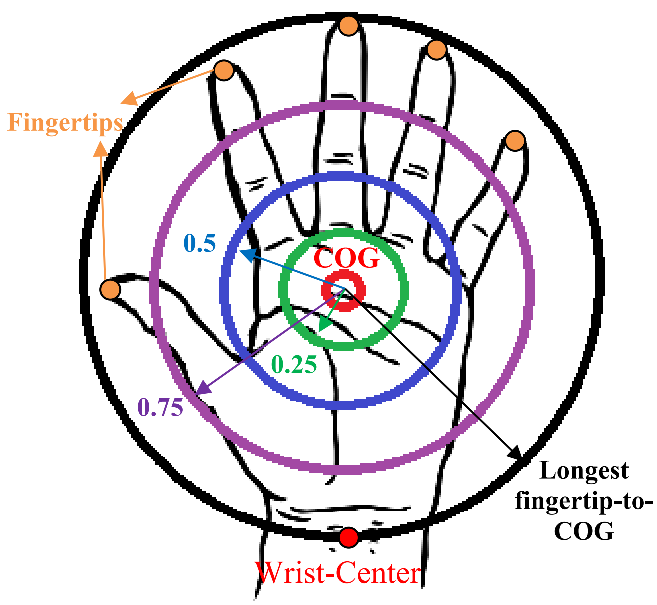


Figure 7.7. Distance condition proposed in finger detection algorithm.

At this end, further optimization applies to the center of wrist (*Wrist-Center*), extracted from the previous section, by adjusting its height to be located inside of a circle surrounding the hand blob with its center at COG and a radius of average length from all detected fingertip-to-COGs (*fingertip-to-COG$_{avg}$*). This constraint keeps the *Wrist-Center* in the region of the two biggest circles (Figure 7.7), and is formulated as following:

$$(\text{Wrist} - \text{Center})\text{height} < \text{COGheight} + (\text{fingertip} - \text{to} - \text{COGavg}) \quad (7.3)$$

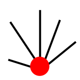
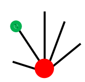
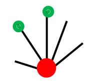
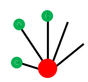
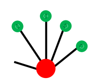
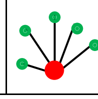
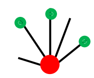
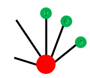
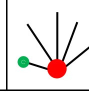
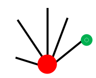
Fingers classification (sorting and indexing): Since the natural sequence of fingers necessitates the sorting of fingers in each frame and should be invariant to the hand roll, fingers/hand pose classification is challenging. Therefore, identifying each one of the raised fingers needs to be done by extracting the features from each finger's collected data.

In this method, the major feature is the angle between the lines of *fingertip-to-COG* and *Wrist-Center-to-COG* (orientation of hand box). While this feature proved its uniqueness and discrimination, it can also be used with other features, e.g., the perpendicular distance between the finger-bases and the vertical line of hand box [3].

Assuming that the hand is initially oriented straight up; in the maximum state, experimentally the index finger can turn between 60 and 120 degrees around the palm center of gravity, while the thumb can move between 120 and 200 degrees [115] (Figure 7.6-b). More or less a similar range of turn can be considered for middle, ring, and little fingers. However, in the finger identification algorithm, for the purpose of avoiding the interference error, it was necessary to classify them in a set of narrower ranges as: $0^\circ <=$ little $< 45^\circ$, $45^\circ <=$ ring $< 80^\circ$, $80^\circ <=$ middle $< 110^\circ$, $110^\circ <=$ index $< 170^\circ$, and $170^\circ <=$ thumb $< 240^\circ$. Furthermore, as experimentally tested and observed, in order to increase the resolution of fingers' angular moves into the safer zones, the location of the COG is shifted up with the amount of one third of the wrist width (the wrist width is naturally correlated to the palm radius).

Discrete controls (recognition process): Now that each finger has been indexed and classified, a user interface controlling algorithm can be carried out to recognize a combination of raised fingers. Table 7.2 depicts the gestures defined in American Sign Language (ASL), which are used in the discrete design of the Race game.

Table 7.2. ASL Gesture definitions used in the discrete design of Race application (the green circles indicate open fingers, while others without a dot are folded).

0/Fist (Rest)	**1** (Select)	**2** (Jump)	**3** (Resize/Shoot down)	**4** (Shoot up)	**5/Palm** (Start activity)	**7** (Start rotation)	**9** (Deselect)	**Open A** (Finish activity)	**Open I** (Finish resize)
									

Analog controls (recognition process): Recognizing the hand's rotation necessitates a trade-off between the speed and the precision of the recognition process. Most studies have focused on machine learning methods which employ hundreds of samples, which slow down the process.

To this end, an algorithm named Relative Displacement has been designed. Hand's rotations consist of three types of rotation around x (left to right), y (down to up), and z (near to far) axes. Through this algorithm, the changes in the height of the CAMShift bounding box –a box surrounding the hand object where the hand is located upward– are found for rotation around x. A similar process is carried out to find the changes in the width of the bounding box for rotation around y. To elaborate for the latter case, a 90˚ is assigned to the maximum half width of the box, where the camera sees the full-size of upright hand. This width is decreased to a portion of $\cos\beta \times$ *half of full length*, where β is the rotation around y axis. Since the full length of the bounding box (the length average in the first 10 initial frames) and the current width of it in every frame are known, through a proportional experimental equation we can find the rotation angle (β) around y, between 90˚ (assigned to the full length of the box) and zero (assigned to the minimum length of the box). Due to the thickness of hand's profile, the current width cannot be zero naturally, and consequently the smallest angle never can be zero without using a proportional equation. According to the capability of recognition for both hands in process, this algorithm supports a hand's rotation of 90˚ to -90˚ with flipping the hand from its right-hand shape to the left-hand-shape. In order to change the sign of rotation during the sequence of hand's movement, a threshold passing of zero to 5˚ is considered.

A similar approach has been applied for the rotation around x axis (α), while the range of 90° to -90° for the rotation can be extracted from flipping the hand to its respective degree of rotation (from upward to downward). However in practice, the rotation, around x, of zero to -90° is not reliable due to the occlusion problem (mixture of hand region with the lower part of the arm). The last type of rotation is around the z axis (γ) which can be easily extracted from the hand roll calculated previously. Eventually, a Kalman filter has been carried out on the extracted rotation angles, which is crucial in smoothing the angle changes.

Detecting the hand's rotation provides a great potential to remotely control analog applications such as a 3D control in virtual environments, robotic end-effectors, or home appliances like a fan.

Both hands (left/right) in process: In the proposed prototype, all the above processes can be performed using both hands. In order to gain this capability, the algorithm sorts the fingers based on their height (from lowest to highest), where the hand is located upward (with a minimum hand roll). In this configuration, the lowest height is associated to the thumb. Therefore, the right hand is detected when the thumb is on the left side of COG, while the left hand would see the thumb on the right side of COG (Figure 7.8).

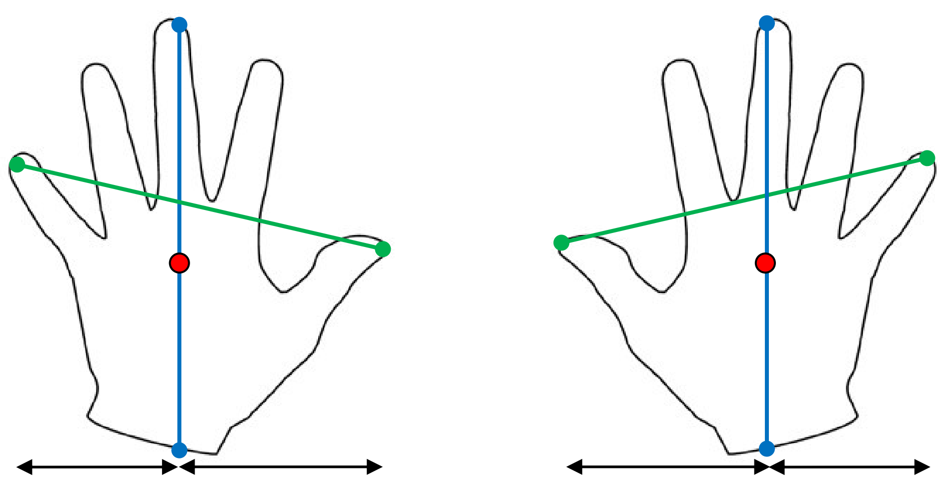


Figure 7.8. Both hands: left/right.

Results and discussion: Figure 7.9 demonstrates the applied design in the gesture module (from left to right and up to down) as follows: a) hand detection, b-d) extraction of hand rotations (30° around x, 49° around y, and -30° around z axes), and e-p) finger gestures recognition from 0 to 9, fist, and palm. The yellow box is the mask area surrounding the hand object. Each finger is identified by their number (from 0 for thumb to 4 for pinky). The green circles represent the last predictions by Kalman respectively on each finger, while the blue circle shows the center of the wrist. The up-right letter shows which hand (left/right) is seen, whereas the up-left numbers presents the number of fingertips and finger-based in the scene.

According to the results, and after background subtraction (in about 3 seconds), the hand region has been detected in less than 5 seconds through the Hough filtering, and the skin sample has been taken only from the palm area (as shown in Figure 7.9 by a purple circle, under the detected green fingers' lines). If there is a minimal hand movement, the system results in the correct counted numbers of fingertips/folds, identified names of fingers, and recognized gestures. However, when the angular movement is involved, there are some false detection mostly due to the local movements of fingers and some miscalculation of hand roll (classification error). The former is expected to be cured by applying a non-linear extended Kalman filter, while the latter can be overcome using a more robust hand roll detection. Wrist and COG have been detected correctly, whether the movement is involved or not. Also, the linear Kalman filter improves the gestures recognition during a short occlusion. Moreover, the experiment shows acceptable rotation recognition around *'y'* (conditional to the depth of the hand remaining unchanged) and *'z'* axes. A jittery and in some extent imprecise rotation recognition has been observed around the *'x'* axis which is due to the fact that a low range of changes in height (may cause by noise) can be converted to a great change in rotation. Furthermore, the experiment shows the best results in the depth range of 20cm to 1.2m from the single web camera. Finally, the same results have been observed for the left and right hands, considered separately.

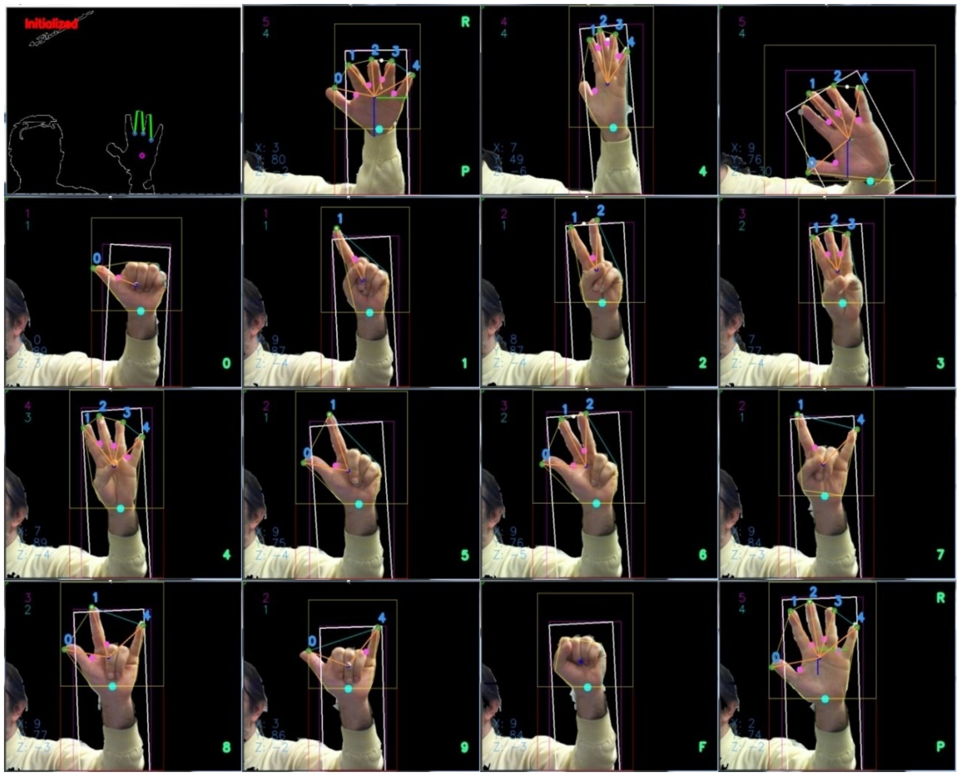


Figure 7.9. Demo of discrete/analog tasks using static/dynamic gestures.

7.1.2. Touch-based Module

For each stage, touch-based controls are defined (based on the same algorithm of Clutch-Gear-Rest explained in the vision-based module) corresponding to the actions to be performed. Table 7.3 shows the settings for desktop controls (mouse and keyboard).

Table 7.3. Design for mouse/keyboard.

Activity	Task	Action					
A) Set/Reset	*Connect pointer*	Move					
B) Activation	*Select*	S Start	Move	Enter Select			
C) City-road	*Limited speed*	S Start	Drag				
D) Road-break	*Jump*	S Start	Move	Space bar Jump			
E) Narrow-road	*Resize + Move*	S Start	Move	↓ ↓ ← Resize & Drag			
F) Bridge	*Resize*	S Start	Move	↓ ↓ Resize		Tab Next	
	Move						← Drag
G) Cruise-control	*Rotate*	S Start	Move	Ctrl ↑ Rotate upward		Ctrl → Rotate rightward	
H) Free-way	*Shoot*	S Start	Move	↓ Shoot downward		↑ Shoot upward	
I) Deactivation	*Deselect*	S Start	Move	Del Deselect			

7.2. Discussion

For the third application, a vision-based system capable of interaction with a user's finger/hand gestures is designed and implemented through some developed methods for detection, segmentation, and tracking of a hand object leading to features extraction, classification, and finally gestures recognition. In this research a novel time-efficient method is presented that detects the fingertips after cropping the irrelevant peripheral parts of an input image employing some processes such as background subtraction and wrist area detection. In order to surmount the effect of ambient disturbances such as noise and illumination changes, the system focuses on the brightness-invariant parameters of a color input, while some effective levels of morphological techniques are also superposed. For the purpose of object (hand) tracking, a robust, yet efficient algorithm is designed to automatically detect the possible hand area prior to localizing a proper color search window in the object's region. Given this sampled color, partial hand segmentation is performed for the stage of smooth tracking. Curvature analysis and moments of hand's blob have been used for features extraction. Eventually, an experimentally set angular classification, in a translation-orientation-scale invariant manner, identifies all the fingers for the matching process of a discrete recognition stage. Furthermore, using the proposed Relative Displacement algorithm, the analog gestures are recognized from extracting the hand's rotations in a 3D space and the trajectory of the palm center.

The controlling touch-less and touch-based options have been chosen to cover mixed combinations of static/dynamic (sign, move, and rotation of hand) and pulse/continuous (single/multi key-strikes and mouse move) inputs. The Race game has been designed with a vast range of discrete/analog tasks (action-based, distance-based, and mixed) under some physics engine (limited/unlimited speed and accelerated motion) involved to cover the common actions (select/deselect, navigate/drag, jump/switch, resize, rotate, and shoot) in a game design.

Chapter 8: Case Study of the Race Game

Having the design of the third application (Race game) for the new case study, explained in the previous chapter, the goal of this chapter is to verify our hypothesis proposed in Chapter 4, by comparing two input methods of touch-based (keyboard/mouse) and finger/hand gestures (webcam), while performing various tasks in a 2D Race game on a desktop display. Figure 8.1 shows a snapshot of the experiment.

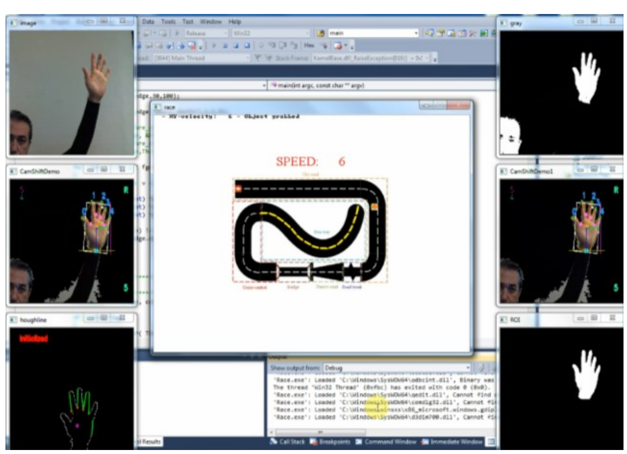


Figure 8.1. Experiment in the Race game.

8.1. User Experiments and Analysis

In order to evaluate the quality of feedback, from the system to the user, four testing sessions have been designed in this experiment, based on the visibility of design (as one of the considerations for product design, proposed in the heuristic evaluation by Nielsen [20]), as follows.

1) **Touch-based with good visibility:** using mouse/keyboard in the same direction of display (close proximity), with speedometer in City-road section, and complete all other sections.
2) **Touch-based with poor visibility:** using mouse/keyboard in a different direction from display (far proximity), without speedometer in City-road section, and wave all other sections by starting and finishing actions.
3) **Touch-less with good visibility:** using gestures in the same direction of display (close proximity), with speedometer in City-road section, and complete all other sections.
4) **Touch-less with poor visibility:** using gestures in a different direction from display (far proximity), without speedometer in City-road section, and wave all other sections by starting and finishing actions.

The experiment instructions are explained to the participants during the training phase. The trial is divided into three phases:

1) Training phase: to get familiar with the application
2) Test phase: the main process to observe the results
3) Satisfaction phase: to complete the questionnaire

In order to compare and evaluate the satisfaction/performance of different input systems in the Race game UI, a user experiment is designed with three variables: input modality (actions of gesture and touch-based), task (activation, city-road, road-break, etc.), and visibility (good and poor visibilities), as shown in Table 8.1. The main criteria in the questionnaire are ease of use, fatigue, naturalness, pleasantness, movability, and overall satisfaction; while the required parameters for efficiency and effectiveness are automatically extracted by the application (Table 8.2) for evaluation of performance. Appendix C gives more details about this experiment.

Table 8.1. Elements in the user experiment for the Race game.

Feedback of task:	Gesture:	Touch-based:
1- Select/Pick	1- Hand (move, sign)	1- Mouse, Keyboard (single strike)
2- Limited-speed Move/Drag	2- Hand (move)	2- Mouse
3- Jump/Switch	3- Hand (move, sign)	3- Mouse, Keyboard (single strike)
4- Parallel Resize & Move	4- Hand (move, signs)	4- Mouse, Keyboard (multi strikes)
5- Serial Resize & Move	5- Hand (move, signs)	5- Mouse, Keyboard (multi strikes)
6- Rotate	6- Hand (move, signs, rotate)	6- Mouse, Keyboard (multi strikes)
7- Shoot/Look	7- Hand (move, signs)	7- Mouse, Keyboard (multi strikes)
8- Deselect/Release	8- Hand (move, sign)	8- Mouse, Keyboard (single strike)
9- Start & Finish	9- Hand (signs)	9- Keyboard (multi strikes)

As shown in Table 8.2, for each type of input, the experiment observer (the program) keeps a record of time (TT, the total time in a task), effective distance (SD, the shortest distance traveled in a task), entire distance (RM, the real motion traveled in a task), number of actions (NA, the number of move/action to do a task), number of trials including errors (NT, the number of move/action to do a task, including the speed/border/select errors), and speedometer (V_t, the object's velocity at every second). The Race game has all three types of activity: action-based, distance-based, and mixed (as explained in Section 3.1).

Table 8.2. Observations on different tasks (X means not applicable).

Activity/Task		TT	SD	RM	NA	NT	V_t	Type
Activation	*Select*		X	X			X	Action
City-road	*Limited speed*							Mixed
Road-break	*Jump*		X	X			X	Action
Narrow-road	*Resize + Move*						X	Mixed
Bridge	*Resize*		X	X			X	Action
	Move						X	Mixed
Cruise-control	*Rotate*						X	Mixed
Free-way	*Shoot*				X	X	X	Distance
Deactivation	*Deselect*		X	X			X	Action
Open & Close	*Start + Finish*		X	X			X	Action

This study is conducted using 20 participants (12 males and 8 females) of different ages (from 15 to 44 years old, with average of 30), and all except one were right-handed. Ethics approval is secured for participant surveys.

8.1.1. Results

The average values of *weighted satisfaction criteria* ($\tilde{X}$), as explained in Section 3.2, show the participants' preferences, for the action-based tasks of the Race game, in the order (from high to low) of pleasantness (6.8), non-fatiguing (6.6), easiness (6.6), naturalness (6.2), and movability (5.9). The order (from high to low) for the mixed tasks of the Race game is pleasantness (6.8), non-fatiguing (6.6), naturalness (6.5), easiness (6.4), and movability (6.0). The order (from high to low) for the distance-based tasks of the Race game is pleasantness (6.8), non-fatiguing (6.5), naturalness (6.5), easiness (6.2), and movability (5.9). Figure 8.2 illustrates the orders.

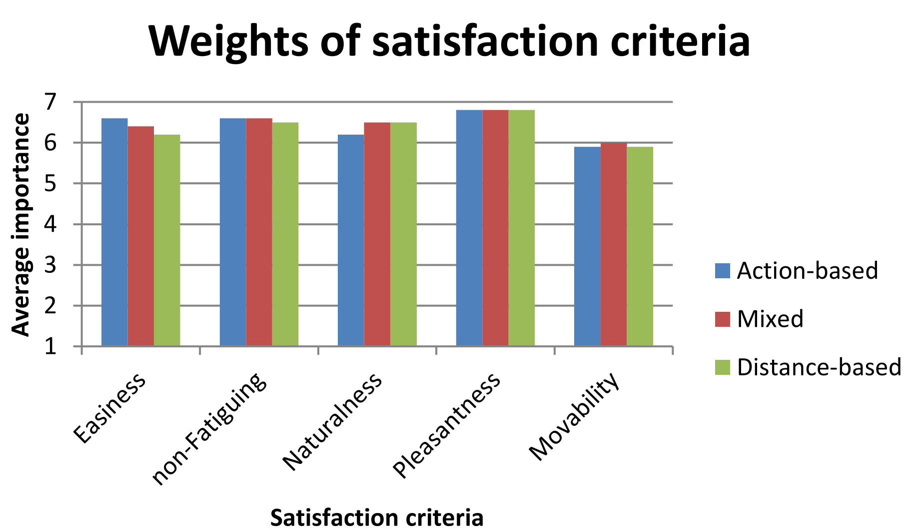


Figure 8.2. Weighted satisfaction criteria for the Race game.

Applying the experiment data into the formulas explained in Chapter 3, Tables C.13 to C.16 enlist the measurements of all required elements to acquire the performance factors for various activities in good and poor visibilities. Also, following the same method explained in Chapter 3 and the details of experiment, Tables 8.3 and 8.4 present the IoC for the Race application.

Table 8.3. Index of Complexity (IoC) for actions in the Race application (part 1).

	Feedback of task	Touch-less modality	Touch-based modality
Action	Select/Pick	Hand (move, sign)	Mouse & Key
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign1 (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonEnter (1D)
IoC	(A) + (2 × L) + (2 × L) + (A) = **6**	(A) + (2 × L) + (A) = **4**	(A) + (2 × L) + (A) = **4**
Action	Limited-speed Move/Drag	Hand (move)	Mouse
Basic-quantities	Activity_action (1D) + Object_limittedVelocity (2D) + Target_collision (2D) + Target_action (1D)	Hand_palm (1D) + Hand_move (2D)	KB_buttonS (1D) + Mouse_move (2D)
IoC	(A) + ((2 × L) + T^{-1}) + (2 × L) + (A) = **7**	(A) + (2 × L) = **3**	(A) + (2 × L) = **3**
Action	Jump/Switch	Hand (move, sign)	Mouse & Key
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign2 (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonSpace (1D)
IoC	(A) + (2 × L) + (2 × L) + (A) = **6**	(A) + (2 × L) + (A) = **4**	(A) + (2 × L) + (A) = **4**
Action	Parallel Resize & Move	Hand (move, signs)	Mouse & Keys
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_resizeAction (1D) + Object_resizeAction (1D) + Object_translation (2D) + Target_collision (2D) + Target_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign3 (1D) + Hand_fist (1D) + Hand_sign3 (1D) + Hand_move (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonDown (1D) + KB_buttonDown (1D) + Mouse_move (1D)
IoC	(A) + (2 × L) + (2 × L) + (A) + (A) + (2 × L) + (2 × L) + (A) = **12**	(A) + (2 × L) + (A) + (A) + (A) + (L) = **7**	(A) + (2 × L) + (A) + (A) + (L) = **6**
Action	Serial Resize & Move	Hand (move, signs)	Mouse & Keys
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_resizeAction (1D) + Object_resizeAction (1D) + Object_nextAction (1D) + Object_translation (2D) + Target_collision (2D) + Target_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign3 (1D) + Hand_fist (1D) + Hand_sign3 (1D) + Hand_signI (1D) + Hand_move (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonDown (1D) + KB_buttonDown (1D) + KB_buttonTab (1D) + Mouse_move (1D)
IoC	(A) + (2 × L) + (2 × L) + (A) + (A) + (A) + (2 × L) + (2 × L) + (A) = **13**	(A) + (2 × L) + (A) + (A) + (A) + (A) + (L) = **8**	(A) + (2 × L) + (A) + (A) + (A) + (L) = **7**

Table 8.4. Index of Complexity (IoC) for actions in the Race application (part 2).

	Feedback of task	**Touch-less modality**	**Touch-based modality**
Action	**Rotate**	**Hand (move, signs, rotate)**	**Mouse & Keys**
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_fixedVelocity (2D) + Object_rotationLeftUp (2D) + Object_rotationUpRight (2D) + Target_collision (2D) + Target_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign7 (1D) + Hand_rotation (2D) + Hand_fist (1D) + Hand_sign7 (1D) + Hand_rotation (2D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonCtrl (1D) + KB_buttonUp (1D) + KB_buttonCtrl (1D) + KB_buttonRight (1D)
IoC	(A) + (2 × L) + (2 × L) + ((2 × L) + T^{-1}) + (2 × r) + (2 × r) + (2 × L) + (A) = **15**	(A) + (2 × L) + (A) + (2 × r) + (A) + (A) + (2 × r) = **10**	(A) + (2 × L) + (A) + (A) + (A) + (A) = **7**
Action	**Shoot/Look**	**Hand (move, signs)**	**Mouse & Keys**
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_fixedAcceleration (2D) + Object_translationRightDown (2D) + Object_translationRightUp (2D) + Target_collision (2D) + Target_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign4 (1D) + Hand_sign4 (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonDown (1D) + KB_buttonUp (1D)
IoC	(A) + (2 × L) + (2 × L) + ((2 × L) + T^{-2}) + (2 × L) + (2 × L) + (2 × L) + (A) = **16**	(A) + (2 × L) + (A) + (A) = **5**	(A) + (2 × L) + (A) + (A) = **5**
Action	**Deselect/Release**	**Hand (move, sign)**	**Mouse & Key**
Basic-quantities	Activity_action (1D) + Pointer_translation (2D) + Object_collision (2D) + Object_action (1D)	Hand_palm (1D) + Hand_move (2D) + Hand_sign9 (1D)	KB_buttonS (1D) + Mouse_move (2D) + KB_buttonDel (1D)
IoC	(A) + (2 × L) + (2 × L) + (A) = **6**	(A) + (2 × L) + (A) = **4**	(A) + (2 × L) + (A) = **4**
Action	**Start & Finish**	**Hand (signs)**	**Keys**
Basic-quantities	Activity_action (1D) + Activity_action (1D)	Hand_palm (1D) + Hand_signA (1D)	KB_buttonS (1D) + KB_buttonF (1D)
IoC	(A) + (A) = **2**	(A) + (A) = **2**	(A) + (A) = **2**

Inserting the information acquired from Tables C.13, C.14, C.15, C.16, 8.3, and 8.4 into the equations proposed in Chapter 3, effectiveness (E_s), efficiency (E_y), simplicity (*RoS*), and similarity (*IoS*) are calculated and listed in Table 8.5 to 8.7. This table also presents the measurements for the overall acceptability (*OA*), easiness ($\hat{E}$), non-fatiguing ($\hat{F}$), naturalness ($\widehat{N}$), pleasantness ($\hat{P}$), movability ($\widehat{M}$), and adjusted satisfaction (S_a) as explained in Chapter 3.

Table 8.5. Measurements of usability elements for action-based interactions in good and poor visibilities of the Race game.

			Usability Factors										
			Independent variables					Dependent variables					
			OA	E_s	E_y	***RoS***	***IoS***	S_a	$\hat{E}$	$\hat{F}$	$\hat{N}$	$\hat{P}$	$\hat{M}$
Context in use	***1***	***[Good | Action | Deactivation | KB]***	6.0	0.9	0.3	1.5	-2	6.2	6.6	6.6	6.0	6.0	5.8
	2	***[Good | Action | Road-break | Gesture]***	6.3	0.7	0.2	1.5	-2	6.3	6.6	6.5	6.0	6.0	6.4
	3	***[Good | Action | Activation | Gesture]***	6.7	0.7	0.1	1.5	-2	6.4	6.6	6.6	6.4	6.1	6.3
	4	***[Good | Action | Deactivation | Gesture]***	6.7	0.7	0.2	1.5	-2	6.5	6.7	6.6	6.5	6.1	6.6
	5	***[Good | Action | Activation | KB]***	6.9	0.9	0.3	1.5	-2	6.3	6.7	6.7	6.6	6.3	5.2
	6	***[Good | Action | Road-break | KB]***	6.9	0.9	0.3	1.5	-2	6.3	6.7	6.7	6.6	6.3	5.2
	7	***[Poor | Action | Start and Finish | KB/Mouse]***	3.5	0.7	0.3	1.0	0	3.5	3.3	3.5	4.8	3.1	3.0
	8	***[Poor | Action | Start and Finish | Gesture]***	3.9	0.6	0.3	1.0	0	4.0	3.5	3.7	4.9	3.1	4.6
	9	***[Poor | Action | Activation | KB/Mouse]***	4.2	0.6	0.1	1.5	-2	4.0	4.8	5.2	3.1	3.2	3.5
	10	***[Poor | Action | Activation | Gesture]***	4.2	0.6	0.1	1.5	-2	4.5	4.7	5.3	3.6	3.2	5.6

Table 8.6. Measurements of usability elements for mixed interactions in good and poor visibilities of the Race game.

			Usability Factors										
			Independent variables					Dependent variables					
			OA	E_s	E_y	***RoS***	***IoS***	S_a	$\hat{E}$	$\hat{F}$	$\hat{N}$	$\hat{P}$	$\hat{M}$
Context in use	***1***	***[Good | Mixed | Cruise-control | KB]***	5.2	0.4	1.2	2.1	-8	5.6	5.2	5.2	5.6	5.2	6.8
	2	***[Good | Mixed | City-road | Gesture]***	5.6	0.6	1.9	2.3	-4	5.9	5.6	5.6	5.7	5.7	6.9
	3	***[Good | Mixed | Bridge | KB/Mouse]***	6.0	0.8	0.4	1.8	-6	5.9	5.4	5.6	6.1	6.0	6.4
	4	***[Good | Mixed | Bridge | Gesture]***	6.0	0.7	0.4	1.6	-5	6.0	5.4	5.6	6.1	6.0	6.9
	5	***[Good | Mixed | Narrow-road | KB/Mouse]***	6.3	0.9	0.4	2	-6	6.2	6.0	6.0	6.2	6.3	6.5
	6	***[Good | Mixed | Narrow-road | Gesture]***	6.3	0.8	0.5	1.7	-5	6.5	6.3	6.3	6.5	6.4	6.9
	7	***[Good | Mixed | Cruise-control | Gesture]***	6.3	0.5	2.4	1.5	-5	6.4	6.4	6.2	6.5	6.2	6.7
	8	***[Good | Mixed | City-road | Mouse]***	6.7	0.7	2.7	2.3	-4	6.2	6.3	6.5	6.1	6.0	6.1
	9	***[Poor | Mixed | City-road | Gesture]***	***5.0***	***0.5***	***1.5***	***2.3***	***-4***	***5.6***	***5.6***	***5.4***	***5.2***	***5.0***	***6.7***
	10	***[Poor | Mixed | City-road | KB/Mouse]***	***5.2***	***0.6***	***2.2***	***2.3***	***-4***	***5.7***	***5.7***	***5.7***	***5.5***	***5.3***	***6.2***

Table 8.7. Measurements of usability elements for distance-based interactions in good visibility of the Race game.

			Usability Factors										
			Independent variables					Dependent variables					
			OA	E_s	E_y	*RoS*	*IoS*	S_a	$\hat{E}$	$\hat{F}$	$\hat{N}$	$\hat{P}$	$\hat{M}$
Context in use	*1*	***[Good | Distance | Free-way | Gesture]***	4.2	0.8	5.8	3.2	-11	4.8	3.9	6.2	4.2	3.9	5.8
	2	***[Good | Distance | Free-way | KB]***	5.2	0.9	6.2	3.2	-11	5.4	4.2	6.4	5.6	5.6	5.2

8.1.2. Analysis

In this section, the data provided in Tables 8.5 to 8.7, for three types of interaction in the Race game, are analyzed. The first two tables include the action-based and mixed interactions respectively, in both designs of good and poor visibilities. The third table includes the distance-based interactions only in the good visibility design, since they are waved in the poor visibility design. The analyses follow the process explained in Section 4.2.

8.1.2.1. Action-based

Easiness: Among the usability factors of interest for this hypothesis, *OA* and *RoS* have strong positive correlations of 0.964 and 0.832 respectively with $\hat{E}$ (Figure 8.3). A multiple linear regression (Table C.17) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, easiness is related to at least one of the usability factors of overall acceptability, efficiency, and simplicity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-values of 0.000 and 0.007 on *OA* and *RoS* respectively, which means they are significantly related to easiness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 8 and 9 in Table 8.5 shows that a gestural interaction with simpler design makes it easier (due to less effort) than the touch-based interaction.

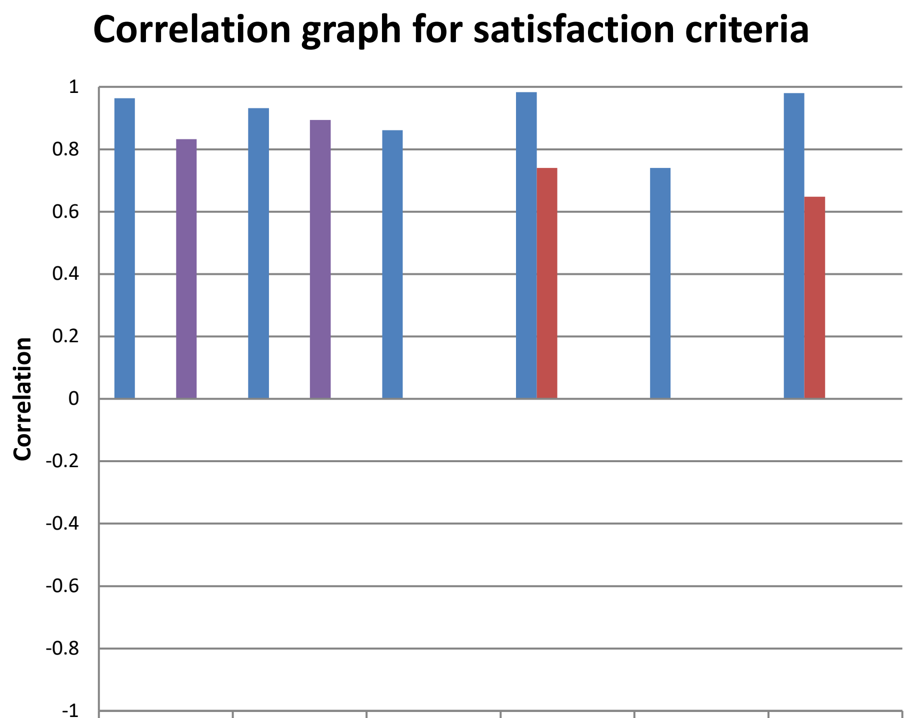
Non-fatiguing: Among the usability factors of interest for this hypothesis, *OA* and *RoS* have strong positive correlations of 0.964 and 0.894 respectively with $\widehat{F}$ (Figure 8.3). A multiple linear regression (Table C.18) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, non-fatiguing is related to at least one of the usability factors of overall acceptability and simplicity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-values of 0.000 and 0.000 on *OA* and *RoS* respectively, which means they are significantly related to non-fatiguing. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 8 and 9 in Table 8.5 shows that a gestural interaction with simpler design makes it less fatiguing (due to less effort) than a touch-based interaction.

Naturalness: Among the usability factors of interest for this hypothesis, *OA* has a strong positive correlation of 0.861 with $\widehat{N}$ (Figure 8.3). A multiple linear regression (Table C.19) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, naturalness is related to at least one of the usability factors of overall acceptability and similarity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.000 on *OA*, which means it is significantly related to naturalness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 1 and 4 in Table 8.5 shows that a gestural interaction on the task of deactivation makes it more natural (due to better match between system and real world) than the touch-based control on the same task.

Pleasantness: Among the usability factors of interest for this hypothesis, *OA* and E_s have meaningful correlations of 0.983 and 0.740 respectively with $\widehat{E}$. In other words, $\widehat{E}$ has a strong positive correlation with *OA*, while there is a moderate correlation with E_s (Figure 8.3). A multiple linear regression (Table C.20) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, pleasantness is related to at least one of the usability factors of overall acceptability and efficiency. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.000 on *OA*, which means it is significantly related to pleasantness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 6 and 7 in Table 8.5 shows that an action with a good visibility is performed with less error and consequently more pleasant (due to better visibility) than under a poor visibility.

Movability: Among the usability factors of interest for this hypothesis, *OA* has a moderate positive correlation of 0.740 with $\widehat{M}$ (Figure 8.3). A multiple linear regression (Table C.21) shows a significant F-value (0.014), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, movability is related to the usability factor of overall acceptability. The positive correlation shows the direction of this relationship. Further analysis of T-test shows the significant P-value of 0.014 on *OA*, which means it is significantly related to movability. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 7 and 8 in Table 8.5 shows that a gestural interaction feels more pleasant (due to better flexibility) than using a touch-based control.

Adjusted satisfaction: Among the usability factors of interest for this hypothesis, *OA* and E_s have meaningful correlations of 0.980 and 0.648 respectively with S_a. In other words, S_a has a strong positive correlation with *OA*, while there is a moderate correlation with E_s (Figure 8.3). A multiple linear regression (Table C.22) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, adjusted satisfaction is related to at least one of the usability factors of overall acceptability, efficiency, and effectiveness. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.000 on *OA*, which means it is significantly related to adjusted satisfaction. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 7 and 8 in Table 8.5 shows that a gestural interaction feels more satisfactory (due to better flexibility, as a dominant effect) than using a touch-based control.



	Easiness	non-Fatiguing	Naturalness	Pleasantness	Movability	Adjusted Satisfaction
OA	0.964	0.932	0.861	0.983	0.74	0.98
Es	N.M.C.	N.M.C.	N.M.C.	0.74	N.M.C.	0.648
Ey	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.
RoS	0.832	0.894	N.M.C.	N.M.C.	N.M.C.	N.M.C.
IoS	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.

Figure 8.3. Correlations of usability factors of hypotheses with satisfaction criteria in the action-based Race game (N.M.C. stands for "no meaningful correlation", with an absolute value of less than 0.25).

8.1.2.2. Mixed

Easiness: Among the usability factors of interest for this hypothesis, OA and E_y have meaningful correlations of 0.699 and 0.373 respectively with $\hat{E}$. In other words, $\hat{E}$ has a moderate positive correlation with OA, while there is a weak correlation with E_y (Figure 8.4). A multiple linear regression (Table C.23) does not show a significant F-value (0.075), which means that the null hypothesis is not rejected. Therefore, easiness is unrelated to the usability factors of overall acceptability, efficiency, and simplicity; even though there is a moderate positive correlation between easiness and overall acceptability. This result is in accordance with its fourth level in the users' feedback about the importance of easiness in this application (as illustrated in figure 8.2).

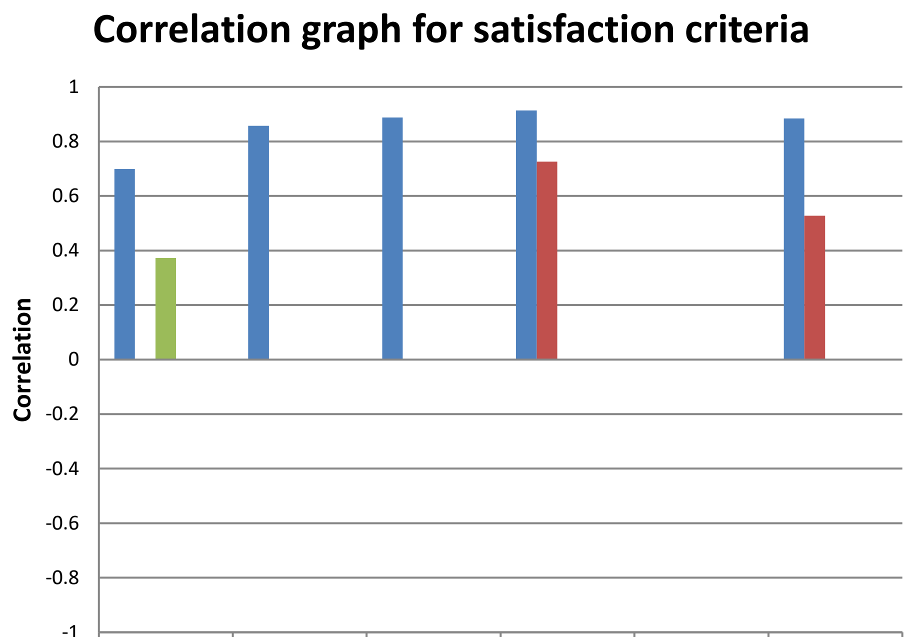
Non-fatiguing: Among the usability factors of interest for this hypothesis, only *OA* has a strong positive correlation of 0.857 with $\hat{F}$ (Figure 8.4). A multiple linear regression (Table C.24) shows a significant F-value (0.005), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, non-fatiguing is related to at least one of the usability factors of overall acceptability and simplicity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.002 on *OA*, which means it is significantly related to non-fatiguing. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 1 and 7 in Table 8.6 shows that a gestural interaction on a task of cruise-control feels less fatiguing (due to better user control) than using a touch-based control.

Naturalness: Among the usability factors of interest for this hypothesis, only *OA* has a strong positive correlation of 0.888 with $\hat{N}$ (Figure 8.4). A multiple linear regression (Table C.25) shows a significant F-value (0.002), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, naturalness is related to at least one of the usability factors of overall acceptability and similarity. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.000 on *OA*, which means it is significantly related to naturalness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 2 and 7 in Table 8.6 shows that a touch based interaction on a city-road task feels more natural (due to better controlling) than using a gestural control.

Pleasantness: Among the usability factors of interest for this hypothesis, *OA* and E_s have meaningful correlations of 0.914 and 0.726 with $\hat{P}$ (Figure 8.4). In other words, $\hat{P}$ has a strong positive correlation with *OA*, while there is a moderate positive correlation with E_s. A multiple linear regression (Table C.26) shows a significant F-value (0.000), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, pleasantness is related to at least one of the usability factors of overall acceptability and effectiveness. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.003 on overall acceptability, which means it is significantly related to pleasantness. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 8 and 10 in Table 8.6 shows that a touch based interaction with a good visibility causes less error (due to better visibility) and consequently makes it more pleasant than with a poor visibility.

Movability: Among the usability factors of interest for this hypothesis, no one has a meaningful correlation with $\widehat{M}$ (Figure 8.4). Lacking of a meaningful correlation on *OA* shows that the users have low preference on movability to be considered in the design of this application. A multiple linear regression (Table C.27) also shows no significant F-value (0.581), which means that the null hypothesis is not rejected. Therefore, movability is unrelated to the usability factor of overall acceptability.

Adjusted satisfaction: Among the usability factors of interest for this hypothesis, *OA* and E_s have meaningful correlations of 0.884 and 0.527 with $\hat{P}$ (Figure 8.4). In other words, S_a has a strong positive correlation with *OA*, while there is a moderate positive correlation with E_s. A multiple linear regression (Table C.28) shows a significant F-value (0.019), which means the null hypothesis can be rejected and the alternative hypothesis is accepted. Therefore, adjusted satisfaction is related to at least one of the usability factors of overall acceptability and effectiveness. The positive correlations show the directions of these relationships. Further analysis of T-test shows the significant P-value of 0.012 on only *OA*, which means it is significantly related to adjusted satisfaction. The comparisons on the experiment results also confirm this conclusion. For example, comparing rows 1 and 7 in Table 8.5 shows that a gestural interaction feels more satisfactory (due to better flexibility, as a dominant effect) than using a touch-based control.



	Easiness	non-Fatiguing	Naturalness	Pleasantness	Movability	Adjusted Satisfaction
OA	0.699	0.857	0.888	0.914	N.M.C.	0.884
Es	N.M.C.	N.M.C.	N.M.C.	0.726	N.M.C.	0.527
Ey	0.373	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.
RoS	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.
IoS	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.	N.M.C.

Figure 8.4. Correlations of usability factors of hypotheses with satisfaction criteria in the mixed-based Race game (N.M.C. stands for "no meaningful correlation", with an absolute value of less than 0.25).

8.1.2.3. Distance-based

The distance-based activity of "shoot" is designed in Freeway, and only for the good visibility. To avoid complexity of the experiment, it is replaced with the action-based activity of "start & finish" in the poor visibility design. Easiness, non-fatiguing, naturalness, pleasantness, and adjusted satisfaction are all strongly correlated to OA, E_s, and E_y. On the contrary, movability is in a negative correlation to OA, which shows that the users have no preference on movability to be considered in the design of this application. No more regression analysis has been done since the number of samples is low. Therefore, the hypothesis cannot be verified statistically. However, according to the positive correlations on the experiment results, all alternative hypotheses are accepted, except for H5 (hypothesis for movability).

8.2. Discussion

Similar to the previous experiments, the wide range of the gamers' skills and age span of the participants conclude that the findings of this study are general. According to the provided analyses, it can be summarized that for most of the cases satisfaction and effectiveness are the major impacts on usability. Consequently, it is inferred that satisfaction and effectiveness are also the main factors of usability affected by the quality of visibility. Having found the main usability factors affecting each interaction, the next step will be the possible solutions based on the sources of violation. Moreover, the results of effectiveness/efficiency in some task repetitions (e.g., start-finish) may provide a learning curve for the type of task.

Further analysis, based on the interaction types (action-based and mixed) and the types of input modality (gesture- and touch-based) involved in the Race application, shows the following results: In the action-based design, the gesture- and touch-based modalities show a similar behavior on the correlations of satisfaction criteria with the controlling factors, where effectiveness has a superior effect compared to efficiency. As expected, this is due to the nature of action-based game applications, in which the accuracy takes a more significant role than the duration of performing a task. On the other hand, according to the findings from the distance-based application of Box (Chapter 5), effectiveness and efficiency have more effects in gesture-based and touch-based modalities respectively. A mixed design includes action- and distance-based interactions. Therefore, from the aforementioned types of interaction, it may be expected to have effectiveness as the main controlling factor in gesture-based, and efficiency/effectiveness as the main controlling factors in touch-based. This expectation is confirmed from the results of experiment in the mixed design of the Race application, where effectiveness has a superior effect compared to efficiency, in both gesture- and touch-based modalities. Furthermore, the provided effort analyses show that the task of rotate has a higher level of satisfaction/performance (consequently higher overall acceptability), using gesture than keyboard. This has been expected, since the hand rotation is a better natural match to the real world task, as the metric of similarity represents it. In addition, the task of parallel, in narrow-road, has slightly a higher overall acceptability than the task of serial, in bridge. This is due to the increased effectiveness and satisfaction.

Finally, similar to the findings from the Box and Slingshot applications (Chapters 5 and 6), the correlation values of overall acceptability for satisfaction criteria are in a similar flow as of the average values of *weighted satisfaction criteria* (as demonstrated in Figure 8.2). As an instance for the critical values with significant differences in the mixed and distance-based tasks, movability has no meaningful positive correlation with overall acceptability. This is also seen in the users' ranking for the importance of different satisfaction criteria, where movability is the least important one for this application. Moreover, the comparison of the correlation results, on naturalness and overall acceptability in the Slingshot (Figure 6.6) and Race (Figures 8.3 and 8.4) applications, shows that the differences of non-meaningful positive and meaningful positive correlations respectively come from the levels of importance for naturalness, ranked by the users dissimilarly due to the different UI designs. Therefore, it is approved that the user's judgment can be reliably considered for setting up the relationships between the observational controlling factors and the satisfaction criteria. The tuning process is the next step, following the loop procedure illustrated in Figure 4.1. The final analysis of hypotheses is summarized in Table 8.8.

Table 8.8. Hypothesis evaluation in the Race application.

	Race (action-based)		**Race (mixed-based)**		**Race (distance-based)**	
	Significant positive correlation	**Hypothesis verification**	**Significant positive correlation**	**Hypothesis verification**	**Significant positive correlation**	**Hypothesis verification**
$\hat{\mathbf{E}}$	*OA, RoS*	✔	—	✘	*N/A*	✔
$\hat{\mathbf{F}}$	*OA, RoS*	✔	*OA*	✔	*N/A*	✔
$\hat{\mathbf{N}}$	*OA*	✔	*OA*	✔	*N/A*	✔
$\hat{\mathbf{P}}$	*OA*	✔	*OA*	✔	*N/A*	✔
$\hat{\mathbf{M}}$	*OA*	✔	—	✘	*N/A*	✘
$\mathbf{S_a}$	*OA*	✔	*OA*	✔	*N/A*	✔

Chapter 9: Conclusion

9.1. Summary

The focus of this dissertation is on designing quantitative approaches to evaluate locomotion HCI systems, such that the need for massive-questionnaires is reduced. For this purpose, this thesis presents new framework for measurement of satisfaction and performance (including effectiveness and efficiency) under the usability structure of HCI systems.

The study in this thesis takes a role of user experience design (UX), which includes interaction design (the design of interactive aspects of a product), usability engineering (evaluating products using usability methods and principles), and information architecture (planning and structuring interactive products). It is structured to help designers understand how to design interactive products that fit with people's need and desire. It has been said that "There are no good user interfaces; just user interfaces that fit". Therefore, in order to gain a maximum fitness, a minimum occurrence of malfunctions (breakdowns in usability) is needed in the context of the expected use of the system.

The Usability framework, proposed in this research, illustrates a big picture of the UI analysis, by the relative values of usability factors, before/after a prototype design. This framework is especially useful when an alpha version of an HCI application is available to be tested through this mechanism and find the correlations of major usability factors and the users' overall acceptability. As a further evaluation in a user-centric application, this correlation can be studied for the usability of sub-factors in satisfaction criteria, such as easiness, non- fatiguing, naturalness, pleasantness, and movability.

On the other hand, this dissertation proposes a formal framework to evaluate the locomotion HCI systems based on the level of effort/complexity designed for interactions. Through the effort-based model, the effects of various input modalities and feedback of tasks are analyzed based on their complexities, defined in new metrics of simplicity and similarity.

The focus of the research in this thesis for input modalities is on kinematic human gestures in a vision-based system versus desktop controllers. In order to develop the generality of the results, unbiased graphical user interfaces are considered for two significant purposes: navigating (including translation and rotation) and switching. These two sets of information can be found in many other applications in various fields such as robotics, education, rehabilitation, smart home, virtual/augmented reality, tele-operation, etc.

Through the three case studies of the Box, Slingshot, and Race game —designed with the choice of natural gestures and enriched with physics engine— the proposed usability framework is examined for verification of usability hypotheses proposed in this dissertation. Analysing these hypotheses shows that overall acceptability, as the main controlling usability factor, is correlated with the user's perception. Having this fact, the user interface designers can confidently develop the user-centric systems based on the users' opinions about the importance and relationships among the usability factors of interest. From this end, the improvement process can be iteratively carried on by adjusting the observational factors of experiments, based on the rule of thumb primitively structured by the users.

As a whole, this thesis proposes formalism for modelling, quantifying, and evaluating of usability factors in locomotion HCI systems. It is shown by the analyses that effectiveness has more effect than efficiency in the action-based design, while it is reverse in the distance-based design. It is also found that the gesture- and touch-based modalities show similar behavior for the correlation patterns between the social and observational factors, in the action-based design. However, the patterns are different in the distance-based design, where the social factors using the touch-based input show more dependency on the controlling factors compared to the gesture-based input. Furthermore, the effort-based factor of simplicity shows more effect on the social factors studied in this dissertation, compared to the similarity factor. Finally, according to the proposed usability model, the HCI designers are recommended to inquire for the users' opinions in the first step of the design process. This will set up reliable relations between the social factors and the observational controlling factors to follow in the fields of interest (e.g., different groups of input modality). Having these application-based relations along with the users' observational (performance) and social (satisfaction) feedback through a user experiment, the next step of the design process will be finding the sources of violation that impact the related controlling factors, and consequently the social dependents. The iterative process of improvement will be performed through the simpler and faster user experiments to gain the observational controlling factors, and redesigning the UI by reducing the related negative impacts in the evaluation process.

9.2. Original Contributions

For the purpose of uniformity, this thesis has re-contextualized the HCI locomotion types in three types of action-based, distance-based, and mix of the first two.

The three main usability factors of effectiveness, efficiency, and satisfaction —defined in ISO 9241-11, as the most popular and accepted usability model— have been quantified in some equations by this thesis. In addition, this model has been expanded by adding a significant element of overall acceptability. This new model supports the usability evaluations in a more user-centric approach, where the factors of performance and satisfaction are evaluated with regard to a ground truth, which is overall acceptability.

Proposing a new effort-based model, based on the complexity of actions in input modality and feedback of task, has provided a new solution to measure the physical/mental complexity. Mental complexity happens in a transition between two interactions, which is still included in this new model. The metric of complexity is a source for quantifying two usability factors of simplicity and similarity.

Through the implementation of three case studies and performing the user experiments, this thesis has shown that how the proposed usability model can be applied in practice.

Finally, a set of hypotheses proposed to facilitate usability evaluations for locomotion HCI systems. The purpose of these hypotheses, which have been mostly confirmed (except for naturalness and movability, in a few cases) through statistical analysis of the proposed usability model, is to predict some social feedback from some observational results.

9.3. Future Work

Although it has been attempted to design and carry out this research as precise and robust as possible, the results of this study are still subject to improvement by increasing the number of experiments.

The proposed method of measurement for complexity, based on the index of complexity, led to defining two new effort-based factors of simplicity and similarity, as controlling factors. Having the index of complexity for input modality and output feedback, some new controlling factors may need to be intuitively defined for better evaluation of social factors.

Finally, for the purpose of generality, testing more HCI applications in various fields, such as robotics, education, rehabilitation, and entertainment provides more sources to verify the hypotheses proposed in this thesis. Therefore, having the current work and preliminary results, the future work plan will focus on the proposed frameworks of usability evaluation for more applications, enriched with higher levels of physics engine, and under various contexts of use.

References

1. Input/output devices and interaction techniques; computing handbook [Online]. http://www.microsoft.com/en-us/research/wp-content/uploads/2016/10/Computing-Handbook-3rd-Edition-Chapter-21.pdf [Accessed: October 2017].

2. T. Hewett, R. Baecker, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong, and W. Verplank, "Association for computing machinery", *ACM SIHCHI*, 1992.

3. M. M. Hasan, and P. K. Mishra, "Hand gesture modeling and recognition using geometric features: a review", *Canadian Journal on Image Processing and Computer Vision*, 2012.

4. T. Schlomer, B. Poppinga, N. Henze, and S. Boll, "Gesture recognition with a Wii controller", *In Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*, 2008.

5. Merriam-Webster Dictionary, [Online]. Available: https://www.merriam-webster.com/dictionary/locomotion [Accessed: December 2016].

6. P. Kortum, "HCI beyond the GUI: design for haptic, speech, olfactory and other nontraditional interfaces", The Morgan Kaufmann series in interactive technologies, pp. 75–106, 2008.

7. C. Boletsis, “The new era of virtual reality locomotion: a systematic literature review of techniques and a proposed typology”, *SINTEF Digital*, 2017.

8. F. Farhadi-Niaki, J. Gerroir, A. Arya, S. A. Etemad, R. Laganière, P. Payeur, and R. Biddle, “Usability study of static/dynamic gestures and haptic input as interfaces to 3D games,” *The Sixth International Conference on Advances in Computer-Human Interactions*, 2013.

9. ISO 9241 Part 11, “Guidance on usability: Ergonomic requirements for office work with visual display terminals”, International Standards Organization, 1998.

10. N. El-Bendary, H. M. Zawbaa, M. S. Daoud, A. E. Hassanien, and K. Nakamatsu, “ArSLAT: Arabic Sign Language Alphabets Translator”, *Computer Information Systems and Industrial Management Applications*, 2010.

11. D. Gorodnichy, and A. Yogeswaran, “Detection and tracking of pianist hands and fingers”, *Third Canadian Conference on Computer and Robot Vision*, 2006.

12. H. An, and D. Kim, “Hand gesture recognition using 3D depth data”, Department of Computer Science and Engineering, Pohang University of Science and Technology, Korea, 2009.

13. D. Lee, and S. Lee, “Vision-based finger action recognition by angle detection and contour analysis”, *Journal of Electronics and Telecommunications Research Institute*, vol. 33(3), pp. 415–422, 2011.

14. Y. Fang, K. Wang, J. Cheng, and H. Lu, “A real-time hand gesture recognition method”, *In Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 995–998, 2007.

15. F. Farhadi-Niaki, R. GhasemAghaei, and A. Arya, “Empirical study of a vision-based depth-sensitive human-computer interaction system”, *The 10th Asia Pacific Conference on Computer Human Interaction*, 2012.

16. A. Jaimes, and N. Sebe, “Multimodal human computer interaction: a survey”, *Computer Vision and Image Understanding*, vol. 108(1–2), pp. 116–134, 2007.

17. G. Ye, "Applying vision to intelligent human-computer interaction", Ph.D. dissertation, Department of Computer Science, Johns Hopkins University, 2005.

18. D. A. Norman, "The psychology of everyday things", Basic Books, New York, 1998.

19. R. Jacob, J. Legget, B. Myers, and R. Pausch, "An agenda for human-computer interaction research: Interaction styles and input/output devices", *Behaviour and Information Technology*, 1993.

20. J. Nielsen, "Heuristic evaluation", *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

21. D. A. Norman, "Emotion design: why we love (or hate) everyday things", Basic Books, 2004.

22. R. GhasemAghaei, R. Biddle, A. Arya, "The MADE framework: multimodal software for affective education", *In Proceedings of EdMedia: World Conference on Educational Media and Technology, AACE*, 2015.

23. R. GhasemAghaei, R. Biddle, A. Arya, "Multimodal software for affective education: UI design", *In Proceedings of EdMedia: World Conference on Educational Media and Technology, AACE*, 2015.

24. R. GhasemAghaei, R. Biddle, A. Arya, "Multimodal software for affective education: UI evaluation", *In Proceedings of EdMedia: World Conference on Educational Media and Technology, AACE*, 2015.

25. R. GhasemAghaei, A. Arya, R. Biddle, "Design practices for multimodal affective mathematical learning" *International Symposium on Computer Science and Software Engineering, IEEE, CSSE*, pp. 1-7, 2015.

26. R. GhasemAghaei, A. Arya, R. Biddle, "Evaluating software for affective education: a case study of affective heuristics", *In Proceedings of EdMedia: World Conference on Educational Media and Technology, AACE*, pp. 567–574, 2016.

27. R. GhasemAghaei, A. Arya, R. Biddle, “Affective walkthroughs and heuristics: evaluating Minecraft Hour of Code”, *International Conference on Human-Computer Interaction*, Springer International Publishing, 2017.

28. N. Bevan, and K. Holdaway, “User needs for user system interaction standards”, Technology Standards, Evans, Butterworth Heinemann, 1993.

29. N. Bevan, “Measuring usability as quality of use”, *Journal of Software Quality*, Issue 4, pp. 115–140, 1995.

30. E. L. Koua, and M. J. Kraak, “A usability framework for the design and evaluation of an exploratory geovisualization environment”, *Proceedings of the International Conference on Information Visualization*, pp. 153–158, 2004.

31. P. Booth, “An introduction to human-computer interaction”, Lawrence Erlbaum Associates, 1989.

32. R. A. Guillemette, “The evaluation of usability in interactive information systems”, In Human factors in information systems, pp. 207–221, 1995.

33. R. Kengeri, C. D. Seals, H. D. Harley, H. P. Reddy, and E. A. Fox, “Usability study of digital libraries”, *International Journal on Digital Libraries*, pp. 157–169, 1999.

34. K. Kim, “A model of digital library information seeking process (DLISP model) as a frame for classifying usability problems”, PhD dissertation, Rutgers University, 2002.

35. B. Shackel, “The concept of usability”, *Proceedings of IBM Software and Information Usability Symposium*, pp. 1–30, 1981.

36. B. Shackel, “Ergonomics in design for usability”, *Proceedings of the BCS HCI Specialist Group*, Cambridge, Cambridge University Press, 1986.

37. B. Shackel, “Usability: Context, framework, definition, design and evaluation”, In Human Factors for Informatics Usability, New York, Cambridge University Press, pp. 21–37, 1991.

38. J. Nielsen, “Usability engineering”, Cambridge, Academic Press, 1993.

39. M. Gluck, A descriptive study of the usability of geospatial metadata; annual review of OCLC research [Online]. Available: http://www.oclc.org/research/publications/arr/1997/gluck/gluck_frameset.htm [Accessed: November 2017].

40. T. Brinck, D. Gergle, and S. D. Wood, "Designing web sites that work: Usability for the web", Morgan Kaufmann, 2002.

41. M. Clairmont, R. Dickstein, and V. Mills, Testing of usability in the design of a new information gateway [Online]. Available: http://www.library.arizona.edu/library/teams/access9798 [Accessed: October 2017].

42. E. Furtado, J. J. Vasco Furtado, F. Lincoln Mattos, and J. Vanderdonckt, "Improving usability of an online learning system by means of multimedia, collaboration and adaptation resources", In Usability evaluation of online learning programs, pp. 69–86, 2003.

43. D. Hix, and H. Rex Hartson, "Developing user interfaces: Ensuring usability through product & process", John Wiley, 1993.

44. J. S. Dumas, and Janice C. Redish, "A practical guide to usability testing", Norwood, Ablex Publishing Co., 1993.

45. A. Oulanov, and E. F. Y. Pajarillo, "CUNY+ Web: Usability study of the Web-based GUI version of the bibliographic database of the City University of New York (CUNY)", The Electronic Library, vol. 20 (6), pp. 481–487, 2002.

46. K. MacLean, "Haptic interaction design for everyday interfaces", *Human Factors and Ergonomics*, vol. 4(1), pp. 149–194, 2008.

47. M. Simoes-Marques, and I. L. Nunes, "Usability of Interfaces", In Tech, *Ergonomics – A System Approach*, 2012.

48. T. Jokela, N. Iivari, J. Matero, and M. Karukka, "The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11", *Proceedings of the Latin American Conference on Human-computer Interaction*, 2003.

49. J. Jeng, “Usability Assessment of academic digital libraries: effectiveness, efficiency, satisfaction, and learnability”, *Libri*, vol. 55(2–3), pp. 96–121, 2008.

50. A. Seffah, M. Donyaee, R. B. Kline, and H. K. Padda, “Usability measurement and metrics: a consolidated model”, *Software Quality Journal*, vol. 14(2), pp. 159–178, 2006.

51. R. K. Mehta, “Integrating physical and cognitive ergonomics”, *IIE Transactions on Occupational Ergonomics and Human Factors*, pp. 83–87, 2016.

52. F. Vallée-Tourangeau, “Interactivity, efficiency, and individual differences in mental arithmetic”, *Experimental Psychology*, pp. 302–311, 2013.

53. C. M. Schlick, C. Winkelholz, F. Motz, S. Duckwitz, and M. Grandt, “Complexity assessment of human computer interaction”, *Journal of Theoretical Issues in Ergonomics Science*, v. 11(3), pp. 151–173, 2009.

54. G. Z. Bedny, and W. Karwowsky, “Human-computer interaction and operators’ performance: optimizing work design with activity theory”, CRC Press, 2011.

55. R. Laban, and F. C. Lawrence, “Effort”, Macdonald & Evan Ltd., London, 1947.

56. R. Laban, “Modern educational dance”, Macdonald & Evan Ltd., London, 1963.

57. A. Camurri, C. Canepa, S. Ghisio, and G. Volpe, “Automatic classification of expressive hand gestures on tangible acoustic interfaces according to Laban’s theory of effort”, *Gesture-Based Human-Computer Interaction and Simulation*, 2007.

58. S. Gibet, N. Courty, and J. F. Kamp, “Gesture in human-computer interaction and simulation”, *International Gesture Workshop*, France, 2005.

59. H. A. Van Veen, H. K. Distler, S. J. Braun, and H. H. Bülthoff, “Navigating through a virtual city: Using virtual reality technology to study human action and perception”, *Future Generation Computer Systems*, vol. 14, pp. 231–242, 1998.

60. M. Nielsen, M. Storring, T. B. Moeslund, and E. Granum, “A procedure for developing intuitive and ergonomic gesture interfaces for HCI”, *Gesture-Based Communication in Human-Computer Interaction - 5th International Gesture Workshop*, pp. 409–420, 2004.

61. G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, L. Bouillon, and J. Vanderdonckt, "A unifying reference framework for multi-target user interfaces", *Interacting with Computer*, vol. 15(3), pp. 289–308, 2003.

62. B. Bomsdorf, and R. Blum, "Early prototyping of 3D-gesture interaction within the presentation-gesture-dialog design space", *In Proceedings of the 16th HCI International*, 2014.

63. B. Frohlich, H. Tramberend, A. Beers, M. Agrawala, and D. Baraff, "Physically-based manipulation on the responsive workbench", *In Proceedings of the IEEE Virtual Reality*, pp. 5–11, 2000.

64. M. Boulabiar, G. Coppin, and F. Poirier, "The study of the full cycle of gesture interaction, the continuum between 2D and 3D", *In Proceedings of the 16th HCI International*, 2014.

65. J. L. Reisman, P.L. Davidson, and J. Y. Han, "A screen-space formulation for 2D and 3D direct manipulation", *In Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, pp. 69, 2009.

66. B. Rouxel, F. Poirier, J. Antoine, and G. Coppin, "What you draw is what you search: the analog gesture", *In Proceedings of the 16th HCI International*, 2014.

67. B. Yoo, J. Han, C. Choi, H. Ryu, D. S. Park, and C. Y. Kim, "3D remote interface for smart displays", *The ACM conference on Human Factors in Computing Systems*, 2011.

68. M. C. Cabral, C. H. Morimoto, and M. K. Zuffo, "On the usability of gesture interfaces in virtual reality environments", *The 2nd Latin American Conference on Human-Computer Interaction*, 2005.

69. N. Villaroman, D. Rowe, and B. Swan, "Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor", *The 12th ACM Special Interest Group Conference on Information Technology Education*, 2011.

70. G. Bailly, R. Walter, J. Muller, T. Ning, and E. Lecolinet, "Comparing free hand menu techniques for distant displays using linear, marking and finger-count menus", *The International Federation for Information Processing*, 2011.

71. J. Kang, D. Seo, and D. Jung, "A study on the control method of 3-dimensional space application using Kinect system", *International Journal of Computer Science and Network Security*, vol. 11(9), 2011.

72. A. Bragdon, R. DeLine, K. Hinckley, and M. R. Morris, "Code Space: touch + air gesture hybrid interactions for supporting developer meetings", *The Interactive Tabletops and Surfaces,* 2011.

73. M. Bhuiyan, and R. A. Picking, "Gesture controlled user interface for inclusive design and evaluative study of its usability", *Journal of Software Engineering and Applications*, 2011.

74. L. C. Ebert, G. Hatch, G. Ampanozi, M. J. Thali, and S. Ross, "You can't touch this: touch-free navigation through radiological images", *Special Area Groups of Educators*, 2011.

75. P. Kortum, "HCI beyond the GUI: design for haptic, speech, olfactory, and other nontraditional interfaces," Morgan Kaufmann Publishers, pp. 75–106, 2008.

76. J. Weissmann, and R. Salomon, "Gesture recognition for virtual reality applications using data gloves and neural networks," *International Conference on Neural Networks*, vol. 3, pp. 2043–2046, 1999.

77. T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill, "A hand gesture interface device", *CHl+GI*, pp. 189–192, 1987.

78. D. Tzovaras, "Multimodal user interfaces: from signals to interaction", Springer, Heidelberg, 2008.

79. J. Nespoulous, P. Perron, and A. R. Lecours, "The biological foundations of gestures: motor and semiotic aspects", Lawrence Erlbaum Associates, Hillsdale, 1986.

80. A. Kendon, "Current issues in the study of gesture, in the biological foundation of gestures: motor and semiotic aspects", Lawrence Erlbaum Associates, Hillsdale, pp. 23–47, 1986.

81. V. Pavlovic, and T. Huang, “Visual interpretation of hand gestures for human-computer interaction: a review”, *The IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19(7), pp. 667, 1997.

82. R. Vatavu, S. Pentiuc, and C. Chaillou, “On natural gestures for interacting in virtual environments”, *Advances in electrical and computer engineering*, vol. 24(5), 2005.

83. M. A. Srinivasan, What is haptics? [Online]. Available: http://www.sensable.com/documents/documents/what_is_haptics.pdf [Accessed: April 2014].

84. Sensible Technologies, The Phantom devices, [Online]. Available: http://www.sensable.com/products/phantom_ghost/phantom-omni.asp [Accessed: April 2014].

85. Immersion Corporation, The Cyber Grasp, Cyber Touch, Cyber Glove devices, [Online]. Available: http://www.immersion.com/3d/products/cyber_grasp.php [Accessed: April 2014].

86. V. Hayward, and M. Cruz-Hernandez, “Tactile display device using distributed lateral skin stretch”, *In Proceedings of the 8th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, vol. 2, pp. 1309–1314, 2000.

87. C. Wagner, S. Lederman, and R. Howe, “A tactile shape display using RC servomotors”, *In Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator*, pp. 354–355, 2002.

88. M. Cavusoglu, D. Feygin, and F. Tendick, “A critical study of the mechanical and electrical properties of the Phantom haptic interface and improvements for high performance control”, *Teleoperators and Virtual Environments*, vol. 11(6), pp. 555–568, 2002.

89. T. Massie, and J. Salisbury, “The Phantom haptic interface: a device for probing virtual objects,” *In Proceedings of Symposium on Haptic Interfaces for Virtual Environment and Teleoperator*, 1994.

90. S. Dimitrov, HCI Challenges, [Online]. Available: www.smilen.net/st/files/st_intro_01.ppt [Accessed: September 2016].

91. A. Chapanis, and R. G. Kinkade, “Design of controls”, *Human engineering guide to equipment design*, Washington, DC, 1972.

92. I. S. MacKenzie, “Input devices and interaction techniques for advanced computing”, *Virtual environments and advanced interface design*, UK: Oxford University Press, pp. 437–470, 1995.

93. D. Glowinski, S. Y. Coll, N. Baron, M. Sanchez, S. Schaerlaeken, and D. Granjean, “Body, space, and emotion: A perceptual study”, *Human Technology*, University of Jyvaskyla, 2017.

94. E. Müller, and S. Lindinger, “Science and skiing IV”, *Meyer & Meyer Sport*, 2009.

95. H. Sharp, Y. Rogers, and J. Preece, “Interaction design: beyond human computer interaction”, John Wiley & Sons, 2007.

96. J. Nielsen, “Finding usability problems through heuristic evaluation”, *Proceedings ACM CHI Conference*, pp. 373–380, 1992.

97. A. Sergeev, Users and goals [Online]. Available: http://ui-designer.net/usability/index.htm. [Accessed: April 2017].

98. A. Holzinger, G. Searle, T. Kleinberger, A. Seffah, and H. Javahery, “Investigating usability metrics for the design and development of applications for the elderly”, *International Conference on Computers for Handicapped Persons*, 2008.

99. Australian Bureau of Statistics, Correlation and causation, [Online]. Available: http://www.abs.gov.au/websitedbs/a3121120.nsf/home/statistical+language+-+correlation+and+causation [Accessed: May 2018].

100. Complete Dissertation by statistics solutions, Establishing cause and effect, [Online]. Available: http://www.statisticssolutions.com/establishing-cause-and-effect [Accessed: May 2018].

101. Statistical Education trough Problem Solving, Pearson correlation, [Online]. Available: http://www.statstutor.ac.uk/resources/uploaded/pearsons.pdf [Accessed: April 2018].

102. Pearson correlation coefficient, Wikipedia [Online]. Available: https://en.wikipedia.org/wiki/Pearson_correlation_coefficient [Accessed: April 2018].

103. L. K. Gray-Steinhauer, Correlation & linear regression, University of Alberta [Online]. Available: https://sites.ualberta.ca/~lkgray/uploads/7/3/6/2/7362679/slides_-_correlation_linearregression.pdf [Accessed: May 2018].

104. S. D. Jost, Linear correlation, DePaul University [Online]. Available: http://condor.depaul.edu/sjost/it223/documents/correlation.htm [Accessed: May 2018].

105. W. Benson, Correlation and multiple regressions in Excel, Nevada State College [Online]. Available: https://www.youtube.com/watch?v=zoAEkmbW4ak [Accessed: May 2018].

106. E. Jesse, Linear regression & causal models [Online]. Available: https://www.youtube.com/watch?v=0ZXNBSERdZ4 [Accessed: May 2018].

107. C. de Roiste, T. Paakkonen, and V. Kalakoski, Cognitive ergonomics, Osh Wiki [Online]. Available: https://oshwiki.eu/wiki/Cognitive_ergonomics [Accessed: November 2018].

108. Kinect for Windows Sensor Components and Specifications, Microsoft [Online]. Available: https://msdn.microsoft.com/en-us/library/jj131033.aspx [Accessed: April 2016].

109. G. R. Bradski , and A. Kaehler, “Learning OpenCV”, O’Reilly Media Inc., 2013.

110. K. Chalidabhongse, and H. Davis, “Real-time foreground-background segmentation using Codebook model”, Computer Vision Lab, Department of Computer Science, University of Maryland, 2005.

111. P. Fillo, “Hand tracking and gesture recognition using echo state neural networks”, *6th Student Research Conference in Informatics and Information Technologies*, vol. 2, pp. 459–466, 2010.

112. S. K. Kang, M. Y. Nam, and P. K. Rhee, "Color based hand and finger detection technology for user interaction", *IEEE International Conference on Convergence and Hybrid Information Technology*, pp. 229–236, 2008.

113. J. Kilian, "Simple image analysis by moments", OpenCV library documents, 2001.

114. J. L. Raheja, K. Das, and A. Chaudhary, "An efficient real time method of fingertip detection", *7th International Conference on Trends in Industrial Measurements and Automation*, pp. 447–450, 2011.

115. A. Davison, "Killer game programming in Java - Java gaming and graphics programming", O'Reilly, Chapter 5, 2005.

Appendix A: Experiment for the Box Game

A.1. Experimental Instruction, Observation, and Questionnaire

A.1.1. Experiment

To compare two input methods of touch-based (keyboard/mouse) and full-body gestures (Kinect), in a 3D Box game and on two different settings of desktop and large-scale display.

A.1.1.1. Training Session

A.1.1.1.1. Step 1

- Ten minutes of practicing the navigation (moving the pointer on the scene using keyboard/mouse and body movements), to get used to the application by running the following tasks (one type of display is randomly selected for each user to practice):

a. Look around (align the crosshair with) to see the two elevated different boxes at the center of camera view (2D movement).
    i. Keyboard: Press the left and right arrow keys to pan the camera left and right, and the up and down arrow keys to tilt it up and down.
    ii. Gesture (arm): Move your right arm left and right to pan the camera left and right, and move it up and down to tilt the camera up and down.
b. Move and collide the center of camera view with the two un-elevated different boxes (2D movement).
    i. Keyboard: Use the 'W' key to move the camera forward, the 'S' key to move backwards, the 'A' key to move left, and the 'D' key to move right.
    ii. Gesture (waist): Lean left and right to move the camera in the left and right directions and lean forward and backward to move it forwards and backwards.
c. Jump the camera view up to hit the two elevated different boxes (1D movement)
    i. Keyboard: Press the spacebar key to jump the camera up.
    ii. Gesture (legs): Jump up above a certain upward velocity to jump the camera up.
d. Combination of all above three basic tasks in order to collide the center of camera view with the elevated two different boxes (3D movement).
    i. Keyboard: Control the camera view by a combination of all above keyboard controls (arrow, 'W', 'S', 'A', 'D', and spacebar keys) in a 3D movement.
    ii. Gesture (arm + waist + legs): Control the camera view by a combination of all above gestures (arm, waist, and legs) in a 3D movement.

A.1.1.1.2. Step 2

- Five minutes of practicing the navigation (moving the pointer on the scene using keyboard/mouse and arm movements), to get used to the application by running the following tasks (one type of display is randomly selected for each user to practice):

e. Target selection (grab) with moving the small spherical cursor to collide it with the five target boxes (3D movement).
    i. Keyboard/Mouse: Use the up and down arrow keys to move the cursor in and out, and use the mouse to move the cursor left, right, up, and down.
    ii. Gesture (arm): Control the 3D cursor by moving the position of you right arm in all directions. This is similar to a combination of the Look and the Move tasks in Step 1 (but is performed in 3D).

A.1.1.2. Test Session

A.1.1.2.1. Step 1

A.1.1.2.1.1. Keyboard-Desktop

The user is asked to hit the two boxes (in each of four tasks) using the keyboard on a desktop display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point.

A.1.1.2.1.1.1. Task

Run the steps "a-b-c-d (i)" as explained in training session (Section A.1.1.1.1):

1- Look ("a (i)"): Hit the two boxes with the center of camera view (2D movement: left, right, up, and down).

2- Move ("b (i)"): Collide the center of camera view with the two boxes (2D movement: left, right, zoom in, and zoom out).

3- Jump ("c (i)"): Hit the two boxes with the center of camera view (1D movement: jump up).

4- Combination ("d (i)"): Collide the center of camera view with the two boxes (3D movement: left, right, up, down, zoom in, zoom out, and jump up).

A.1.1.2.1.1.2. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table A.1.

Table A.1. Questions for the tasks in Step1.

Satisfaction criteria for primitive tasks			1	2	3	4	5	6	7
1	How easy was to:	look?							
		move?							
		jump?							
		do the combination?							
2	How light (non-fatiguing) was to:	look?							
		move?							
		jump?							
		do the combination?							
3	How natural was to:	look?							
		move?							
		jump?							
		do the combination?							
4	How pleasant was to:	look?							
		move?							
		jump?							
		do the combination?							
5	How flexible (mobility in the area) was to:	look?							
		move?							
		jump?							
		do the combination?							
6	How overall satisfactory was to:	look?							
		move?							
		jump?							
		do the combination?							

A.1.1.2.1.2. Keyboard-LargeScreen

The user is asked to hit the two boxes (in each of four tasks) using the keyboard on a large-screen display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same process as explained in the Section A.1.1.2.1.1 is followed (“a-b-c-d (i)”).

A.1.1.2.1.3. Gesture-Desktop

The user is asked to hit the two boxes (in each of four tasks) using the gestures on a desktop display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same process as explained in the Section A.1.1.2.1.1 is followed ("a-b-c-d (ii)").

A.1.1.2.1.4. Gesture-LargeScreen

The user is asked to hit the two boxes (in each of four tasks) using the gestures on a large-screen display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same process as explained in the Section A.1.1.2.1.1 is followed ("a-b-c-d (ii)").

A.1.1.2.2. Step 2

A.1.1.2.2.1. Keyboard/mouse-Desktop

The user is asked to hit the five boxes using the keyboard/mouse on a desktop display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point.

A.1.1.2.2.1.1. Task

Run the step "e (i)" as explained in training session (Section A.1.1.1.2):

1- Target selection (grab): Collide the small spherical cursor with the five target boxes (3D movement).

A.1.1.2.2.1.2. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table A.2.

Table A.2. Questions for the task in Step 2.

Satisfaction criteria for primitive tasks		1	2	3	4	5	6	7
1	How easy was to grab?							
2	How light (non-fatiguing) was to grab?							
3	How natural was to grab?							
4	How pleasant was to grab?							
5	How flexible (mobility in the area) was to grab?							
6	How overall satisfactory was to grab?							

A.1.1.2.2.2. Keyboard-LargeScreen

The user is asked to hit the five boxes using the keyboard/mouse on a large-screen display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same process as explained in the Section A.1.1.2.2.1 is followed ("e (i)").

A.1.1.2.2.3. Gesture-Desktop

The user is asked to hit the five boxes using the gestures on a desktop display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same processes as explained in the Section A.1.1.2.2.1 are followed ("e (ii)").

A.1.1.2.2.4. Gesture-LargeScreen

The user is asked to hit the five boxes using the gestures on a large-screen display, while the experiment observer keeps a record of the total time and the scores (number of moves in different directions) from the start to the end point. The same processes as explained in the Section A.1.1.2.2.1 are followed ("e (ii)").

A.1.2. Extra Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table A.3.

Table A.3. Extra questions for having a better feedback.

Q1: Your gaming skill's level?

1) Novice
2) Somehow experienced
3) Average experienced
4) Very experienced
5) Expert

Q2: When it comes to playing a 3D computer game, please rank the following options from 1 (the least important) to 7 (the most important). You may use the same rank if they have the same weight of importance to you.

1) Easy to interact
2) Non-fatiguing interaction
3) Natural interaction
4) Fun to use
5) Mobility

Q3: What combination of keyboard/mouse and gestures capabilities would you prefer to use on this application?

1) Just keyboard/mouse
2) Mostly keyboard/mouse
3) Equal use of keyboard/mouse and gestures
4) Mostly gestures
5) Just gestures

Table A.4. User comments.

Q4: What did you like/dislike about these <u>gesture applications</u>? **Like:**   **Dislike:** Q5: What did you like/dislike about these <u>keyboard/mouse applications</u>? **Like:**   **Dislike:** Q6: Please share your opinion about mandatory features (<u>must have</u>) and desirable features (<u>nice to have</u>) to improve the gesture and keyboard/mouse applications? **Gesture:**   **Keyboard/mouse:**

A.2. Extra Results of Experiment

Table A.5. Performance data for interactions in the Box game.

Performance of interactions (distance-based)									
Feedback of task	**Modality**		**Output device**	**TT**	**SD**	**RM**	**AV**	**AR**	**R**
Parallel-find	Touch-based	KB	Desktop	21.69	69.2	109.8	3.19	0.63	2.5
Grab		Mouse-KB		46.78	58.6	104.6	1.25	0.56	2.5
Parallel-find		KB	Large-screen	23.92	69.2	111.6	2.89	0.62	2.5
Grab		Mouse-KB		43.91	58.6	112.7	1.33	0.52	2.5
Parallel-find	Touch-less	Full body	Desktop	39.57	69.2	177.4	1.75	0.39	2.5
Grab		Arm		73.50	58.6	136.3	0.80	0.43	2.5
Parallel-find		Full body	Large-screen	32.74	69.2	153.8	2.11	0.45	2.5
Grab		Arm		56.23	58.6	119.6	1.04	0.49	2.5

Table A.6. Summary output of regression analysis for easiness in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.987563996
R Square	0.975282645
Adjusted R Square	0.956744629
Standard Error	0.163006614
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	4.193715375	1.397905125	52.60987162	0.001136044
Residual	4	0.106284625	0.026571156		
Total	7	4.3			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-0.38631653	0.635375638	-0.60801281	0.57600969	-2.15040211	1.37776905
OA	0.770052504	0.13978428	5.508863379	0.005297253	0.381949122	1.158155885
E_y	0.889148147	0.208476044	4.26498954	0.013001854	0.310325855	1.467970439
RoS	0.507818994	0.256944847	1.976373532	0.119295694	-0.20557426	1.221212256

Table A.7. Summary output of regression analysis for non-fatiguing in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.983748314
R Square	0.967760745
Adjusted R Square	0.954865043
Standard Error	0.153934571
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	3.556520739	1.778260369	75.04521693	0.000186622
Residual	5	0.118479261	0.023695852		
Total	7	3.675			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-0.88980452	0.564117471	-1.57733905	0.175544638	-2.33991464	0.560305604
OA	1.14381802	0.107288863	10.66110669	0.000125661	0.868023217	1.419612824
RoS	0.122647018	0.239847516	0.51135413	0.630876681	-0.49390064	0.739194684

Table A.8. Summary output of regression analysis for naturalness in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.277076421
R Square	0.076771343
Adjusted R Square	-0.29252011
Standard Error	0.518323778
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	0.111702304	0.055851152	0.207888216	0.818979802
Residual	5	1.343297696	0.268659539		
Total	7	1.455			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	5.693374695	1.761366925	3.232361533	0.023144949	1.165636871	10.22111252
OA	0.072014181	0.345051138	0.208705822	0.842913842	-0.81496800	0.958996368
IoS	0.113322624	0.17589995	0.644244776	0.547802552	-0.33884259	0.565487839

Table A.9. Summary output of regression analysis for pleasantness in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.955773337
R Square	0.913502672
Adjusted R Square	0.878903741
Standard Error	0.139117853
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	1.021981114	0.510990557	26.40262695	0.002200426
Residual	5	0.096768886	0.019353777		
Total	7	1.11875			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	2.180703067	0.55120486	3.956247897	0.010783184	0.763785867	3.597620266
OA	0.501869858	0.186390076	2.692578217	0.043165351	0.022738915	0.981000801
E_s	0.905759162	1.226674708	0.738385781	0.493458339	-2.24750855	4.059026884

Table A.10. Summary output of regression analysis for movability in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.724014171
R Square	0.52419652
Adjusted R Square	0.44489594
Standard Error	1.070649281
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	7.577260702	7.577260702	6.610248258	0.042275794
Residual	6	6.877739298	1.146289883		
Total	7	14.455			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-4.20981241	3.631090029	-1.15937979	0.290350831	-13.0947696	4.675144815
OA	1.707551708	0.664148167	2.571040307	0.042275794	0.082439686	3.332663729

Table A.11. Summary output of regression analysis for adjusted satisfaction in the Box game.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.983687316
R Square	0.967640736
Adjusted R Square	0.943371287
Standard Error	0.143909448
Observations	8

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	2.477160283	0.825720094	39.87073479	0.001942046
Residual	4	0.082839717	0.020709929		
Total	7	2.56			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	0.021613307	0.61501656	0.035142642	0.973649798	-1.68594641	1.729173025
OA	1.213140364	0.205354075	5.907554361	0.004109604	0.642986046	1.783294681
E_s	-2.91395976	1.269837397	-2.29475031	0.083419564	-6.43959359	0.611674058
E_y	0.335320489	0.182060979	1.841803175	0.139306832	-0.17016182	0.840802802

Appendix B: Experiment for the Slingshot Game

B.1. Experimental Instruction, Observation, and Questionnaire

B.1.1. Experiment

To compare three input methods -- 3D mouse (haptic device), dynamic arm gestures (Kinect depth camera), and static finger gestures (Kinect depth camera) -- in a 3D Slingshot game and on a large-scale display.

B.1.1.1. Training Session

- Fifteen minutes of practicing the tasks (five minutes on each device).
- Try the following four primitive tasks to get used to the application in order to run the test session precisely (press 'h' for Haptic device, and 'k' for Kinect device):
    a. Move the pointer (tool) around
        i. 3D Mouse: Move the mouse handle

   ii. Dynamic Gestures: Move your palm (should run 'wave' gesture to activate the pointer)
   iii. Static Gestures: Move your palm (should run 'wave' gesture to activate the pointer)

b. Grab the ball (object)
   i. 3D Mouse: Locate the pointer in the ball space and keep pressing the mouse button
   ii. Dynamic Gestures: Locate the pointer in the ball space and draw a circle with your palm around the ball
   iii. Static Gestures: Locate the pointer in the ball space and fist your palm

c. Pull the ball (object with the elastic band) and point towards the pin (target)
   i. 3D Mouse: Pull the mouse handle backward (still hold the mouse button pressed)
   ii. Dynamic Gestures: Pull your palm backward
   iii. Static Gestures: Pull your fist backward

d. Throw the ball (object) towards the pin (target)
   i. 3D Mouse: Release the pressed mouse button
   ii. Dynamic Gestures: Push your palm towards the screen
   iii. Static Gestures: Open your fist to a palm position

B.1.1.1.1. Learning curve

B.1.1.1.1.1. 3D mouse

The user is asked to try as many shots needed for 10 successful hits using the 3D mouse, while the experiment observer keeps a record of the time between each two consecutive hits and the total number of shots, until the user hits 10 times.

B.1.1.1.1.2. Dynamic gestures

The user is asked to try as many shots needed for 10 successful hits using the dynamic gestures, while the experiment observer keeps a record of the time between each two consecutive hits and the total number of shots, until the user hits 10 times.

B.1.1.1.1.3. Static gestures

The user is asked to try as many shots needed for 10 successful hits using the static gestures, while the experiment observer keeps a record of the time between each two consecutive hits and the total number of shots, until the user hits 10 times.

B.1.1.2. Test Session

B.1.1.2.1. 3D mouse

The user is asked to try 5 shots using the 3D mouse while the experiment observer keeps a record of the total time and the scores (number of hits out of 5) from the start to the end point.

B.1.1.2.1.1. Task

Run the steps “a-b-c-d (i)” as explained in training session:

1- Move the pointer around (“a (i)”): Move the mouse handle
2- Grab the ball (“b (i)”): Locate the pointer in the ball space and keep pressing the mouse button
3- Pull the ball and point towards the pin (“c (i)”): Pull the mouse handle backward (still hold the mouse button pressed)
4- Throw the ball towards the pin (“d (i)”): Release the pressed mouse button

B.1.1.2.1.2. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied), in the following table.

Table B.1. Questions for four primitive tasks.

Satisfaction criteria for primitive tasks			1	2	3	4	5	6	7
1	How easy was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							
2	How light (non-fatiguing) was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							
3	How natural was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							
4	How pleasant was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							
5	How flexible (mobility in the area) was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							
6	How overall satisfactory was to:	move the pointer?							
		grab the ball?							
		pull the ball?							
		throw the ball?							

B.1.1.2.2. Dynamic gestures

The user is asked to try 5 shots using the dynamic gestures while the experiment observer keeps a record of the total time and the scores (number of hits out of 5) from the start to the end point.

B.1.1.2.2.1. Task

Run the steps “a-b-c-d (ii)” as explained in training session:

1- Move the pointer around (“a (ii)”): Move your palm (should run ‘wave’ gesture to activate the pointer)
2- Grab the ball (“b (ii)”): Locate the pointer in the ball space and draw a circle with your palm around the ball
3- Pull the ball and point towards the pin (“c (ii)”): Pull your palm backward
4- Throw the ball towards the pin (“d (ii)”): Push your palm towards the screen

B.1.1.2.2.2. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) into the Table B.1.

B.1.1.2.3. Static gestures

The user is asked to try 5 shots using the static gestures while the experiment observer keeps a record of the total time and the scores (number of hits out of 5) from the start to the end point.

B.1.1.2.3.1. Task

Run the steps “a-b-c-d (iii)” as explained in training session:

1- Move the pointer around (“a (iii)”): Move your palm (should run ‘wave’ gesture to activate the pointer)
2- Grab the ball (“b (iii)”): Locate the pointer in the ball space and fist your palm
3- Pull the ball and point towards the pin (“c (iii)”): Pull your fist backward
4- Throw the ball towards the pin (“d (iii)”): Open your fist to a palm position

B.1.1.2.3.2. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) into the Table B.1.

B.1.2. Extra Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied), in the following table.

Table B.2. Extra questions for having a better feedback.

Q1: Your gaming skill's level?

1) Novice
2) Somehow experienced
3) Average experienced
4) Very experienced
5) Expert

Q2: When it comes to playing a 3D computer game, please rank the following options from 1 (the least important) to 7 (the most important). You may use the same rank if they have the same weight of importance to you.

1) Easy to interact
2) Non-fatiguing interaction
3) Natural interaction
4) Fun to use
5) Mobility

Q3: What combination of 3D mouse and gestures capabilities would you prefer to use on this application?

1) Just 3D mouse
2) Mostly 3D mouse
3) Equal use of 3D mouse and gestures
4) Mostly gestures
5) Just gestures

Table B.3. User comments.

Q4: What did you like/dislike about these gesture applications?

Like:

Dislike:

Q5: What did you like/dislike about this 3D mouse application?

Like:

Dislike:

Q6: Please share your opinion about mandatory features (must have) and desirable features (nice to have) to improve the gesture and 3D mouse applications?

Gestures:

3D Mouse:

B.2. Extra Results of Experiment

Table B.4. Performance data for interactions in the Slingshot game.

Performance of interactions (mixed-based)								
Feedback of task	**Modality**	**TT**	**SD**	**AV**	**NT**	**NA**	**AR**	**R**
Grab & Hit	Haptic	25.5	247.5	9.71	5	4.1	0.82	2
	Dynamic gesture	28.21	247.5	8.77	5	4	0.80	2
	Static posture	22.12	247.5	11.19	5	4.8	0.96	2

Appendix C: Experiment for the Race Game

C.1. Experimental Instruction, Observation, and Questionnaire

C.1.1. Experiment

To compare two input methods of touch-based (keyboard/mouse) and finger/hand gestures (webcam), performing various tasks in a 2D Race game on a desktop display. There are four testing sessions to evaluate the visibility in design, as follows.

1) **Touch-based with good visibility:** using mouse/keyboard in the same direction of display (close view of keyboard), with speedometer in City-road section, and complete all other sections.
2) **Touch-based with poor visibility:** using mouse/keyboard in a different direction from display (far view of keyboard), without speedometer in City-road section, and wave all other sections by starting and finishing actions.

3) **Touch-less with good visibility:** using gestures in the same direction of display (close view of gesture results), with speedometer in City-road section, and complete all other sections.
4) **Touch-less with poor visibility:** using gestures in a different direction from display (far view of gesture results), without speedometer in City-road section, and wave all other sections by starting and finishing actions.

C.1.1.1. Training Session

- Ten minutes of practicing the navigation (moving the pointer on the scene using keyboard/mouse and hand/fingers movements), to get used to the application by running the following tasks (using the desktop display):
  a. Set/Reset: Set the pointer (in the beginning), or leave and reconnect the selected object (when it is out of range or you need to rest, to locate the mouse or your hand in a proper position). This is also done at the end of each section.
     i. Mouse:
        1. replace the mouse to a proper position
        2. start/resume the task
     ii. Gesture (RH: sign, move):
        1. show the sign of ‘0’ (fist) in camera’s field of view (move it to a comfortable position out of camera’s field of view and rest, if your hand is fatigue or out of range)
        2. show the sign of ‘5’ (palm) in camera’s field of view and start/resume the task
  b. Activation: Move the pointer to collide with the object (green box) and select it. A red circle inside the green box shows that the object is selected.
     i. Keyboard/Mouse:
        1. press the ‘S’ key to start the process
        2. move the mouse in X/Y directions to locate the pointer on the object
        3. press the “Enter” key to select the object, and complete the process

4. press the ‘F’ key at any time to finish the process

ii. Gesture (RH: move, sign):

1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
2. move your hand in X/Y directions to locate the pointer on the object
3. show the sign of ‘1’ (index finger) to select the object, and complete the process
4. show the sign of “Open A” (thumb finger) at any time to finish the process

c. City-road: Move the object under a limited range of speed (a speedometer shows the object’s speed, which should not be more than 80) to the end line. The object will relocate at the middle of the road (the dotted line) to be moved again, if it gets off the road (hits the border). It will also stop, if its speed surpasses 80. Try to stay on the middle of the road (dotted line).

i. Mouse:

1. press the ‘S’ key to start the process
2. locate on the object and move the mouse in X/Y directions to carry the object to the end, and complete the process
3. press the ‘F’ key at any time to finish the process

ii. Gesture (RH: move):

1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
2. locate on the object and move your hand in X/Y directions to carry the object to the end, and complete the process
3. show the sign of “Open A” (thumb finger) at any time to finish the process

d. Road-break: Jump the object over the breakage.

i. Keyboard:

1. press the ‘S’ key to start the process

2. press the “Space” key to jump the object up, and complete the process
3. press the ‘F’ key at any time to finish the process

ii. Gesture (RH: sign):

1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
2. show the sign of ‘2’ (index and middle fingers) to jump the object up, and complete the process
3. show the sign of “Open A” (thumb finger) at any time to finish the process

e. Narrow-road: Resize the object to fit in the path while moving it to the end line (at the same time in parallel actions). The object will stop (and relocate at the middle line of the road) to be moved again, if it gets off the road (hits the border). Try to stay on the middle of the road (dotted line).

i. Keyboard/Mouse:

1. press the ‘S’ key to start the process
2. move the mouse left to carry the object, press the “Down” arrow-key twice to reduce the size of object to its smallest (all at the same time), and complete the process at the end
3. press the ‘F’ key at any time to finish the process

ii. Gesture (RH: move, sign):

1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
2. move your hand to the left to carry the object, show the sign of ‘3’ (thumb, index, and middle fingers) twice (show the palm of your hand in between) to reduce the size of object to its smallest (all at the same time), and complete the process at the end
3. show the sign of “Open A” (thumb finger) at any time to finish the process

f. Bridge: Resize the object to fit in the path, and then move it to the end line (one after the other in serial actions). The object will stop (and relocate at the middle line of the road) to be moved again, if it gets off the road (hits the border). Try to stay on the middle of the road (dotted line).

    i. Keyboard/Mouse:

        1. press the ‘S’ key to start the process
        2. press the “Down” arrow-key twice to reduce the size of object to its smallest
        3. press the “Tab” key to finish the resizing process and start the moving process
        4. move the mouse left to carry the object to the end, and complete the process
        5. press the ‘F’ key at any time to finish the process

    ii. Gesture (RH: move, sign):

        1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
        2. show the sign of ‘3’ (thumb, index, and middle fingers) twice (show the palm of your hand in between) to reduce the size of object to its smallest
        3. show the sign of ‘Open I’ (pinky finger) to finish the resizing process and start the moving process
        4. move your hand to the left and carry the object to the end, and complete the process
        5. show the sign of “Open A” (thumb finger) at any time to finish the process

g. Cruise-control: Rotate the object on the turns of the road, while it moves towards the end line automatically at a fixed speed (to the left at the first straight/turn, upwards at the second straight/turn). The object will relocate (at the middle line of the turns) to be rotated again, if it gets off the road (hits the border). Try to stay on the middle of the road (dotted line).

    i. Keyboard:

1. press the ‘S’ key to start the process
2. at the first straight: do nothing
3. at the first turn:
    a. press and hold the “Ctrl” key to start the rotation process
    b. press the “Up” arrow-key (as needed) to direct the object upward
    c. release the “Ctrl” key to finish the rotation process
4. at the second straight: do nothing
5. at the second turn:
    a. press and hold the “Ctrl” key to start the rotation process
    b. press the “Right” arrow-key (as needed) to direct the object to the right
    c. release the “Ctrl” key to finish the rotation process
6. press the ‘F’ key at any time to finish the process

ii. Gesture (RH: sign, rotate):

1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
2. at the first straight: do nothing
3. at the first turn:
    a. show the sign of ‘7’ (index, middle, and pinky fingers) to start the rotation process
    b. rotate your hand clockwise (as needed) to turn the object upward around the corner
    c. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to finish the rotation process
4. at the second straight: do nothing
5. at the second turn:

a. show the sign of ‘7’ (index, middle, and pinky fingers) to start the rotation process
b. rotate your hand clockwise (as needed) to turn the object rightward around the corner
c. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to finish the rotation process

6. show the sign of “Open A” (thumb finger) at any time to finish the process

h. Free-way: Guide the shot object and control its direction on the road, while it moves towards the end line automatically at a fixed acceleration (to the right). Try to stay on the middle of the road (dotted line).
   i. Keyboard:
      1. press the ‘S’ key to start the process
      2. press the “Down/Up” arrow-keys (as needed) to direct the object downward/upward respectively
      3. press the ‘F’ key at any time to finish the process
   ii. Gesture (RH: move):
      1. show the sign of ‘5’ (palm), straight up and wide open (with middle finger pointed at 12 o’clock), to start the process
      2. show the sign of ‘3’ (thumb, index, and middle fingers) or ‘4’ (index, middle, ring, and pinky fingers), as needed (show the palm of your hand for a neutral control), to direct the object downward/upward respectively
      3. show the sign of “Open A” (thumb finger) at any time to finish the process

i. Deactivation: Release/Deselect the object by breaking the connection between the pointer and the object. The object (green box) will relocate to the initial position at the starting point of game.
   i. Keyboard:
      1. press the ‘S’ key to start the process

2. press the "Del" key to deselect the object
3. press the 'F' key at any time to finish the process

ii. Gesture (RH: sign):

1. show the sign of '5' (palm), straight up and wide open (with middle finger pointed at 12 o'clock), to start the process
2. show the sign of '9' (middle, ring, and pinky fingers) to deselect the object
3. show the sign of "Open A" (thumb finger) at any time to finish the process

C.1.1.2. Test Session

Layout: For the purpose of generality, a Race game (Figure 1) is designed with a vast range of tasks under some physics engine involved to cover the common actions in a game design. The UI is with various terrains (actions) including: activation (select the object), city-road (move under a limited speed), road-break (jump/switch to the other side), narrow-road (resize to fit and move, at the same time in parallel actions), bridge (resize to fit, then move in serial actions), cruise-control (rotate on the road turns with a fixed speed), free-way (shoot and control the direction with an acceleration), deactivation (deselect the object), and reset (reconnect the object when out-of-range).

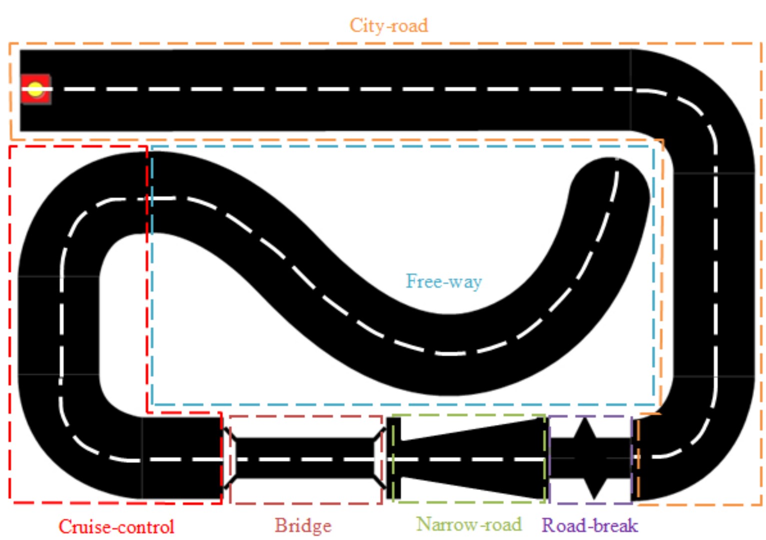


Figure C.1. Race game UI and its sections.

C.1.1.2.1. Good visibility

The user is asked to perform different tasks using the touch-less (finger/hand gestures) and touch-based (keyboard/mouse) inputs with a good visibility of feedback. As shown in Table C.1, for each type of input, the experiment observer (the program) keeps a record of time (TT, the total time in a task), effective distance (SD, the shortest distance traveled in a task), entire distance (RM, the real motion traveled in a task), number of actions expecting ideally (NA, the number of move/action to do a task), number of trials including errors (NT, the number of move/action to do a task, including the speed/border/select errors), and speedometer (V_t, the object's velocity at every second).

Table C.1. Observations on different tasks (X means not applicable).

Task		**TT**	**SD**	**RM**	**NA**	**NT**	**V_t**	**Type**
Activation:	*Select*		X	X			X	Action
City-road:	*Limited speed*							Mixed
Road-break:	*Jump*		X	X			X	Action
Narrow-road:	*Resize + Move*						X	Mixed
Bridge:	*Resize*		X	X			X	Action
	Move						X	Mixed
Cruise-control:	*Rotate*						X	Mixed
Free-way:	*Shoot*				X	X	X	Distance
Deactivation:	*Deselect*		X	X			X	Action

C.1.1.2.1.1. Touch-based input

Please run the steps "A-B-C-D-E-F-G-H-I (i)" as explained in training session (Section C.1.1.1). Table C.2 shows the summary.

Table C.2. Design for mouse/keyboard.

Activity	Task	Action					
A) Set/Reset	*Connect pointer*	Move					
B) Activation	*Select*	[S] Start	Move	[Enter] Select			
C) City-road	*Limited speed*	[S] Start	Drag				
D) Road-break	*Jump*	[S] Start	Move	[Space bar] Jump			
E) Narrow-road	*Resize + Move*	[S] Start	Move	[⌄] [⌄] ← Resize & Drag			
F) Bridge	*Resize*	[S] Start	Move	[⌄] [⌄] Resize		[Tab] Next	
	Move						← Drag
G) Cruise-control	*Rotate*	[S] Start	Move	[Ctrl] [^] Rotate upward		[Ctrl] [>] Rotate rightward	
H) Free-way	*Shoot*	[S] Start	Move	[⌄] Shoot downward		[^] Shoot upward	
I) Deactivation	*Deselect*	[S] Start	Move	[Del] Deselect			

C.1.1.2.1.1.1. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table C.3.

Table C.3. Questions for satisfaction of tasks.

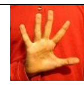
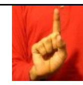
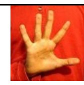
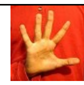
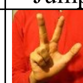
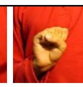
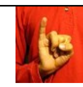
Satisfaction criteria for different tasks			1	2	3	4	5	6	7
A	Set/Reset (pointer control)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
B	Activation (select)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
C	City-road (limited speed)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
D	Road-break (jump)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
E	Narrow-road (parallel resize and move)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
F	Bridge (serial resize and move)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

G	Cruise-control (rotate)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
H	Free-way (shoot)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
I	Deactivation (deselect)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

C.1.1.2.1.2. Touch-less input

Please run the steps “A-B-C-D-E-F-G-H-I (ii)” as explained in training session (Section C.1.1.1). Table C.4 shows the summary.

Table C.4. Design for hand gestures.

Activity	Task	Action						
A) Set/Reset	*Connect pointer*	 Neutral	Move	 Start				
B) Activation	*Select*	 Start	Move	 Select				
C) City-road	*Limited speed*	 Start	Drag					
D) Road-break	*Jump*	 Start	Move	 Jump				
E) Narrow-road	*Resize + Move*	 Start	Move	   Resize & Drag				
F) Bridge	*Resize*	 Start	Move	   Resize			 Next	
	Move							Drag
G) Cruise-control	*Rotate*	 Start	Move	 Rotate upward		 Neutral	 Rotate rightward	
H) Free-way	*Shoot*	 Start	Move	 Shoot downward		 Shoot upward		
I) Deactivation	*Deselect*	 Start	Move	 Deselect				

C.1.1.2.1.2.1. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table C.5.

Table C.5. Questions for satisfaction of tasks.

Satisfaction criteria for different tasks			1	2	3	4	5	6	7
A	Set/Reset (pointer control)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
B	Activation (select)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
C	City-road (limited speed)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
D	Road-break (jump)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
E	Narrow-road (parallel resize and move)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
F	Bridge (serial resize and move)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

G	Cruise-control (rotate)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
H	Free-way (shoot)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
I	Deactivation (deselect)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

C.1.1.2.2. Poor visibility

The user is asked to perform different tasks using the touch-less (finger/hand gestures) and touch-based (keyboard/mouse) inputs with a poor visibility of feedback. As shown in Table C.6, for each type of input, the experiment observer (the program) keeps a record of time (TT, the total time in a task), effective distance (SD, the shortest distance traveled in a task), entire distance (RM, the real motion traveled in a task), number of actions expecting ideally (NA, the number of move/action to do a task), number of trials including errors (NT, the number of move/action to do a task, including the speed/border/select errors), and speedometer (V_t, the object's velocity at every second).

Table C.6. Observations on different tasks (X means not applicable).

Task		TT	SD	RM	NA	NT	V_t	Type
Activation:	*Select*		X	X			X	Action
City-road:	*Limited speed*							Mixed
Road-break:	*Start and finish*		X	X			X	Action
Narrow-road:	*Start and finish*		X	X			X	Action
Bridge:	*Start and finish*		X	X			X	Action
Cruise-control:	*Start and finish*		X	X			X	Action
Free-way:	*Start and finish*		X	X			X	Action
Deactivation:	*Start and finish*		X	X			X	Action

C.1.1.2.2.1. Touch-based input

Please run the steps "A-B-C (i)" as explained in training session (Section C.1.1.1), and "D-E-F-G-H-I (only starting and finishing actions using keyboard)". Table C.7 shows a summary.

Table C.7. Design for mouse/keyboard.

Activity	Task	Action			
A) Set/Reset:	*Connect pointer*	Move			
B) Activation:	*Select*	s Start	Move	Enter Select	
C) City-road:	*Limited speed*	s Start	Drag		
D) Road-break:	*Start and Finish*	s Start	f Finish		
E) Narrow-road:	*Start and Finish*	s Start	f Finish		
F) Bridge:	*Start and Finish*	s Start	f Finish		
G) Cruise-control:	*Start and Finish*	s Start	f Finish		
H) Free-way:	*Start and Finish*	s Start	f Finish		
I) Deactivation:	*Start and Finish*	s Start	f Finish		

C.1.1.2.2.1.1. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table C.8.

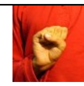
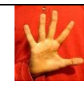
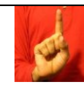
Table C.8. Questions for satisfaction of tasks.

Satisfaction criteria for different tasks			1	2	3	4	5	6	7
A	Set/Reset (pointer control)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
B	Activation (select)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
C	City-road (limited speed)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
D	Start (open)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
E	Finish (close)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

C.1.1.2.2.2. Touch-less input

Please run the steps “A-B-C (ii)” as explained in training session (Section C.1.1.1), and “D-E-F-G-H-I (only starting and finishing actions using gestures)”. Table C.9 shows a summary.

Table C.9. Design for hand gestures.

Activity	Task	Action		
A) Set/Reset:	*Connect pointer*	 Neutral	Move	 Start
B) Activation:	*Select*	 Start	Move	 Select
C) City-road:	*Limited speed*	 Start	Drag	
D) Road-break:	*Start and Finish*	 Start	 Finish	
E) Narrow-road:	*Start and Finish*	 Start	 Finish	
F) Bridge:	*Start and Finish*	 Start	 Finish	
G) Cruise-control:	*Start and Finish*	 Start	 Finish	
H) Free-way:	*Start and Finish*	 Start	 Finish	
I) Deactivation:	*Start and Finish*	 Start	 Finish	

C.1.1.2.2.2.1. Questions

Please rate your satisfaction on a scale of 1 to 7 (1 for absolutely unsatisfied, and 7 for extremely satisfied) in Table C.10.

Table C.10. Questions for satisfaction of tasks.

Satisfaction criteria for different tasks			1	2	3	4	5	6	7
A	Set/Reset (pointer control)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
B	Activation (select)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
C	City-road (limited speed)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
D	Start (open)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							
E	Finish (close)	How easy was?							
		How light (non-fatiguing) was?							
		How natural was?							
		How pleasant was?							
		How flexible (mobility in the area) was?							
		How overall satisfactory was?							

C.1.2. Extra Questions

Please answer the extra questions in Tables C.11 and C.12.

Table C.11. Extra questions for having a better feedback.

Q1: Your gaming skill's level?

1) Novice
2) Somehow experienced
3) Average experienced
4) Very experienced
5) Expert

Q2: When it comes to playing a computer game, please rank the following options from 1 (the <u>least</u> important) to 7 (the <u>most</u> important). You may use the same rank if they have the same weight of importance to you.

1) Easy to interact
2) Non-fatiguing interaction
3) Natural interaction
4) Fun to use
5) Mobility

Q3: What combination of <u>keyboard/mouse</u> and <u>gestures</u> capabilities would you prefer to use on this application?

1) Just keyboard/mouse
2) Mostly keyboard/mouse
3) Equal use of keyboard/mouse and gestures
4) Mostly gestures
5) Just gestures

Table C.12. User comments.

Q4: What did you like/dislike about these gesture applications?

Like:

Dislike:

Q5: What did you like/dislike about these keyboard/mouse applications?

Like:

Dislike:

Q6: Please share your opinion about mandatory features (must have) and desirable features (nice to have) that are missing in design, to improve the gesture and keyboard/mouse applications?

Gesture:
Mandatory:

Desirable:

Keyboard/mouse:
Mandatory:

Desirable:

C.2. Extra Results of Experiment

Table C.13. Performance data for interactions in good visibility of the Race game; part 1 (touch-based).

Performance of interactions (action-based, distance-based, and mixed)												
Visibility	**Modality**			**Activity/Task**	**TT**	**SD**	**RM**	**AV (AF)**	**NT**	**NA**	**AR (SR)**	**R**
Good	Touch-based	Move_Button		Activation (Action)	4	0	0	0.3	1.1	1	0.9	9
		Move	City-road	Horizontal drag (Mixed)	11	242	256	22.6	1.8	1	0.6	9
				First curve (Mixed)	2	55	61	26.2	1.3	1	0.8	9
				Vertical drag (Mixed)	4.1	93	101	22.7	1.6	1	0.6	9
				Second curve (Mixed)	1.7	55	59	32.4	1.2	1	0.8	9
				Total (Mixed)	18.6	445	477	23.9	5.9	4	0.7	9
		Move_Button		Road-break (Action)	3.1	0	0	0.3	1.1	1	0.9	9
		Move_Buttons		Narrow-road (Mixed)	14	55	62	3.9	3.5	3	0.9	9
		Move_Buttons	Bridge	Resize (Action)	10.6	0	0	0.2	2.3	2	0.9	9
				Horizontal drag (Mixed)	3.1	55	58	17.7	1.3	1	0.8	9
				Total (Mixed)	13.7	55	58	4.0	3.6	3	0.8	9
		Move_Buttons	Cruise-control	Horizontal move (Mixed)	0.4	21	21	52.5	1.2	1	0.9	9
				First rotate (Mixed)	7.1	55	176	7.7	3.4	1	0.3	9
				Vertical move (Mixed)	0.9	45	45	50.0	1.2	1	0.8	9
				Second rotate (Mixed)	7.4	55	188	7.4	3.8	1	0.3	9
				Total (Mixed)	15.8	176	430	11.1	9.6	4	0.4	9
		Move_Buttons		Free-way (Distance)	3.4	191	221	56.2	0.0	0	0.9	9
		Move_Button		Deactivation (Action)	3.4	0	0	0.3	1.1	1	0.9	9

Table C.14. Performance data for interactions in good visibility of the Race game; part 2 (touch-less).

Performance of interactions (action-based, distance-based, and mixed)												
Visibility	**Modality**		**Activity/Task**		**TT**	**SD**	**RM**	**AV (AF)**	**NT**	**NA**	**AR (SR)**	**R**
Good	Touch-less	Move_Sign	Activation (Action)		6.8	0	0	0.1	1.4	1	0.7	9
		Move	City-road	Horizontal drag (Mixed)	13.2	242	261	18.3	2.0	1	0.5	9
				First curve (Mixed)	3.6	55	62	15.3	1.5	1	0.7	9
				Vertical drag (Mixed)	6.2	93	109	15.0	2.2	1	0.5	9
				Second curve (Mixed)	3.4	55	59	16.2	1.4	1	0.7	9
				Total (Mixed)	26.4	445	491	16.9	7.0	4	0.6	9
		Move_Sign	Road-break (Action)		5.3	0	0	0.2	1.4	1	0.7	9
		Move_Signs	Narrow-road (Mixed)		13.5	55	66	4.1	3.8	3	0.8	9
		Move_Signs	Bridge	Resize (Action)	10.1	0	0	0.2	2.8	2	0.7	9
				Horizontal drag (Mixed)	6.1	55	59	9.0	1.5	1	0.7	9
				Total (Mixed)	16.2	55	59	3.4	4.2	3	0.7	9
		Move_Signs_Rotate	Cruise-control	Horizontal move (Mixed)	0.4	21	21	52.5	1.4	1	0.7	9
				First rotate (Mixed)	3.4	55	84	16.2	2.3	1	0.4	9
				Vertical move (Mixed)	0.9	45	45	50.0	1.5	1	0.7	9
				Second rotate (Mixed)	3.6	55	88	15.3	2.7	1	0.4	9
				Total (Mixed)	8.3	176	238	21.2	7.8	4	0.5	9
		Move_Signs	Free-way (Distance)		1.2	63	78	52.5	0.0	0	0.8	9
		Move_Sign	Deactivation (Action)		5.6	0	0	0.2	1.4	1	0.7	9

Table C.15. Performance data for interactions in poor visibility of the Race game; part 1 (touch-based).

Performance of interactions (action-based and mixed)												
Visibility	**Modality**		**Activity/Task**		**TT**	**SD**	**RM**	**AV (AF)**	**NT**	**NA**	**AR (SR)**	**R**
Poor	Touch-based	Move_Button	Activation (Action)		7.2	0	0	0.1	1.7	1	0.6	9
		Move	City-road	Horizontal drag (Mixed)	13.6	242	271	17.8	2.4	1	0.4	9
				First curve (Mixed)	2.2	55	60	25.0	1.3	1	0.8	9
				Vertical drag (Mixed)	5.1	93	109	18.2	1.8	1	0.6	9
				Second curve (Mixed)	1.6	55	61	34.4	1.3	1	0.8	9
				Total (Mixed)	22.5	445	501	19.8	6.7	4	0.6	9
		Buttons	Start and Finish	Road-break (Action)	10.2	0	0	0.2	4.3	2	0.5	9
				Narrow-road (Action)	8.3	0	0	0.2	3.6	2	0.6	9
				Bridge (Action)	7.6	0	0	0.3	2.8	2	0.7	9
				Cruise control (Action)	7.9	0	0	0.3	2.8	2	0.7	9
				Free-way (Action)	6.1	0	0	0.3	2.5	2	0.8	9
				Deactivation (Action)	4.4	0	0	0.5	2.4	2	0.9	9
				Total (Action)	44.5	0	0	0.3	18.2	12	0.7	9

Table C.16. Performance data for interactions in poor visibility of the Race game; part 2 (touch-less).

Performance of interactions (action-based and mixed)													
Visibility	**Modality**		**Activity/Task**		**TT**	**SD**	**RM**	**AV (AF)**	**NT**	**NA**	**AR (SR)**	**R**
Poor	Touch-based	Move_Sign	Activation (Action)		7.9	0	0	0.1	1.6	1	0.6	9
		Move	City-road	Horizontal drag (Mixed)	16.4	242	261	14.8	2.5	1	0.4	9
				First curve (Mixed)	3.9	55	64	14.1	1.6	1	0.6	9
				Vertical drag (Mixed)	7.6	93	119	12.2	2.3	1	0.4	9
				Second curve (Mixed)	4	55	63	13.8	1.5	1	0.7	9
				Total (Mixed)	31.9	445	507	13.9	7.8	4	0.5	9
		Signs	Start and Finish	Road-break (Action)	9.1	0	0	0.2	4.7	2	0.4	9
				Narrow-road (Action)	8.4	0	0	0.2	3.7	2	0.5	9
				Bridge (Action)	6.7	0	0	0.3	2.8	2	0.7	9
				Cruise control (Action)	5.4	0	0	0.4	2.5	2	0.8	9
				Free-way (Action)	5.5	0	0	0.4	2.6	2	0.8	9
				Deactivation (Action)	4.7	0	0	0.4	2.5	2	0.8	9
				Total (Action)	39.8	0	0	0.3	18.7	12	0.6	9

Table C.17. Summary output of regression statistical analysis for easiness in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.992829382
R Square	0.985710183
Adjusted R Square	0.978565274
Standard Error	0.205757054
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	17.52198421	5.840661403	137.9598021	6.34874E-06
Residual	6	0.254015792	0.042335965		
Total	9	17.776			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-2.08409924	0.750122647	-2.77834465	0.032067021	-3.91958323	-0.24861524
OA	0.692014071	0.087204897	7.935495553	0.000212807	0.478631375	0.905396766
E_y	1.079223112	1.082696586	0.99679183	0.357351738	-1.57003999	3.72848622
RoS	2.599880248	0.64877845	4.007346805	0.007059679	1.01237657	4.187383926

Table C.18. Summary output of regression statistical analysis for non-fatiguing in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.995340886
R Square	0.990703479
Adjusted R Square	0.988047331
Standard Error	0.137828791
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	14.17102257	7.085511285	372.9849436	7.74679E-08
Residual	7	0.132977429	0.018996776		
Total	9	14.304			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-1.27530747	0.310735352	-4.10415958	0.004549182	-2.01007982	-0.54053512
OA	0.542505744	0.04532089	11.97032424	6.46484E-06	0.435338869	0.649672619
RoS	2.868036221	0.300804069	9.534565914	2.92661E-05	2.156747625	3.579324817

Table C.19. Summary output of regression statistical analysis for naturalness in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.986870922
R Square	0.973914217
Adjusted R Square	0.966461136
Standard Error	0.235835995
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	14.53566969	7.267834843	130.6727012	2.86691E-06
Residual	7	0.389330315	0.055618616		
Total	9	14.925			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	0.357392891	0.331867422	1.076914657	0.317226558	-0.42734886	1.142134645
OA	1.214218138	0.077547637	15.65770641	1.04865E-06	1.030847114	1.397589161
IoS	1.013761995	0.128674906	7.878474739	0.000100495	0.709494192	1.318029798

Table C.20. Summary output of regression statistical analysis for pleasantness in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.987986377
R Square	0.97611708
Adjusted R Square	0.969293389
Standard Error	0.27061385
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	20.95137701	10.47568851	143.0482463	2.10526E-06
Residual	7	0.512622989	0.073231856		
Total	9	21.464			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-1.71371269	0.533010096	-3.21515991	0.014752091	-2.97408129	-0.45334409
OA	0.984052575	0.087863689	11.19976394	1.00902E-05	0.776287965	1.191817185
E_s	1.660139661	0.982234428	1.690166435	0.134841991	-0.66247568	3.982755011

Table C.21. Summary output of regression statistical analysis for movability in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.74074181
R Square	0.548698429
Adjusted R Square	0.492285732
Standard Error	0.863452486
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	7.251598434	7.251598434	9.726505975	0.01425593
Residual	8	5.964401566	0.745550196		
Total	9	13.216			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	1.672464673	1.169804585	1.429695776	0.190671829	-1.02510953	4.370038882
OA	0.641507292	0.205694732	3.118734675	0.01425593	0.16717439	1.115840194

Table C.22. Summary output of regression statistical analysis for adjusted satisfaction in the Race game with good and poor visibilities (action-based).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.98108217
R Square	0.962522224
Adjusted R Square	0.943783336
Standard Error	0.291675421
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	13.10955269	4.369850897	51.3649596	0.000113521
Residual	6	0.510447309	0.085074551		
Total	9	13.62			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	0.789859742	0.603182189	1.309487841	0.238275185	-0.68607390	2.265793388
OA	0.883055345	0.115513154	7.644630201	0.000261583	0.60040484	1.16570585
E_s	-0.33225928	1.636302565	-0.20305492	0.845803123	-4.33614742	3.671628851
E_y	-0.13912054	1.641922255	-0.08473028	0.935232277	-4.15675957	3.878518478

Table C.23. Summary output of regression statistical analysis for easiness in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.811026486
R Square	0.65776396
Adjusted R Square	0.48664594
Standard Error	0.308542798
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	1.09780805	0.365936017	3.84391989	0.075525899
Residual	6	0.57119195	0.095198658		
Total	9	1.669			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	2.846982625	1.735230704	1.640694012	0.151970642	-1.39897394	7.092939199
OA	0.49947048	0.207078509	2.411986069	0.052431912	-0.00723237	1.006173339
E_y	0.21171815	0.129481793	1.635119073	0.153141212	-0.10511238	0.528548684
RoS	-0.13659111	0.419998269	-0.32521828	0.756057241	-1.16428985	0.891107626

Table C.24. Summary output of regression statistical analysis for non-fatiguing in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.878346845
R Square	0.771493181
Adjusted R Square	0.706205518
Standard Error	0.227752163
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	1.225902664	0.612951332	11.81682956	0.005703574
Residual	7	0.363097336	0.051871048		
Total	9	1.589			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	1.154622917	1.228919852	0.939542896	0.378732494	-1.75131076	4.060556602
OA	0.697969735	0.149275548	4.675713777	0.002272215	0.344989155	1.050950315
RoS	0.284057505	0.271366789	1.046765915	0.330000348	-0.35762298	0.925737996

Table C.25. Summary output of regression statistical analysis for naturalness in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.909305853
R Square	0.826837134
Adjusted R Square	0.77736203
Standard Error	0.2041636
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	1.393220571	0.696610286	16.7121857	0.002160686
Residual	7	0.291779429	0.041682776		
Total	9	1.685			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	1.661807489	0.762728564	2.178766559	0.065760354	-0.14175897	3.465373948
OA	0.675653228	0.118074834	5.722245843	0.000718654	0.396450612	0.954855844
IoS	-0.06448325	0.053016881	-1.21627777	0.263296064	-0.18984825	0.060881748

Table C.26. Summary output of regression statistical analysis for pleasantness in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.934345106
R Square	0.873000776
Adjusted R Square	0.836715284
Standard Error	0.197455721
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	1.876078668	0.938039334	24.05922358	0.000729968
Residual	7	0.272921332	0.038988762		
Total	9	2.149			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	1.553750925	0.698527212	2.224324118	0.061483666	-0.09800346	3.205505311
OA	0.641923551	0.14708404	4.364331784	0.003296738	0.294125063	0.989722039
E_s	0.760887793	0.537432908	1.415781917	0.199758586	-0.50993909	2.031714681

Table C.27. Summary output of regression statistical analysis for movability in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.198761818
R Square	0.03950626
Adjusted R Square	-0.08055545
Standard Error	0.307780271
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	0.031170439	0.031170439	0.329049602	0.581982179
Residual	8	0.757829561	0.094728695		
Total	9	0.789			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	7.206924101	1.045152719	6.895570348	0.000125086	4.796797608	9.617050594
OA	-0.10186418	0.177578676	-0.57362845	0.581982179	-0.51136134	0.30763298

Table C.28. Summary output of regression statistical analysis for adjusted satisfaction in the Race game with good and poor visibilities (mixed).

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.88717989
R Square	0.787088158
Adjusted R Square	0.680632237
Standard Error	0.180683376
Observations	10

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	3	0.724121105	0.241373702	7.393559228	0.019348203
Residual	6	0.195878895	0.032646482		
Total	9	0.92			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	3.091831111	0.639211866	4.836942607	0.002889762	1.52773602	4.655926203
OA	0.529971963	0.151791553	3.491445681	0.012960527	0.158551412	0.901392514
E_s	-0.24523382	0.659215756	-0.37200844	0.722671417	-1.85827667	1.367809018
E_y	-0.02798884	0.088966895	-0.31459840	0.76371911	-0.24568299	0.189705306

C.3. Extra Details of Design in the Race Game

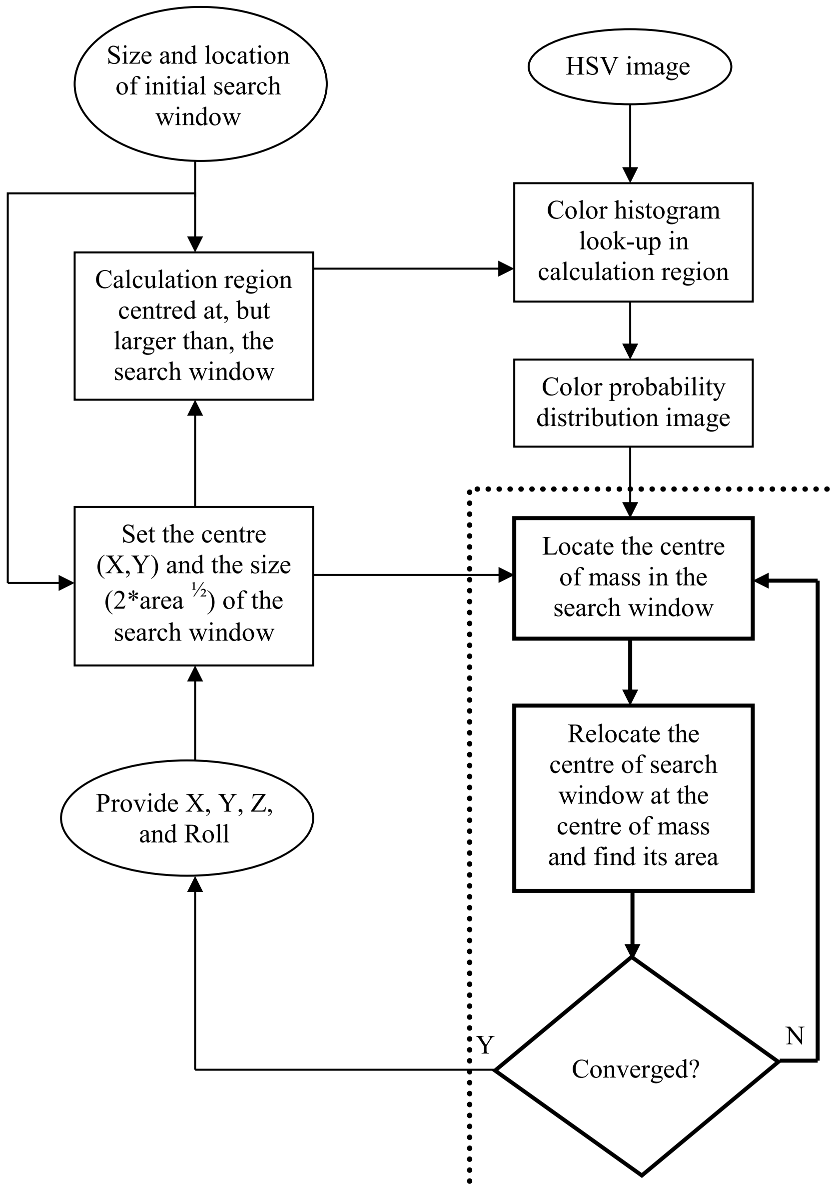


Figure C.2. Block diagram of CAMShift algorithm.

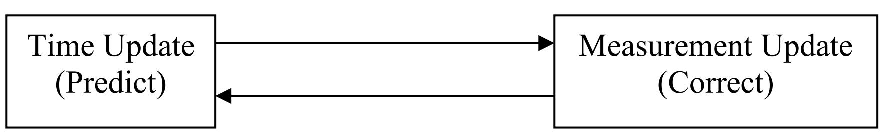


Figure C.3. Ongoing discrete Kalman filter's cycle.

C.3.1. Wrist Detection and Hand Cropping in Race Game

According to the following histogram generating functions [114], the wrist end is detected from the binary silhouette matrix of the binary image (*imb)*, with m rows and n columns.

$$H_x = \sum_{y=n}^{1} imb(x,y) \quad , \quad H_y = \sum_{x=m}^{1} imb(x,y) \tag{C.1}$$

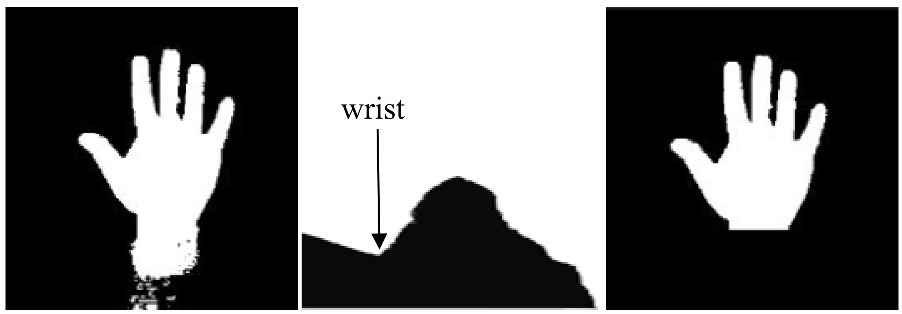


Figure C.4. Hand cropping; Left to right: (a) initial image, (b) inclination in hand-arm histogram where wrist end is located on the valley (since wrist is the narrowest part of a hand), and (c) cropped hand image respectively.

As Figure C.4 shows, the valley in H_y provides the possible height of the wrist location where the width of object in the region of interest begins to increase (from bottom to up) in an appropriate height value amount of height. From this end, when the height of the wrist is detected, it can find the width of the wrist, and consequently the center of the wrist is found. Having the four sides of a rectangle around the object (from up, left, and right limited to the hand's surrounding box comes from the CAMShift process; and from bottom to the wrist line calculated in this section), the hand image is cut off to a correct segmentation for further processes.

C.3.2. Extraction of Palm Center and Hand Roll in Race Game

The center of gravity (COG) of the user's hand, obtained by color histogram projection, is given as first moments [109,113] of the probability distribution image during the course of hand tracking algorithm operation where *(x, y)* range over the search window, and *I(x, y)* is the pixel (probability) value at *(x, y)*.

$$\bar{x} = \frac{M_{10}}{A} = \frac{M_{10}}{M_{00}} \quad , \quad \bar{y} = \frac{M_{01}}{A} = \frac{M_{01}}{M_{00}} \tag{C.2}$$

The spatial moment of zero order (M_{00}) is the area (*A*) of the object. The central moments of order up to 2 are:

$$\mu_{00} = M_{00} \tag{C.3}$$

$$\mu_{01} = 0 \quad , \quad \mu_{10} = 0 \tag{C.4}$$

$$\bar{x} = \frac{M_{10}}{A} = \frac{M_{10}}{M_{00}} \quad , \quad \bar{y} = \frac{M_{01}}{A} = \frac{M_{01}}{M_{00}} \tag{C.5}$$

$$\mu_{11} = M_{11} - \bar{x}M_{01} = M_{11} - \bar{y}M_{10} \tag{C.6}$$

$$\mu_{20} = M_{20} - \bar{x}M_{10} \quad , \quad \mu_{02} = M_{02} - \bar{y}M_{01} \tag{C.7}$$

The second order central moments construct a covariance matrix with its eigenvectors corresponding to the major and minor axes of the image intensity. The angle of the eigenvector associated with the largest eigenvalue provides the image *orientation*, Θ (as shown in Figure C.5). The eigenvalues (λ_i) of the covariance matrix are proportional to the squared length of the eigenvector axes [96].

$$\mu'_{20} = \mu_{20}/\mu_{00} = M_{20}/M_{00} - \bar{x}^2 \tag{C.8}$$

$$\mu'_{02} = \mu_{02}/\mu_{00} = M_{02}/M_{00} - \bar{y}^2 \tag{C.9}$$

$$\mu'_{11} = \mu_{11}/\mu_{00} = M_{11}/M_{00} - \bar{x}\bar{y} \tag{C.10}$$

$$COV[I(x,y)] = \begin{bmatrix} \mu'_{20} & \mu'_{11} \\ \mu'_{11} & \mu'_{02} \end{bmatrix} \tag{C.11}$$

$$\Theta = \frac{1}{2}\arctan\left(\frac{2\mu'_{11}}{\mu'_{20} - \mu'_{02}}\right) \tag{C.12}$$

$$\lambda_i = \frac{\mu'_{20} + \mu'_{02}}{2} \pm \frac{\sqrt{4(\mu'_{11})^2 + (\mu'_{20} - \mu'_{02})^2}}{2} \tag{C.13}$$

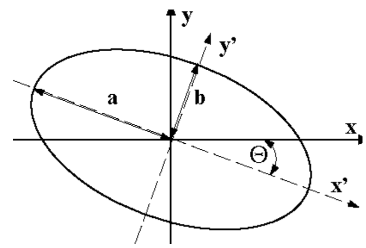


Figure C.5. Image ellipsis.

Moments η_{ij}, where $i + j \geq 2$, is invariant to both *translation* and *scale* if the corresponding central moment is divided by the properly scaled *(00)*th moment, using the following equation:

$$\eta_{ij} = \frac{\mu'_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \tag{C.14}$$

When used in hand tracking, the above equations give us the hand roll (as shown in Table C.29), length and width [113].

Table C.29. Tilt angle $\boldsymbol{\Theta}$ (orientation) of an image ellipsis.

$\boldsymbol{\mu'_{20} - \mu'_{02}}$	$\boldsymbol{\mu'_{11}}$	$\boldsymbol{\Theta}$
Zero	Zero	0°
Zero	Positive	+45°
Zero	Negative	−45°
Positive	Zero	0°
Negative	Zero	−90°
Positive	Positive	$\frac{1}{2}\arctan\left(\frac{2\eta_{11}}{\eta_{20}-\eta_{02}}\right)$, $0° < \Theta < 45°$
Positive	Negative	$\frac{1}{2}\arctan\left(\frac{2\eta_{11}}{\eta_{20}-\eta_{02}}\right)$, $-45° < \Theta < 0°$
Negative	Positive	$\frac{1}{2}\arctan\left(\frac{2\eta_{11}}{\eta_{20}-\eta_{02}}\right) + 90°$, $45° < \Theta < 90°$
Negative	Negative	$\frac{1}{2}\arctan\left(\frac{2\eta_{11}}{\eta_{20}-\eta_{02}}\right) - 90°$, $-90° < \Theta < -45°$

It is possible to calculate moments which are invariant under *translation*, changes in *scale*, and also *rotation*. Most frequently used are the Hu set of invariant moments (there are eight Hu sets, among which only the first one is the target for this concept):

$$I_1 = \eta_{20} + \eta_{02} \tag{C.15}$$

The first Hu set of invariant moment, I_1, presents the moment of inertia around the image's centroid, where the pixels' intensities presents the physical density [99].