OOH, WHAT'S THIS BUTTON DO?

OOH, WHAT'S THIS BUTTON DO? PHYSICAL REQUIREMENTS OF VIDEO GAMING

BY
SASHA M. SORAINE, B.Eng.

A THESIS
SUBMITTED TO THE DEPARTMENT OF COMPUTING AND SOFTWARE
AND THE SCHOOL OF GRADUATE STUDIES
OF MCMASTER UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE

Master of Applied Science (2018)
(Computing and Software)

McMaster University
Hamilton, Ontario, Canada

TITLE:	Ooh, what's this button do? Physical requirements of video gaming
AUTHOR:	Sasha M. Soraine B.Eng., (Software Engineering) McMaster University, Hamilton, Canada
SUPERVISOR:	Dr. Jacques Carette
NUMBER OF PAGES:	xix, 352

Lay Abstract

A lay abstract of not more 150 words must be included explaining the key goals and contributions of the thesis in lay terms that is accessible to the general public.

Abstract

Video games as a system are composed of two component systems: the player and the game. The interaction between these two create specific gameplay experiences which can be described mechanically by player actions and gameplay challenges. We systematically look at potential player actions (as defined by basic cognitive and motor abilities) and gameplay challenges to understand how they relate to each other. We quantify these relationships by the importance of each action to the completion of a challenge. We summarize these relationships in several tables, separated by controller context. From these tables we draw conclusions about areas for novel gameplay, game analysis, and the impact of challenge design on people of differing abilities by examining trends in the data. We end by exploring ways to improve our methodology, refine our data, and other avenues to explore in the future.

Your Dedication
Optional second line

Acknowledgements

Acknowledgements go here.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Video games can be described as a system of systems; the component systems are the player and the game. Ernest Adams presents a model of the relationship between these systems [10], where the game (boxed in red) is divided into two of its components (user interface and core mechanics):

For our purposes, we are focused on the core mechanics of the game and so can treat the user interface like it's invisible. We end up with an idealized video game model closer to this:

The interactions between these systems create a specific gameplay experience which can be described mechanically by actions and challenges. We note here that the meaning of action has changed somewhat between Figures 1.1 and 1.2. In the original model actions are the in-game actions that are being performed by the player's avatar (e.g. jumping, swinging a sword, running) because of the player's inputs (e.g. pressing a button). In our idealised model, we use actions to mean inputs as we are interested in understanding what (physically and cognitively) the player does to respond to the challenge.

We are most interested in understanding the interactions of these systems as described by actions and challenges. We believe that understanding this interaction will allow us to examine how existing games are played, and potentially develop novel gameplay. To understand the interactions of these systems

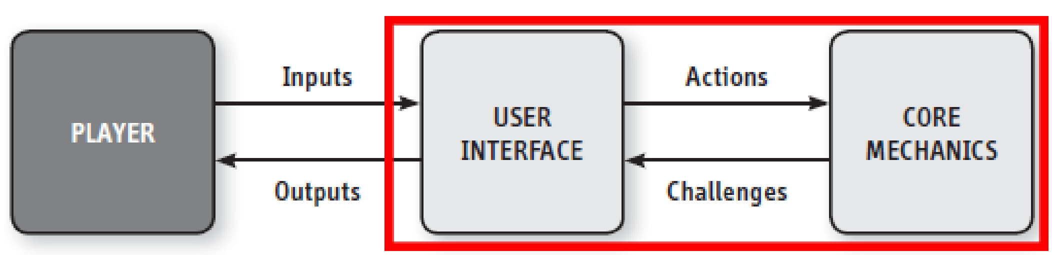

Figure 1.1: Adams' model of the relationship between the Player and the Game.

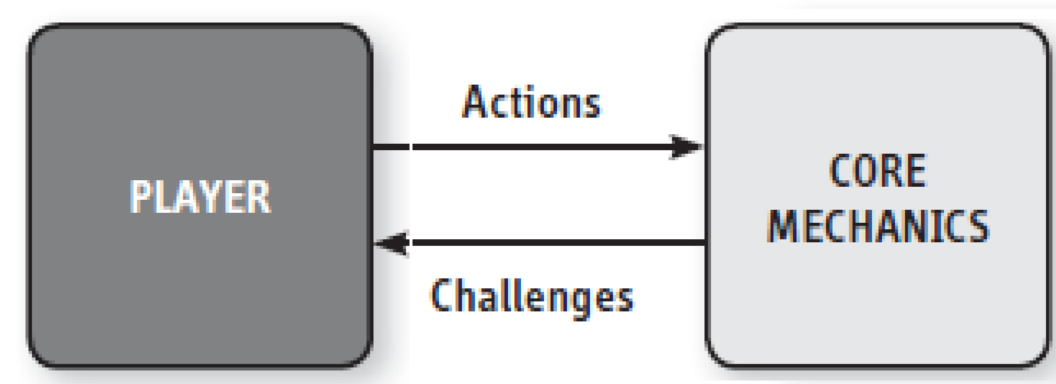

Figure 1.2: Idealised model of the relationship between the Player and the Game.

we need a player model (to describe the actions a player makes to interact with the game), and a gameplay model (to describe the challenges that the game poses for the player).

1.1 Problem Statement and Motivations

We want to study how motor and basic cognitive abilities are used to interact with gameplay challenges. This would help us find areas for novel gameplay and give us a way to discuss how players of differing abilities are impacted by challenge design.

We want to do this research for the following reasons:

1. We want to design better challenges for people of differing abilities
2. We want to explore novel motor experiences in games (potentially making new challenges)
3. We want to have a better way to formally discuss and categorize gameplay

1.2 Game System

As previously stated, the game system is (at a high level) comprised of two components [1]: the user interface, and the core mechanics. We scope away

[1] Adams describes a 3rd component — the storytelling engine. We neglect this component due to the scope of our work, which focuses on gameplay and not presentation and aesthetic game elements.

from the user interface because it would mean addressing questions of game aesthetics, presentation, and the "message" of a game. We instead focus on the core mechanics and specifically its output — challenges.

Core mechanics consist of the data and the algorithms that precisely define the game's rules and internal operations [10]. By extension the core mechanics define the gameplay by generating challenges for the player to complete. We informally understand challenges as atomic instances of gameplay; conversely gameplay is a series of challenges collected together. We turn to game studies and existing research on gameplay for a formal definition. Though we found 6 frameworks for analysing gameplay and categorizing challenges [10, 29, 55, 61, 108, 127], we found no agreed upon definition. We decided to make our own through synthesizing elements of the other definitions and our own experiences resulting in Definition 1.2.1:

Definition 1.2.1. *A **challenge** is an in-game activity with a success condition which engages the player in a way that requires some level of proficiency in at least one dimension (cognitive or physical).*

The goal (*success condition*) is defined either by the game or the player; therefore, we capture meta-game challenges like speedrunning, the Nuzlocke challenge, and other forms of innovative play in this definition.

To analyse the interactions between the player and game systems, we need a gameplay model to describe the challenges that will be presented to the player. This model should draw lines between *challenge families* (groups of "similar" challenges) and the challenges within them. We believe that with this distinction, we can make better inferences about which challenges work well together and other generalized trends. Before we can make a model like that, we need to define same and implicitly define different by extension.

Definition 1.2.2. *Two game challenges are the **same** if they:*

1. *Involve the same motor and cognitive skills from a player,*
2. *Occur over similar periods of time, and*
3. *Are performance-bounded by the same skill.*

The first point of this definition seems obvious we are looking to group together challenges that are mechanically experienced the same way by the player. In our experience as players, we found many games that are aesthetically different but mechanically the same. Since we want our model to be able to focus on the underlying gameplay and not the aesthetic differences, this point should group those instances together.

The third point is tied to the first; we are looking to group mechanically identical challenges, and we want these groups to be defined by a "dominant ability". The reason behind this is that there are a finite amount of human cognitive and motor abilities, and many games use the same subsets. If we were to just group them by presence of an ability, we would end up with wildly different challenge experiences being considered as the same. For example, shooting in a game like Halo and button mashing in a Mario Party minigame use the same hand movements and rely on the player's perception and attentional processes. However, when both games are played, they are experienced extremely differently. We believe this is because the "dominant ability" in these instances are different. For Halo your attentional and perceptional abilities are more important to react to enemies, while in Mario Party your fine motor abilities are pushed to the limit as you try to mash a button faster than your opponents. By identifying groups not just by abilities involved, but also the relative importance of each ability (*ability configuration*) we can get more well-defined groupings.

The second point comes from our experience as players, where we found that the duration of a challenge would impact how we approached it. Generally, we believe that this is due to fatigue. A game that asks you to swing a motion controller around for 5 seconds is measurably less taxing than a game that asks you to swing it around for an hour. Challenges that cause different amounts of fatigue are experienced differently even if their ability configurations are the same.

This definition of same (Def. 1.2.2) ignores the impact of aesthetics on experience. Games do not exist in a bubble; how a game is emotionally and socio-culturally perceived by a player has a definite effect on how they interpret and interact with the game. For example, while mechanically the games Bejewelled and HuniePop are identical, the aesthetics create a different experience making some players uncomfortable with the latter, but indifferent to the former. The topic of how players experience games on a meta level and how the mechanics may support that perception is out of our wheelhouse, and therefore out of scope of this project.

1.3 Player System

There are many ways to approach modeling a person. We approach player modeling from the player-as-machine perspective. This means that we focus on low level, mechanical player response (e.g. motor movements, perception, etc). We remove from our scope more complex processes like emotion, and societal level influences. We take inspiration from Newell's Model Human Processor [37] (see Figure 1.3) which separates a person into 3 subsystems (perceptual, motor, and cognitive) with a shared memory store.

Newell's model is insufficient for us to work with as it does not describe the individual abilities we are looking to model in any depth. We would like

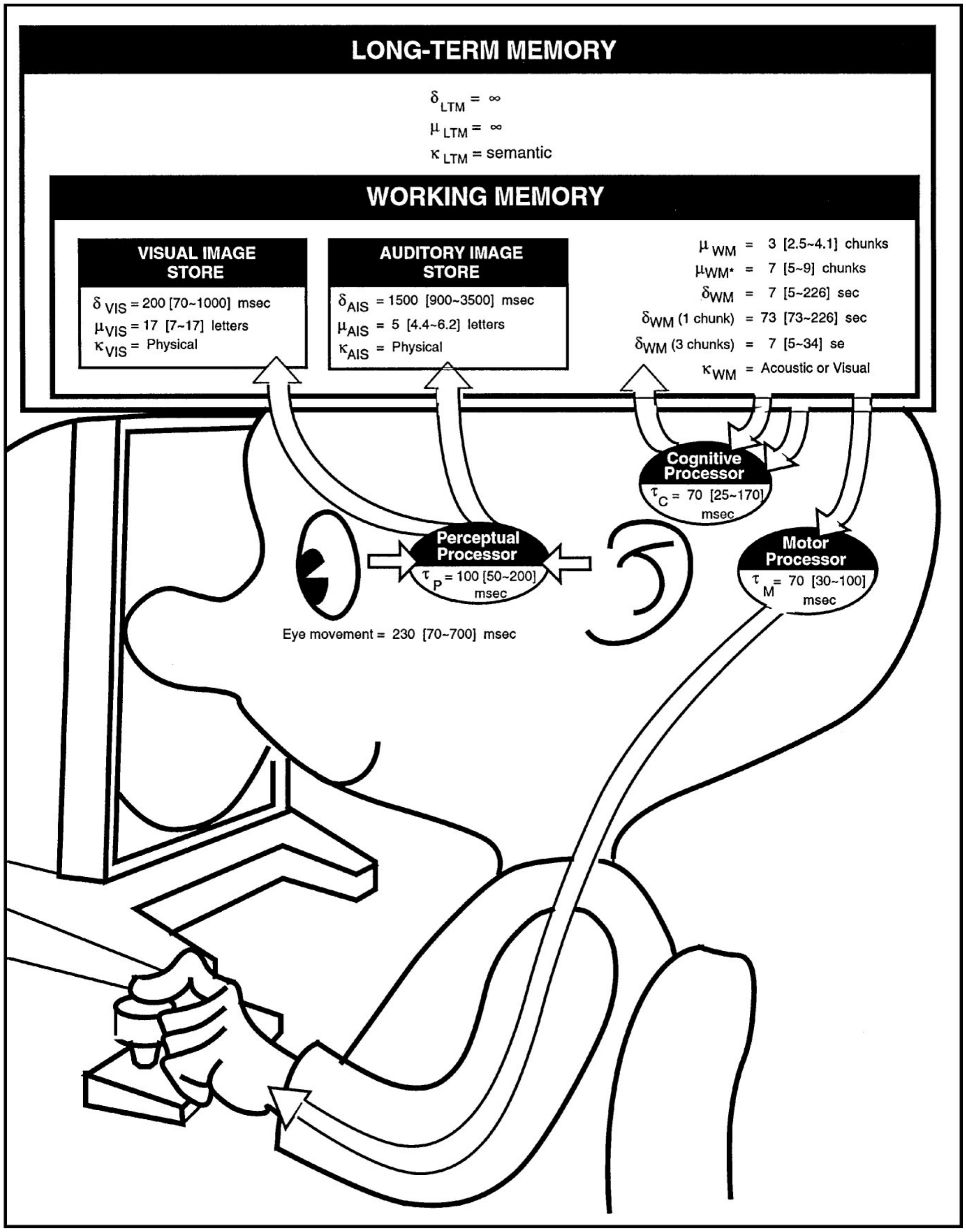

Figure 1.3: Idealised model of the relationship between the Player and the Game.

to have a complete model that could more accurately, describe the various cognitive and motor abilities of a player. However, modeling both would be too time intensive, and so we decide to focus on the motor abilities. Actions, for the scope of this research, are the resulting motor output of the player system after it uses various processes to interpret information. We need a player motor model to understand these actions which can concretely distinguish between different movements. Though we focus on the motor aspect, we do not want to neglect the cognitive areas too much; therefore, our player model would also need to encompass basic cognitive abilities (perception, attention, memory).

1.4 Outline and Contributions

This goal of this thesis is to analyse the interaction between the player and game models as defined by actions and challenges. We do this by creating several analysis tables which combine the player actions and gameplay challenges in order for us to analyse one with respect to the other.

Our major contributions are:

1. A definition of challenges
2. A definition of what it means for two challenges to be the "same"
3. Identifying 18 challenge types
4. Identifying 21 separate motor abilities as being relevant to games
5. An analysis of the relationship between challenges and motor abilities

Part I

Gameplay Challenges

This part of the thesis outlines a gameplay model based on challenges. We do this first by exploring existing models of challenges and gameplay. From there we determine whether any existing model can be used, and if not how to modify one to meet our expectations of a model. Our expectations are that the existing frameworks are able to delineate between different gameplay experiences (based on Definition 1.2.2). In this part we also explain our scoping for the gameplay model.

Before moving forward with discussing challenges, we need to explain the differences between challenges and puzzles. Challenges as we have described them, seem as though they are all puzzles that the player needs to solve. Janet Murray takes it a step further saying that all video games are "procedural puzzles" [130]. There is a lot of debate within the game design and game studies communities as to whether puzzles can be considered games [108]. From our perspective, puzzles cannot be easily dismissed as that would mean ignoring an entire market of games (puzzle games, e.g. Bejewelled, Candy Crush, Two Dots). In his book The Art of Game Desgin: A Book of Lenses, Schell defines puzzles as "a game with a dominant strategy" [181]. Though we agree with this statement, we believe it would be more correct to call a puzzle a challenge with a dominant strategy. Though puzzles are deeply ingrained in games, and often appear to serve as the main form of challenge in many games, we believe they are a subset of existing challenges. By acknowledging a puzzle as a type of challenge we can see that puzzles can both be used as the main gameplay (e.g. Bejewelled), or as a part of a larger game (e.g. Broken Age). We also see how puzzle-style challenges can be combined with more kinesthetic challenges (e.g. speed challenges) to create new gameplay experiences.

0.1 Related Works

In surveying literature on formal gameplay analysis we found six frameworks that deal with challenge in games. These are Karhulahti's challenges [108], Bjork and Holopainen's patterns in game design [29], Djaouti et al's bricks [55], Feil and Scattergood's challenges [61], McMahon's et al's challenges [127] and Adams' challenges [10]. In order to properly discuss and analyse these frameworks, we need to specify what we believe makes a good challenges framework. We believe a good framework should distinguish challenges as gameplay activities, and therefore be able to accurately describe different gameplay experiences. This means it should describe both the mechanics of the challenges (what the player is doing) and how the player experiences it. The following works have been analysed with this in mind.

Karhulahti defines challenges as a goal with an uncertain outcome [108] (a definition he borrows from Malone [125]). Karhulahti proposes two main challenge types: kinesthetic (where the "required nontrivial effort is at least partly psycho-motor"), and non-kinesthetic (where the "required nontrivial effort is entirely cognitive"). He separates non-kinesthetic challenges into static (puzzles) and dynamic (strategic) challenges. Dynamic challenges are further categorized into one of five types: directly dynamic, indirectly dynamic, totally dynamic, and quasi-dynamic. In the context of video games, he explains that any gameplay challenge (e.g. completing a test room in Portal) is potentially composed of a non-kinesthetic challenge (solving the challenge cognitively) and a kinesthetic challenge (executing that solution). Though we agree that gameplay challenges have both cognitive-focused and motor focused parts, we believe his framework is too focused on a system level description of challenges (i.e. whether the outcome is deterministic). We believe that it is more useful in game analysis to work at a gameplay level that describes the core challenge concept as experienced by the player. For example, in Karhulahti's framework solving a puzzle in Professor Layton and the Diabolical Box, solving Uncharted: The Lost Legacy's shadow puzzle, and solving a case in Phoenix Wright: Ace Attorney are all the same challenge (static challenge structure). However, in playing these games it's easy to see they are entirely different experiences that call on the player to use different abilities.

In Patterns in Game Design, Bjork and Holopainen outline a framework to describe and analyse games [29]. Along with this framework, they developed a set of tools called patterns which are "semiformal interdependent descriptions of commonly recurring parts of the design of a game that concerns gameplay." These patterns are functionally building blocks in game design, and so they include elements that can define challenges (e.g. goals, actions, obstacles). Though it is possible to combine patterns to create a potential list of "challenges", we believe that the created challenges would not be descriptive enough to distinguish between gameplay instances that use the same components but are experienced differently by the player. Consider the *guard* pattern ("to hinder other players or game elements from accessing a particular area in the game or a particular game element"); when combined with the patterns *characters* ("abstract representations of persons in a game"), *enemies* ("avatars and units that hinder the players trying to complete the goals"), and *aim & shoot* ("the act of taking aim at something and then shooting it") it can describe scenarios as different as protecting Ashley in Resident Evil 4, and protecting baby Mario in Yoshi's Island. Resident Evil 4 is a 3D, third-person action/survival horror game, where the player (as Leon Kennedy) must protect Ashley from being kidnapped by infected villagers. This is done by killing enemies before they can reach her, or as they are trying to escape with her as the player navigates through the levels. The player's primary form of defense are guns, to be aimed and fired from an over the shoulder perspective. The player has the ability to command Ashley to wait or follow them at will. Yoshi's Island is a 2D, side-scrolling action/platformer game, where the player controls a series of yoshi as they escort baby Mario to rescue baby Luigi. The player can protect baby Mario by having Yoshi create and throw eggs at enemies and environmental obstacles. Though the description of the gameplay is appropriate based on the components,the player's experience of the gameplay (i.e. how they interact with and interpret these challenges) is wildly different due to a variety of factors (e.g. camera perspective, pacing, etc). This difference in experience relates to how differently a player thinks when approaching these two scenarios (guarding a character you can control vs. one you cannot) and how they choose to approach this challenge. Creating challenges out of patterns would not give us the ability to describe how these challenges are meant to be played. As we are looking to relate challenges to player cognitive and motor abilities, it is important that our challenge descriptions evoke more understanding about how the challenges are played.

Djaouti, Alvarex, Jessel, Methel, and Molinier created a gameplay mechanic focused classification and analysis tool called "bricks"[55]. They identified three types of bricks: play, game, and meta. *Play bricks* are the actions that a player can take, and include: manage, random, shoot, write, move, and select. *Game bricks* are the goals of the game, and include: destroy, match, avoid, and create. Play bricks and game bricks combine to form *meta-bricks*, which describe families of challenges. Examples of meta-bricks include the "DRIVER" meta-brick, which combines the "MOVE" and "AVOID" bricks. In creating this tool they analysed 588 single player computer games, most of which they reported as being "arcade" or "casual" games. This limitation in the scope of analysed games makes this tool concise, but not powerful enough to describe all types of challenges across game types. For instance, combat in League of Legends at its most abstract it would be classified as a "KILLER" meta-brick, combining the "SHOOT" and "DESTROY" bricks. However, the player actually does a significant amount of character management (e.g. monitoring cooldowns, buying items) and requires a large amount of strategic knowledge (e.g. timing, last hits, abilities) to actually engage in combat effectively. This depth of challenge is lost in this model. A related issue of this framework is that it does not account for the various ways in which these challenges present themselves and how that influences player reactions. For example, Djaouti et al. use Pac-Man and Need for Speed as examples of the "DRIVER" meta-brick, explaining that this meta-brick exists in any game where the player is tasked with navigating around obstacles on a map. This implies larger games like The Elder Scrolls V: Skyrim would also use the "DRIVER" meta-brick. Though in an abstract form this is true, the change in perspective (camera model), context, and general complexity of the game changes how the player chooses to act in regards to a "DRIVER" style challenge. Therefore this tool lacks the ability to capture a nuanced description of the player's experience. Though Djaouti et al. present this tool as a work in progress to which more game bricks should be added, there is too much work to be done in adding bricks to this framework to make it worthwhile for us. As well, this tool completely separates the player from the gameplay, which is something we feel limits its ability to thoroughly analyse a game.

In Beginning Game Level Design, Feil and Scattergood explain that challenges are defined by "objectives, and the barriers that prevent players from achieving [them]" [61]. They identify six standard challenges: time, dexterity, endurance, memory/knowledge, cleverness/logic, and resource control. Though these challenges incorporate an element of the player's experience into their definitions, they were too broad to be meaningful. For example, they defined dexterity challenges as "some sort of feat that requires dexterity". Though Feil and Scattergood explicitly list these six challenges, they use the term rather liberally throughout the rest of their book explaining that certain genres put more emphasis on combat, movement, or puzzle challenges. As such, we can see that the list they present is incomplete, and so would require not just more precision in the given challenges' definitions but a complete expansion.

Ernest Adams, in his book The Fundamentals of Game Design, defined challenges as "any task set for the player that is non-trivial to accomplish" [10]. He presents 10 major challenge types, subdivided into 30 specific challenges (see Table 1). Adams' challenges, while extensive, are incredibly broad. This broadness creates situations where multiple gameplay instances, which are sufficiently different in play experiences, are blanketed under the same challenge. For example, speed and reaction time challenges can describe gameplay instances ranging from quick time events in God of War, to button mashing mini-games in Mario Party.

McMahon, Wyeth, and Johnson presented a refined form of Adams' challenges, scoped down to 16 challenges [127]. Through a session with a focus group, they renamed several of Adams' challenges and added 3 new categories of challenge. This resulted in the following challenges: thinking outside of the box, solving the unsolved, mastering complex controls, remembering, identifying and applying patterns, searching for objects and spaces, navigating and understanding spaces, micromanaging, planning and strategizing, overcoming obstacles and enemies, collecting, creating something for its own sake, directing narrative, role-playing, understanding active systems, trivia, and construction with a functional goal. We have multiple issues with this refinement; firstly, we believe that this condensed list further obscures challenges. For example, the "thinking outside the box" challenge is associated with the games Portal and World of Goo. There is no indication from this challenge type about how the gameplay is actually experienced. The types of puzzles presented in Portal are extremely different from those in World of Goo due to the mechanics and perspective of the games. We agree that Adams' list needs to be refined, but

Challenge Type	**Challenges**
Physical Coordination	Speed and Reaction Time Accuracy and Precision Timing and Rhythm Learning Combination Moves
Formal Logic	Deduction and Decoding
Pattern Recognition	Static Patterns Patterns of Movement and Change
Time Pressure	Beating the Clock Achieving something before someone else
Memory and Knowledge	Trivia Recollection of objects or patterns
Exploration Challenges	Identifying spatial relationships Finding keys (unlocking any space) Finding hidden passages Mazes and Illogical spaces
Conflict	Strategy, tactics, and logistics Survival Reduction of enemy forces Defending vulnerable items or units Stealth
Economic	Accumulating resources or points (growth) Establishing efficient production systems Achieving balance or stability in a system Caring for living things
Conceptual Reasoning	Sifting clues from red herrings Detecting hidden meanings Understanding social relationships Lateral thinking
Creation and Construction	Aesthetic success(beauty or elegance) Construction with a functional goal

Table 1: Gameplay Challenges from Adams

believe it needs to be expanded and made more specific. As well, we take issue with the way that the new challenges were created. The focus group generating these new activities seem to have assigned entire games as being representative of their new challenge types. Games are extremely large entities that are composed of multiple instances of challenges. We believe that the development of challenges should be based in specific gameplay instances.

0.2 Methodology

Having explored the various existing gameplay models and finding them lacking (for our purposes), we choose to create our own. Though we intended to make a new gameplay model, we believe that it would be more prudent to manipulate an existing model to suit our purposes. Adams' list is currently the most in-depth and explicit, even though his definitions are not precise enough, and so we used it as the starting point for our own analysis.

From analysing his definitions and comparing them to existing games, we realised his atomic challenges would need to be further deconstructed to provide a better overall picture for our new framework. These deconstructions would need to focus equally on the task asked of the player, and the mechanical actions that are required to complete them. For challenges that rely more heavily on motor skills, the mechanics will be essential in explaining and justifying where challenge boundaries lie.

Our methodology in addressing this was to understand the core qualities of his challenges, and compile examples of existing games with gameplay instances that matched those qualities. We then looked within those examples and grouped them based on commonalities like time limits, goals, input methods and physical movements, etc. This grouping activity gave us a better idea of where lines existed between challenges that Adams' would have considered the same. We then looked at each individual grouping and listed the qualities of those groups, creating an ad-hoc definition of those challenge types. The goal of these new definitions is to capture specific gameplay experiences, allowing us to easily see when gameplay instances are similar or different from a play perspective.

The following set of chapters will outline more formalised definitions for our proposed challenges, along with examples of the new challenges in existing commercial games.

0.3 Scoping

As we can see from Table 1, there are many challenges to explore; each potentially exploding when we begin to decompose them. It is unrealistic to believe that our project could devote time to all of them. We therefore choose to limit ourselves to discussing a sample of challenges in depth, rather than trying to achieve a shallow breadth of understanding. In this section we will further narrow our scope in regards to the types of challenges we will discuss, their various factors, etc.

0.3.1 Selection of Challenges

Since we are unable to adequately explore all of Adams' challenges at this time, it is important that the challenges we choose to study are good. At this time, a good selection should showcase the following underlying assumptions of our work:

1. that Adams' groupings are too broad
2. that certain challenges are more physically (motor skill) oriented
3. that certain challenges are more thinking (cognitive skill) oriented

Due to our limitations in scope, we focus on these first two assumptions. As such, we select challenges that exemplify these assumptions to illustrate why video games need to be examined in this way. We will therefore be examining Adams' speed and reaction time challenges, and learning combination moves challenge to examine the first and second assumptions respectively.

Decomposing Challenges Rationale

Before moving on to the individual chapters, we provide a condensed explanation as to why we believe Adams groupings to be too broad. Though this section presents a broad rationale as to why we have decomposed certain challenges, we believe that the following chapters which are devoted to the decompositions, will further support our reasoning.

In his categorization, Adams originally grouped speed and reaction time challenges together. This is primarily due to their similarities in mechanics and appearance in video games, as Adams makes special mention of the commercial genres in which they can be found [10]. By grouping them together this way it conflates the different skills necessary to each type of challenge, and creates an incorrect understanding of the essence of these challenges themselves. Consider the cartoon-ish simulation game Cooking Mama [53], and the first-person shooter game Halo: Combat Evolved [31]; both use speed/reaction time challenges under Adams categorization. The difference in the presentation of challenges, and the contexts of speed and reaction time challenges in each game prove that they are wildly different. Cooking Mama mentally prepares the player for any reaction time tasks, thereby priming them for response to a cue. In contrast the pacing and atmosphere of Halo requires that players are caught off guard in reaction time challenges. From the stand point of cognitive scientists this would indicate different processes at play. The discrepancy between the skills used in both games and perceptions from players illustrates the need to separate them into distinct categories, which Adams 'current categorization does not do. This is why this paper proposes separating speed and reaction time challenges into their own categories.

Motor Oriented Rationale

This section presents a broad rationale for why we believe certain challenges are more motor or cognitive oriented. As with the previous section (Section 0.3.1), we believe that the following chapters which explain the challenges will further support our reasoning.

All interactive activities require some combination of cognitive and motor abilities; catching a ball requires the cognitive systems of attention and perception, along with the motor systems of gross motor abilities (e.g. arm movements). The same is true with video games, which require perceptional skills at minimum on the cognitive side, and fine motor abilities (e.g. finger movements for controller input) on the motor side. We believe that the difficulty of any gameplay challenge is largely due to which cognitive and motor abilities it asks the player to use, and how developed those skills are in the player. For example, in chess the challenge of the game is in anticipating your opponent's movement and developing a strategy to win in spite of them. For the average young child (e.g. 4-6 years old), they have the capacity to learn the mechanics of the game (how pieces move, what pieces are worth) and to enact individual moves (move their pieces using their hands and arms), but are not developmentally able to strategize further than their current move. This is why when a child and an adult are playing a game of chess to the best of their respective abilities, it's more likely that the adult will win. Therefore we would conclude that cognitive skills are the limiting factor in a game like chess. Obviously there are exceptions to this, as child prodigies exist in many areas, but we believe this reasoning holds for the average individual at any given life stage. A video game example that is age independent is memorization challenges; in the Mario Party 3 mini-game Messy Memory requires players to memorize the order of objects on three different shelves and to replace them accurately after the objects are dropped to the floor [94]. The difficulty in this game is remembering the object locations, as picking up and placing the objects on the shelf is a single button press. For children with developing short term memories this task is incredibly difficult, but as their memory improves with age the task becomes easier. A more motor focused example would be rhythm games like Dance Dance Revolution, where an elderly individual can cognitively understand the challenge of the game (hit the buttons at the right time using your feet), but will have a hard time enacting them due to limitations in the motor abilities (moving slower due to old age). We believe that in being able to identify which challenges are more cognitive or motor focused, we will be able to better understand how players interact with their games, why certain games "fail" in the market (if they are unplayable for their target demographic), and whether there are any untapped potential challenges based on cognitive and motor abilities not currently being used.

0.3.2 Restrictions on Player Ability

As the game industry expands to include markets such as eSports and mobile gaming, the range of skill required to play any individual game varies immensely. This range in skill means it will be almost impossible for our framework to be generalizable to every potential player, because what is difficult for a novice may be elementary to an expert level player. In an attempt to make the results of this preliminary study as universal as possible, we decided to focus on the abilities and play styles of the "average" player. In this case, we identify the average player to be between 18-35 years of age. This age was chosen to correspond with majority of players in the 2017 gaming market. The Entertainment Software Association (ESA) reports that the average gamer is 35 years old, with 72% of the market being 18 years or older [17]. This age also corresponds with the physical and mental peak of human development, and as such will give us a good baseline for future comparison. In creating the persona of an "average player" we immediately eliminate various modes of play and additional abilities and challenges from our framework. Expert level players often stray from the intended interaction methods with the games, focusing on meta-level gaming or creating harder challenges for themselves. Included in expert level play are speed-runs, professional/competitive eSports play, and engaging in other forms of challenge that are not imposed by the game. An example of the last one is the Nuzlocke challenge in Pokemon, wherein the player cannot revive any "fainted" Pokemon, must release them immediately upon "fainting", must catch the first Pokemon they encounter in an area, and must restart the game should they "black/white out" (have all their Pokemon "faint"). Meta challenges like these are meant to add additional levels of difficulty for players who have already bested the main game. There are too many types of meta-challenges possible to try and encapsulate them all in our framework. As such, restricting ourselves to discussing "average players" also restricts us to discussing "average challenges" - and so we can apply our framework to a wider set of games.

0.3.3 Restrictions on Game Types

Similar to the restrictions on type of player, we are looking to restrict ourselves to discussing "average" games as well. For our purposes this excludes augmented reality (AR) and virtual reality (VR) games. Though these are growing markets, there is still not enough widespread gaming development for them in order to justify examining them. Our framework is open for modification in the future when a larger catalogue of games exist for these platforms, such that we could adequately support any claims made about these platforms and their interactions.

As noted by those before us [10, 61], many instances of challenges in video games are actually "composite challenges" (gameplay created from multiple challenge types being used simultaneously). Composite challenges often have a primary challenge style, with certain gameplay elements being added from a secondary challenge type to either increase the difficulty or create variety in gameplay experiences. An example of this Cooking Mama's "Add Flavouring" mini-game [53]; on the surface it looks identical to other rapid tapping style challenges such as Almost a Hero [26] and Cookie Clicker [107], but when played the difference in challenge context (multiplayer vs. single player) shows an added "beating the clock/racing other players" element that changes the gameplaying experience. Since the goal of our model is to create definitions along the lines of gameplay experiences we restrict ourselves to only talking about "pure challenges" (not "composite challenges").

Chapter 1

Speed Challenges

Speed challenges, as defined by Adams, "test the player's ability to make rapid inputs on the controls" [10]. As success in these challenges is determined solely by how quickly a series of inputs is made, this would make it an almost entirely psycho-motor process. In examining commercial games, the majority of challenges appear to have a speed component to them – whether it is finishing a race in first place, or performing a special move, both must be completed in a timely manner. The difference between a challenge that has a speed component, and a speed challenge, is the necessity of cognitive skills beyond simple perception. Therefore, when analysing gameplay as a whole, there are actually very few pure speed challenges in contemporary video games. In examining a wide variety of commercial games, it was found that the speed challenges available to video games are: button mashing, rapid analog stick rotation, rapid tapping, scribbling, rapid controller rotation, and rapid controller shaking. The differences between these are minor in appearance and are often due to the difference in input devices when they are implemented.

1.1 Button Mashing

Button mashing is a type of video game challenge in which the player must press a button on their controller, or key on their keyboard, as quickly as possible. This can further be broken up into three types of button mashing challenges: single input, multiple input, and alternating input. An important aspect of button mashing challenges is that they are time sensitive. That is to say, there is usually a predetermined amount of time in which the players'inputs are registered as being relevant to the challenge at hand, after which pressing the buttons has no impact on the results of the challenge. Though button mashing is commonly seen in video games, it rarely exists in isolation; rather, it is often incorporated as a component in other challenge types, such as reaction time challenges. As such, the majority of examples for button mashing in isolation come from games that are compilations of mini-games, or games where the combat is turn based.

Before moving forward, it is important to note the difference between button mashing as a challenge in a video game, and button mashing as a strategy employed by players. The main difference between the two is randomness. Button mashing as a challenge is guided and not random as it is the intended play method. This would mean that the game must instruct the player in some way that button mashing is the appropriate action in this scenario. In opposition, button mashing as a strategy is predicated on randomness; the player chooses to ignore the intended challenge and supplements the gameplay with random input. Public opinion in the video game community derides button mashing as a strategy and actively criticises the inclusion of button mashing segments as poor game design [254, 242, 247, 241]. It is difficult to tell whether this contempt is due to conflating button mashing as a challenge and button mashing as a strategy. Most discussion of button mashing in public forums seems to indicate that this is the case, as it tends to only discuss button mashing in relation to combat styled scenarios. Button mashing as a strategy is found in response to several different challenge types. Most notably, it is often referred to in relation to the appearance of the "learning complex moves and combos" challenge type, wherein players will repeatedly press the attack button without regard to the predefined combination moves. When "learning complex moves and combos" challenges are presented in combat, it is often possible for inexperienced players to successfully complete the combat through the use of strategic button mashing, at times even being able to best other players due to the innate randomness. This is then perceived as unfair as it rewards a lack of skill, and it is this unfairness that seems to be the source of public ire against button mashing as a whole [254, 242, 247, 241]. Professional opinion on the topic is mixed [64]. This is possibly due to the growing overlap between players and creators in the video game field, which would then result in public opinion being imposed on game design. As more players become designers, they would choose only to include what they feel is "good" gameplay challenges into their games. Button mashing is a legitimate gameplay challenge, and it is not the purpose of this paper to determine whether any form of gameplay is "good" or "bad". Thus in this paper, any further reference to button mashing is meant to refer to the intended method of gameplay and not the strategic choice.

1.1.1 Single Button Input

Single input button mashing is the most frequently seen variation of button mashing. Games as different as South Park: The Stick of Truth [157], Bayonetta [166, 167], and Mario Party [92] all employ this type of challenge. Single input button mashing conforms to the straightforward definition of a button mashing challenge, wherein the player must repeatedly press a specific single button on the controller as fast as possible within a given time limit. The most common examples come in the form of mini-games, such as those found in the Mario Party series. One particularly illustrative example is the "Manic Mallets" mini-game in Mario Party 5 [96]. In said mini-game, teams of two players must repeatedly hit a switch with a hammer to avoid being crushed by a bigger hammer. The mini-game lasts for only ten seconds, and successful completion is only dependent on the number of hammer hits the players executed during the time limit. In this way, it can be seen that the only way to play is through button mashing, and it is not a strategy that the players'choose to employ. Another case of single input button mashing in contemporary games is through combat. South Park: The Stick of Truth has multiple attacks that requires the player to mash on an indicated button to complete the move successfully. One such attack is "Dragon's Breath", a starting move for the mage class, where the player is informed to mash the A button in order to wave a lit firecracker in their opponent's face [157]. The game imposes an implicit time limit for how long the button mashing is useful by syncing the button mashing with the length of the attack animation. An identical button mashing combat set up is found in the Bayonetta series by Platinum Games. In the Bayonetta games it is possible for the player to trigger what the game calls "Torture Attacks" while in the middle of combat these attacks serve as cinematic button mashing challenges that land the finishing blow to the enemy and increase the player's score [166, 167].

1.1.2 Alternating Button Input

Alternate input button mashing is equally as abundant as single input button mashing in the commercial video game sphere. Alternating input button mashing requires that players repeatedly press two specific buttons in sequence. Examples of this type of challenge can be seen in the Mario Party franchise. In Mario Party 2, the mini-game "Psychic Safari" tasks players with powering up an ancient relic and destroying their opponent's relic using their psychic powers within 5 seconds [93]. In order to do so the game instructs the player to press the A and B button alternately [93]. Similar mini-games exist across the Mario Party series such as: "Rockin'Raceway" (Mario Party 3) [94], and "Slime Time" and "Take a Breather" (Mario Party 4) [95]. Another example is the speed skating family of events from the Nintendo DS version of Mario and Sonic at the Winter Olympic Games. All of the speed skating events are played by requiring the play to alternately mash the L and R shoulder buttons [186]. Cycling in *Mario and Sonic at the Olympic Games* for the Nintendo DS

Figure 1.1: Dragon's Breath attack from South Park: The Stick of Truth with instructions on screen.

Figure 1.2: Mario and Sonic at the Olympic Games cycling events with controls.

is played the same way (Figure 1.2) [183].Though there are numerous examples of alternate input button mashing on its own, it also frequently appears alongside the other forms of button mashing in a singular challenge. In the Mario Party 3 mini-game “Ridiculous Relay” the 3 player team is tasked with performing various input patterns to complete their version of the relay race. This includes an alternating input button mashing segment, a learning combination moves segment, and a single input button mashing segment all in succession [94].

1.1.3 Multiple Button Input

Multiple input button mashing is the least frequent variation of button mashing. There are very few cases of it appearing in commercial games outside of mini-game based party games. Multiple input button mashing requires that the player coordinate pushing multiple buttons simultaneously and rapidly. One example of a multiple input button mashing challenge is the “Mecha-Marathon” mini-game in Mario Party 2 [93]. In this game players must simultaneously press the A and B buttons as quickly as they can within ten seconds. The number of times the player is able to successfully press this button combination determines how far their wind-up doll will fly after the 10 seconds are up; the winner is determined by whose wind-up doll flew the furthest [93]. As with alternating input button mashing, multiple input button mashing is also found as a component of larger challenge types that combine button mashing types. The Mario Party 4 mini-game “Mario Medley” presents multiple input button mashing alongside alternating input button mashing [95]. There is no set reason as to why multiple input button mashing is less commonly used than its counterparts. It is possible that its unpopularity could stem from the difficulty in coordinating multiple simultaneous button presses. This difficulty could potentially explain why this style of challenge tends to be limited to two button input, as having to remember and coordinate pressing more buttons would be too taxing on the player’s cognitive and psycho-motor skills. Another possible reason that this type of challenge is not popular is the similarity in skills used to the single button input challenge. It is possible that designers do not consider them to be different enough from each other, and as such choose

Figure 1.3: Final Fantasy X, Lulu's Overdrive instructions

to include the cognitively simpler single input challenge type instead.

1.2 Rapid Analog Stick Rotation

Rapid analog stick rotation is a video game challenge in which the player must rotate the analog joysticks on their controller multiple times within a given time frame. An important distinction about this particular challenge is that the interpretation of the mechanics is more lenient than with button mashing. For button mashing, it is important that the game dictates which buttons are to be pressed in order for the input to be considered correct — by contrast, this challenge does not necessarily need to instruct the player which direction to spin the analog stick on their controller as the input is considered the same regardless of whether it was spun clockwise or counter-clockwise. One example of this type of challenge is Lulu's "overdrive "from Final Fantasy X [205]. When activated the player is instructed to "Rotate the right analog stick round and round "and a timer begins to count down from four seconds [205]. This particular challenge adds a level of difficulty by having the size of one rotation be conditional upon the character 's magic stat, meaning that at its most difficult this particular challenge would require the player to rotate the stick 720 degrees to be considered one valid rotation [205]. Difficulty scaling like this does not often occur in challenges of this type, and frequently it is a 360 degree rotation of the controller that equates to one successful rotation. In South Park: The Stick of Truth, the Jew class character has an attack that also falls under the category of rapid analog stick rotation. The "Whirling Doom "attack instructs the player to "spin the left analog stick "in order to send a dreidel flying at the enemy to deal damage [157]. Rapid analog stick rotation is also found in the Mario Party mini-game "Pedal Power" [92]. The game instructs the players to "rotate the joystick to light up the light bulb" [92] As with the previous examples, the direction of rotation is left unspecified and irrelevant to the completion of the challenge.

Figure 1.4: Goat Simulator Interface on Pixel XL

1.3 Rapid Tapping

Rapid tapping is a type of video game challenge in which the player must press the touch screen on their touch device as quickly as possible. At its core, rapid tapping is button mashing as experienced through a touch interface. This paper considers it a separate challenge type due to this difference in hardware interface and how that difference affects the specific psycho-motor and cognitive skills that are used to complete the challenge. As with the button mashing challenge type before it, rapid tapping is subdivided into more specific challenge types. The subtypes of rapid tapping are: indiscriminate rapid tapping, and alternating tapping. It is interesting to note that multi-touch tapping challenges are not present as a subtype; this is because they were unable to be found when the research into video game challenge types was being conducted. This is not to say there aren't games that utilise multi-touch abilities on a touch screen. Many games that are ported from consoles to touch screen mobile devices will include onscreen controls to replace the console controller, and will therefore use the multi-touch functions to emulate analogue stick movements alongside buttons [49, 34, 28]. Therefore lack of a multi-touch rapid tapping sub-challenge is only meant to indicate that while multi-touch controls and challenges exist in games, they are not used in a rapid tap context.

1.3.1 Indiscriminate Tapping

Indiscriminate tapping is a rapid tapping sub-challenge that tasks the player to tap on the touch screen as quickly as possible. Indiscriminate rapid tapping is not the touch screen equivalent of single input button mashing. In terms of frequency of appearance in commercial games, indiscriminate tapping is no where near as prevalent across all platforms. This is potentially due to designers wanting to leverage the accuracy and precision abilities of touch screens, where previously with controllers, accuracy and precision challenges were difficult to implement and play. The "indiscriminate" refers to the fact that tapping anywhere on the screen counts as a valid tap. This is an important distinction to make as the inclusion of a target tapping location would add a level of cognitive complexity to this challenge that would remove it from the speed challenge category. In *Cooking Mama* for the Nintendo DS, the mini-games "Tenderize the Meat", and "Add Flavoring" are both examples of this challenge; players are given 10 seconds to tap the screen until the on-screen meter is full (Figure **??**) [53]. The meter fills when tapping anywhere on screen, although the meter does fill faster if the player taps all over the screen rather than just in one area. This encourages indiscriminate tapping but does not penalize consistency. If the player does not sufficiently fill the meter in the ten seconds the titular character, Mama, appears on screen, "fixes" the dish and the game allows the player to move on to the next challenge. *Dumb Ways to Die*'s Hot Head [106] mini-game is another example.

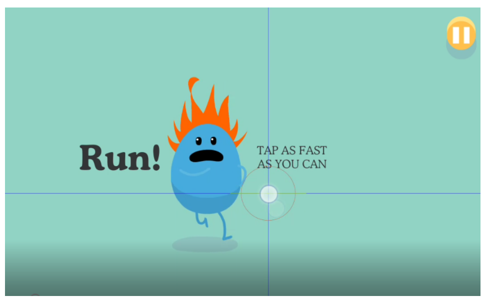

Figure 1.5: Dumb Ways to Die's Hot Head mini-game

Indiscriminate rapid tapping is a common challenge type in mobile games, specifically through the "idle" game genre like Cookie Clicker [107], Almost a Hero [26], and many others.

1.3.2 Alternate Tapping

Alternate rapid tapping is virtually the touch screen equivalent of alternate input button mashing. In both mechanics and appearance the two challenge types are exceptionally similar. Alternate rapid tapping involves the player tapping two or more distinct parts of the screen in a particular repeated and alternating sequence. Commonly this will be implemented as the player alternating tapping between the left and right half of the screen, but any subdivision of the screen fits this description as well. Mario and Sonic at the London 2012 Olympic Games for Nintendo 3DS has an example of the left-right version of alternate tapping in the mini-game "100m Freestyle/Paddle Fingers" [189]. The game instructs players to lay down their 3DS and alternately tap the left and right sides of the screen in order to fill a stamina bar and make their character swim faster [189].

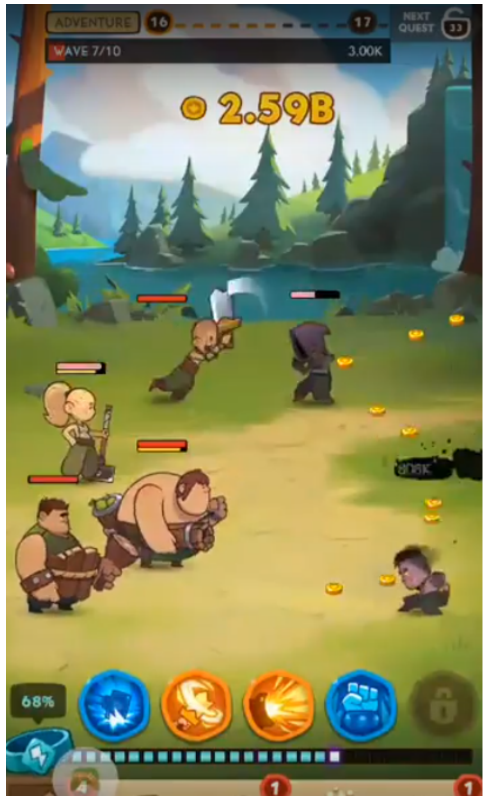

Figure 1.6: Almost A Hero interface for Android devices where players can tap anywhere

1.4 Scribbling

Scribbling is a gameplay challenge that involves repeated quick, fluid, and continuous motion on a touch screen within a set interval of time. Scribbling style challenges only appear in games designed for a touch screen interface, most notably for touch screens that require the use of a stylus. In the context of these video game challenges scribbling is defined using the Oxford English Dictionary definition of to "write or draw (something) carelessly or hurriedly". This definition implies the necessity of speed inherent to the gaming challenge as well as encapsulating the idea that the player may be asked to draw something that is not a recognizable shape or polygon. Scribbling is subdivided into the following challenges: rapid line drawing, and rapid shape drawing. Though similar, the reason for the subdivision is due to the added mental process of shape recognition, which requires a person to recall what a shape looks like before trying to simulate it on a gamepad. In contrast, repeatedly moving a stylus back and forth as if scribbling a line makes for a cognitively simpler task. It is important to note that scribbling challenges are often open to interpretation by the player. Most of the time the player is just told to rub on the screen leaving it up to them to determine the most efficient way to do it. As previously stated this paper concerns itself with the challenge types and not the strategies the player's employ to complete them.

1.4.1 Rapid Line Drawing

Rapid line drawing is a scribbling sub-challenge that requires the player to move their stylus in a back and forth motion. In this type of challenge, the accuracy of the line drawn by the player is considered unimportant. Cooking Mama employs this challenge in its most basic form. During the "Make Shape "mini-game, the player is instructed to "the ingredients in hands to make shape "accompanied by a directional arrow from one hand to the other when the player begins playing [53]. It is important to note that though the player is required to move the stylus back and forth it does not mean they have to follow a set line. Meaning that so long as the game encourages players to make quick, single motion strokes from one point on the screen to another it would qualify as a rapid line drawing task. Examples of rapid line drawing challenges that do not give the player a set line are found in WarioWare: Touched! for the Nintendo DS. "Impressionism" is one of these mini-games, in which the player is instructed to "rub the paper to make a copy of the coin underneath" [101]. To do so the player must repeatedly move the stylus all over the screen to uncover the image below within the time limit. Similar mechanics are employed in the mini-games "Chalk Full", and "Pet Petter" from the same game. Rapid line drawing also appears alongside other challenge types in single mini-games. Mario and Sonic at the London 2012 Olympic Games for the 3DS uses rapid line drawing in their "4 x 100m Relay" event [189]. In this mini-game the players are instructed to slide their stylus back and forth from left to right in order to make their character run faster, and then touch the screen when appropriate to pass the baton [189]. Cooking Mama also uses this gameplay challenge in the "Grate" mini-game [53]. In "Grate" the player is instructed to slide their stylus left and right across the screen to grate a food item, and tap the screen when the grate become blocked with shavings. Rapid line drawing can also be found in the mobile gaming industry. Dumb Ways to Die features this challenge through the "eat a two week old unrefrigerated pie "in which the player must wipe vomit off the screen [106]. This minigame is mechanically identical to the WarioWare challenges with the stylus being replaced by the player's finger.

1.4.2 Rapid Shape Drawing

Rapid shape drawing is a scribbling subtype that combines the motions of rapid line drawing and rapid analog stick rotation. This type of challenge requires

Figure 1.7: Clean Up Throw Up mini-game from Dumb Ways to Die 2

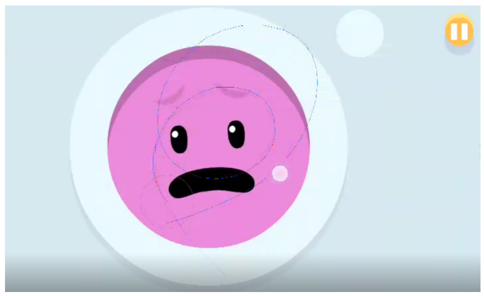

Figure 1.8: Dumb Ways to Die Spin! mini-game

players draw a reasonable approximation of the challenge specified shape as many times as possible within a given time limit. Most often the shape presented is a circle, though other types of shapes could also appear without the set of necessary psycho-motor and cognitive skills changing. Cooking Mama involves this type of gameplay challenge in the “Wash Rice” and “Mix” mini-games [53]. In these challenges the players are instructed by the on screen directions to draw small circles on the screen as quickly as possible to mix the ingredients or wash the rice [53]. These mini-games add a further level of challenge by requiring that the player aim to have the amount of stirring fall in the green range of an on screen gauge at the end of the time limit (see Figure below). This additional level of complexity is not required for a challenge to be of this type. Dumb Ways to Die 2 incorporates this challenge without this complexity. The “spin” mini-game requires the player draw a circle as quickly as possible to complete the hammer toss [128]. The “Do or Dry” mini-game in *Dumb Ways to Die* is the same (Figure 1.8). As with the rapid line drawing before it, rapid shape drawing does not consider accuracy of shape tracing essential to successful completion.

1.5 Rapid Controller Rotation

Rapid controller rotation is a gameplay challenge that involves the player spinning the controller or handheld device as quickly as possible in the given time limit. As with the rapid analog stick rotation, the direction of rotation is irrelevant to the successful completion of this challenge type. Mario and Sonic at the Olympic Games for the Wii tasks players to hold their Wii remote vertically and rotate it to complete the "Hammer Throw" event [182]. Mario and Sonic at the Rio 2016 Olympic Games for the Nintendo 3DS also features the "Hammer Throw" event, but uses the built-in gyroscope and requires the player rotate the entire handheld console [184]. There are many different games that use controller rotation as a mechanic, such as WarioWare: Twisted! [103], Kirby Tilt'N'Tumble [83], and Kirby: Nightmare in Dreamland [82], but it is often used as a navigation or movement mechanic. Though identical psychomotor skills are used in each, the difference in intended play methods and thus cognitive skills demonstrates that they are distinctly different challenges.

1.6 Rapid Controller Shaking

Rapid controller shaking is a gameplay challenge that involves the player making small jerking motions in order to move the controller, or handheld device as quickly as possible. The direction or method of shaking is irrelevant in this task — whether the player decides to shake the controller up and down, left and right, or some degree in between, all are registered as shaking input. The importance is placed on the size of the physical motion used to complete this challenge type. More recent entries in the WarioWare series of games use this challenge type frequently. In WarioWare: Smooth Moves several mini-games instruct the player to shake the controller. These include: Runner 's High, Come to Poppa, A Tale of One Kitty, and It 's a Wrap [102]. Shaking the controller also enacts various actions in many other types of games. Donkey Kong Country: Tropical Freeze also includes rapid controller shaking as a gameplay element, using it to enact ground pounds, roll attacks, and Kong Rolls [174]. In Donkey Kong Country: Tropical Freeze this only becomes a challenge when playing with one of the Wii remote controller schemes. The Last of Us also features this challenge through the inclusion of the flashlight. In the game the player is given a flashlight that needs to be recharged by shaking the controller up and down repeatedly [134]. The rise of accelerometers in controllers and other handheld gaming devices has allowed challenges of this type to flourish. Public opinion on the effectiveness and longevity of this and all other motion control challenge types is varied [255, 256, 250, 244], while professional opinion is almost non-existent outside of the games journalism field [56]. Often, public opinion skews to the side of dislike towards this type of challenge and all other challenge types that use motion controls. Even though opinion is unfavourable with the "hardcore" gamers demographic, the general widespread acceptance of these control schemes implies it is unlikely that these challenge types are going to disappear.

Chapter 2

Reaction Time Challenges

Reaction time challenges as defined by Adams, test the player‘s ability to react to events [10]. This definition gives us insight into the significant difference between reaction and speed challenges. In reaction challenges players undergo a perception-reaction loop, and taxing this process is the main cause of difficulty. Specifically the player must perceive an in-game event, understand what the event means in their current game context, plan an action which responds to this event, and execute that action. This entire process of perceive, plan, and react (reaction processing loop), is handled by the player in a matter of nano-seconds, allowing for a smooth gameplay experience. This is entirely different from the way that a player handles a speed challenge, which requires no planning.

Reaction time challenges are more abundant in contemporary games in comparison to their speed counterparts. Through examining various games it was found that the reaction time challenges used in video games can be divided into three categories: combat-based reaction challenges, movement-based reaction challenges, and quick-time events. Each category is host to a variety of subcategories that differ from each other based on the psycho-motor and cognitive processes used.

2.1 Simple Reaction Time (SRT)

Quick time events (QTE) are the most basic form of reaction time challenges. Prominent features between QTE definitions ([176, 110]) are: the input required to successfully complete the quick time event must be explicitly stated for the duration of the challenge, the in-game action associated with the quick time event must not be able to be accomplished through regular gameplay mechanics, and there must be an implicit or explicit time limit in which the challenge must be completed. We use these qualities as the basis for our SRT

Figure 2.1: God of War introduces and explicitly instructs players how to complete the Quick Time Events

definition: *a challenge that tasks the player to produce a specific input in response to a specific stimulus within a given time limit.*

Though we use QTE qualities as a basis, there is a one way implication between SRTs and QTEs; all QTEs are SRTs, but not all SRTs are QTEs. Examples of QTEs can be found in many contemporary games (much to the chagrin of many players [251, 252, 118]), and are often used to enhance player engagement by adding a level of interaction to cutscenes (Heavy Rain [170]) or by creating an isolated cinematic experience outside of normal gameplay (God of War (2005) [179], Far Cry 3 [227], Bayonetta [166], Resident Evil 4 [36]). QTEs tend of manifest outside of normal gameplay, thereby allowing the player to focus entirely on the challenge.

SRTs can be a normal part of gameplay. In The Legend of Zelda: The Wind Waker, the player can "parry" incoming attacks. When a player is "L-targeting" (or locked on) to an enemy, their action button will turn into a star when a "parry" can be performed. This SRT is seamlessly slipped into the combat of the game, as the player has X seconds to react before the "parry" becomes unavailable and the enemy attacks the player.

Even as a normal gameplay mechanic, the game might visually separate an SRT component from other gameplay. An example is Tidus' Overdrive, Swordplay, in Final Fantasy X [205]. When activated, the player is presented with a countdown timer, a meter with a small coloured section in the centre, and the instructions to press "X" at the "right time!". The background colours of the game become muted to have the players focus on the timer and meter. This challenge is not a QTE because it is incorporated as a regular game mechanic that the player can enact whenever they meet it's requirement (filling Tidus' overdrive meter).

SRTs can also be the entirety of the gameplay in a challenge. Quick Draw Corks (Mario Party 2 [93]) provides both auditory and visual stimuli to the players as they compete against each other to see who can press the button as soon as "Go!" appears on screen. The fastest player wins and premature presses are penalized.

SRTs are not limited in input to single button presses. Different reaction inputs can range from pressing a single button (Quickenings in *Final Fantasy*

Figure 2.2: Tidus' Overdrive, Swordplay [205]

12 [201]), mashing buttons (Joel's drowning sequence in *The Last of Us* [134]), wiggling thumbsticks (Leon shaking off a villager in *Resident Evil 4* [36]), to quickly moving controllers (escaping Raw Shocks in *Silent Hill: Shattered Memories* [48]). The time frames involved in all cases are very similar. The variety of responses shows that the focus of this challenge type is on taxing the player's attentional and perceptional abilities (ergo the reaction processing loop) rather than their motor abilities.

2.2 Combat Based Reaction Challenges

Combat like scenarios are extremely common in most commercially successful games, as it is the most typical form of gameplay. As such, players have become extremely proficient at these types of reaction time challenges and rarely realize they are reaction time challenges to begin with. All combat challenges follow the same basic format of: perceive in-game event, figure out appropriate controller input, perform controller input. At the surface level, it would seem that this definition could be applied to any instance of reaction time in a video game, and as such does not merit being a separate category. This is not the case. More in-depth and complex combat systems afford players multiple types of attacks all mapped to different inputs. This increase in button options, increases the amount of mental work the player needs to complete in the planning stage of the reaction processing loop, thus increasing the overall cognitive load. When coupled with reacting to enemy combat and all the potential moves they can make, we can see that this becomes significantly more

Figure 2.3: Final Fantasy 12's Quickening system introduces new quick time event challenge on top of normal gameplay modes.

complex than Simple Reaction Time challenges which tell you explicitly what to do.

Though this paper has chosen to label this category of challenges as "combat" it does not mean that they can only appear in a combat-based context. There exist many opportunities for reaction time challenges that require players to choose between a variety of potential inputs outside of combat scenarios. The following sections will therefore provide examples both in and out of a combat context.

On a meta-level combat is not reaction time focused. Combat in many games generally involves elements such as pattern recognition, strategy, tactics, and logistics which clearly remove it from the realm of reaction time focus (the former elements require much longer cognitive processing times than just pure reaction). We agree with this; however, combat when discussed at that meta-level is a composite challenge as the player is cognitively engaged by the primary strategic challenge of long term fight planning and physically engaged by a significantly more simple — often reaction time based — secondary physical challenge of engaging in the minute to minute combat.

Combat based reaction time challenges can be subdivided into single-input attacks, multiple-input attacks, and motion control attacks. The following sections will address the formal definitions of these challenges and provide examples of them to illustrate how they are used in contemporary games.

2.2.1 Single Input Attacks

Single input attacks are an attacking sub-challenge where the player is tasked with responding to an in-game event with a single button response. The

Figure 2.4: Legend of Zelda combat on NES. Notice that there is a limited range for combat defined by the length of the sword and direction the player is facing.

cognitive complexity of performing this task increases with the number of buttons the player has to choose from.

This type of reaction time challenge is quite abundant in contemporary video games. Examples of it in combat scenarios can be found in games such as The Legend of Zelda series. In the original Legend of Zelda for the NES [154] combat was primarily performed by the player swinging their sword at an enemy. There was only one button that swung the sword, and the player needed to make sure that the enemy they were trying to defeat was within range of their swing. Newer entries in this series use the same principle. The Legend of Zelda: A Link Between Worlds even goes so far as to use exactly the same mechanic [145].

These games offer a variety of options to players through the other attack items. Each item operates on the same mechanic; once it is equipped it only takes a single button input to use them, and they are only effective within a defined range or in a particular scenario. Other games that use the same principle in combat are The Witcher series of games [40, 41, 42].

Shielding, a form of blocking in the *Super Smash Bros.* series of games [21, 22, 84, 85, 87], is another example. The shield is mapped to a single button, and it shrinks and weakens over time (the precise amount varies by game [262]). The shield can also be broken by powerful attacks, which causes the player to become stunned. In order to be effective, the player must perceive what attack is coming, whether it can be blocked, and whether their shield will survive the hit. Shields also have a regeneration rate (which varies by game), adding an extra piece of cognitive information to the planning part of the reaction processing loop (i.e. is my shield available?).

Figure 2.5: Peach shielding against Pikachu's ranged attack [22]

Figure 2.6: Hot Jump Rope mini-game in Mario Party 2. The speed of the flaming rope's spin changes as the challenge goes on.

An example out of combat is the Mario Party 2 mini-game "Hot Rope Jump", which simulates jumping rope [93]. The player must watch the rope, and time their jump accordingly. This mini-game is a reaction time challenge due to the shift in timing of the jump rope; it does not follow any kind of set pattern, so the player cannot infer the rhythms and patterns of its movement, therefore making them rely solely on their perception and reaction time. The planning element of this challenge is in choosing to perform a short or long jump — short is a quick press of the "A" button, while long is a "long" press. This decision needs to be made in relation to the speed of the rope and size of the fire. A long jump coupled with a quickly spinning rope could result in disaster as the player lands without enough time to jump again.

2.2.2 Multiple Input Attacks

Multiple input attacks are an attacking sub-challenge where the player is tasked with responding to an in-game event with a set or series of button presses. Though very similar to single-input attacks the coordination of actions adds new complexity to the challenge that needs to be accounted for.

Figure 2.7: Dodging in Super Smash Bros. 4 requires the player to read their opponent's move and decided how to dodge.

As coordinating button presses is more physically demanding than single button presses, it is often the case that the options available to the player are fewer than with single input attacks. This type of challenge is often found in video game combat as a way to perform a sort of finisher move, not normally accessible through regular combat.

Dodging, countering, and blocking can also be found in the form of multiple input attacks. This happens when the player is given higher degrees of control over these actions. Recent entries in the Super Smash Bros. series of fighting games showcase examples of multiple input combat based reaction challenges in their dodging and countering mechanics. Dodging in Super Smash Bros. 4 can be performed in multiple ways, but all are defined by the same set of inputs: directional input and the button mapped to the shield (L or R) [21]. Countering works the same way but with the shield button replaced with the B button. For both of these potential actions there exists a window of efficiency in which the action is useful. This window is directly tied to the oncoming actions of the opponent player. This relationship between the usage of these actions and the attacking actions of the opponent player are what classify these as reaction based challenges; the players must react to their opponent‘s move for these actions to be completed successfully.

2.2.3 Motion Control Attacks

Motion control attacks are an attacking sub-challenge that relies on players responding with a preset motion to an on-screen event. While mechanically identical to single input attacking challenges, motion control attack sub-challenges differ in terms of pyscho-motor skills used. To reflect the similarities between motion control and single input challenges, it is important to consider the most recent additions to the Legend of Zelda franchise — particularly The Legend of Zelda: Skyward Sword [144]. In Skyward Sword, the principles behind basic combat are the same as previous examples: wait for an enemy character to be in range of your attack, and attack when they open themselves to damage. Though previously done through button input, Skyward Sword requires the player to make a swinging motion with their Wii remote in order to make an effective attack.

Motion control attack sub-challenges rarely occur outside combat scenarios. In the cases that they do, they often fall under the category of simple reaction time sub-challenges that will be explained further on. As such, there is no listed example of motion control attack sub-challenges outside of combat scenarios.

2.3 Movement Based Reaction Challenges

Movement based reaction challenges in essence tasks players with safely overcoming a stationary obstacle using navigational mechanics only. This means that in-game activities such as jumping between platforms or over obstacles, sliding under or through obstacles, crouching in order to avoid obstacles, and any extensions or variations on using movement to overcome an obstacle are all covered under this challenge. This challenge can appear in a variety of contexts depending on the nature of the gameplay. This challenge comes in two broad categories: fixed-time and variable-time.

There are two general distinctions between movement based reaction challenges and combat based reaction challenge. Firstly, movement based reaction challenges often have fewer input options than combat based challenges. This means that movement tends to have a fixed, one-to-one button mapping. The other distinction is that the actions associated with movement based reaction challenges tend to not be contextually dependent. Whereas combat based reaction challenges exist in relation to an enemy or animation, movement based challenges are their own entities that can allow players to control how the challenge is approached.

2.3.1 Fixed Time

In fixed-time jumping challenges the game imposes a time limit on the player to clear the obstacle. The most common examples are endless runner games like Bit.Trip Presents Runner 2 [72], and Temple Run[100] where the camera is constantly moving the player forward so obstacles approach the player at a constant speed. This means that the player will be cycling through the steps of the reaction processing loop many times, and in quick succession. Therefore, the player must allocate a greater amount of attentional resources in order to successfully enact the reaction processing loop. Auto-scroller segments of games, such as the Gummi Missions in Kingdom Hearts [200] and Kingdom Hearts 2 [204], Dives in Kingdom Hearts 3D: Dream Drop Distance [202], and

Figure 2.8: Runner 2 places obstacles within short in game distances so as to design challenges that are taxing on player's reaction time.

Donkey Kong Country: Tropical Freeze [174], are textbook examples of this challenge.

The difficulty of this challenge is determined by the spacing of obstacles, which effectively shortens the time the player has to process and react. In the first level of Runner 2 there are 6 seconds between the start of the first level and the player's first encounter with an obstacle. Though the encounter takes about 6 seconds from the beginning of the game, the player in actuality only has about 2 seconds to react from when the first enemy appears on screen before failing to clear it [72]. As the level continues, the game reduces the amount of time that the player has to react to on screen obstacles, putting more stress on the player's reaction processing loop.

2.3.2 Variable Time

For variable-time jumping challenges the player controls the speed of their movement, and by extension the amount to time they have to deal with an obstacle. The camera is tied to the player's movement, allowing for them to approach obstacles on their own terms. Often the game sets a maximum amount of time in which the player must face the obstacle, and the player has control to strictly reduce the time limit with their movement.

For example, in the first level of Super Mario Bros. [140], if the player stops moving as soon as the first Goomba appears on screen they will have 6 seconds before contact. If the player chooses to move forward at full speed the time til contact is reduced to 2 seconds. This control over timing translates to a significant change in the planning phase of the reaction processing loop. Similar examples can be found in Sonic the Hedgehog [197] and Donkey Kong Country series (Tropical Freeze[174], Returns [173]).

Chapter 3

Advanced Combat Challenges

Advanced combat challenges is a new family of challenges that we are proposing to add to Adams list. It encompasses Adams "learning complex combination moves" challenge, and a new challenge that we call "attack chaining". We propose this family of challenges to explain how challenges like "learning complex combination moves" and "attack chaining" are related. From our understanding of Adams description of "learning complex combination moves" combined with our own knowledge of the competitive fighting game community, it seemed that there was more nuance to Adams originally proposed challenge and the context it existed in.

Primarily he proposed that "learning complex combination moves" was the challenge at the heart of most fighting games. While this is not entirely incorrect, he had not considered the subtle but important differences between combos and chains. Though different in execution and appearance, these challenges are closely related. From our understanding of these challenges (as outlined in the rest of this chapter) they require the same basic set of qualities from a player (speed, timing, memory) in different amounts, and seem to bottleneck the player's performance with motor abilites (you're more likely to mess up the physical component than the cognitive component). As well, he had not considered thoroughly defining his understanding of what a combo move is. In comparing his implied definition to the one used by player communities it was apparent that there was a significant amount of definition needed to be put in place.

Before further discussing this challenge type, it is important to clarify what exactly we mean by "combination (combo) moves" as the term is used differently in the fighting game player community. Adams explains combo moves as "an especially effective or spectacular attack [that would be executed if the player] could rapidly issue a particular sequence of buttons and joystick maneuvers" [10]. The fighting game player community defines a combo as "a string of continuous moves that connect together with no time in between for the opponent to escape" [89]. This definition limits the idea of a "combo" to something that effectively stun-locks an opponent; this creates a large space of possible "combos" as many different move combinations can effectively stun-lock opponents. We believe that this definition is inherently flawed, and more closely resembles a concept that we call "attack chaining" (see Section 3.2). Our understanding of combos more closely aligns with Adams; singular attacks executed by a particular series of inputs. We elaborate on this to define combination moves as strings of input (length 2 or more) defined by the game to create a specific attack/ability. Notice this definition does not say anything about whether the move can be blocked, countered, canceled, evaded, or will result in stun-locking an opponent. We believe that a combination move refers to the combination of inputs, and as such whether it can be reacted to by an opponent is irrelevant to whether the move that was used is a combination move. An example of a combination move using this definition is Vergil's Judgment Cut move in Ultimate Marvel vs Capcom 3 which requires the player to perform the inputs shown in the image to execute the move [35]. The advantage of our definition is its scalability, as it encompasses moves as simple as the "light attack combo" in Super Smash Bros. (pressing the neutral attack button three times in quick succession) and as complicated as Squigly's taunt move "Snake Charmer" in Skullgirls (light punch, light punch, left, light kick, heavy punch). This definition still encapsulates stun-locking moves, but is not limited by them.

Learning complex combination moves is a challenge found primarily in fighting games (e.g. Super Smash Bros., Tekken, Street Fighter), and in hack-and-slash style games (e.g. Devil May Cry, DmC: Devil May Cry, Bayonetta). An example of the "learning complex combination moves" challenge is executing Dante's Cleaver move in DmC: Devil May Cry (see Figure 3.2) [136]. In order to perform this attack, the player must hold the left trigger and press Y twice in quick succession, then pause, and then press Y three more times (full combo: hold left trigger - Y, Y, pause, Y, Y, Y). A move like this showcases the qualities that Adams presents as important to this challenge type; the player must remember the combo's inputs, understand the rhythmic timing that they follow, and be quick in inputting them. These skills are further put to the test as combos become longer and more complex. In DmC: Devil May Cry, there are a variety of combo moves that incorporate switching between weapons for added complexity. The "Heavy Cleaver" is a variation on the previous Cleaver move that does this, wherein the player's new input is: hold right trigger and press Y twice, pause and release right trigger and hold left trigger, and then press Y three more times. An example of a combo move in a more traditional fighting game context is Squigly's taunt move "Snake Charmer" in Skullgirls (light punch, light punch, left, light kick, heavy punch) (see Figure 3.1) [175]. The qualities of speed, timing, and memory are still

Figure 3.1: Squiggly's Level 5 Blockbuster with the component inputs highlighted in the red boxes. Each long box is an individual combo which when strung together as an attack chain creates her Super Move.

Figure 3.2: Dante's Cleaver move in DmC:Devil May Cry as practiced in the Training room

important to the execution of this combo as it was with Dante's Cleaver move, especially considering the number of different button inputs that need to be made (whereas Dantes move only required managing two buttons, Squigly's move manages four).

3.1 Learning Combination Moves

"Learning combination moves" as a challenge is not explicitly defined by Adams, likely due to assuming that the term is self-explanatory. For clarity's sake, this challenge tests the player's ability to both remember a specific series of controller inputs, and execute them perfectly (without making a mistake in either input choice or timing). This implicit understanding that combination move execution is encapsulated in this challenge is what separates learning combination moves from memory and knowledge challenges, which are purely cognitive. Adams explains that executing combination moves (*combos*) requires 3 things: speed, timing, and a good memory [10]. From our stand point, this challenge cannot be further subdivided. It is a challenge that already evokes a very specific instance of gameplay that is easily identifiable in commercial games, so attempting to divide it into components would not be beneficial to our understanding of it as a challenge.

3.2 Attack Chaining

A related concept to "learning combination moves" is attack chaining. In his informal analysis of combos and attack chains in fighting games, game critic and indie developer Richard Terrell defines attack chains as: "a series of player actions that are all successful. Success must be defined explicitly in game terms." [213]. While a useful starting point, we believe a more explicit and concrete definition is needed. Attack chaining is a challenge in which the player must successfully complete a series of individual combat actions (attacks, blocks, cancels, counters, combos, and dodges) in quick succession, which ends when the player receiving a negative combat response (e.g. being hit by an opponent, opponent dodging an attack), the opponent is knocked out, or the player chooses to end it. Our definition of attack chains is similar to the fighting game player communitys definition of a combo. From our definition, we see that their definition of combo (series of attacks from which the opponent cannot escape) is a long attack chain which ends with the opponent being knocked out. It is important to note that this type of sub-challenge only exists in games that have a deeply developed combat system, and as such do not occur in non-combat scenarios. Attack chaining often exists in systems that also support complex combination moves. This can be a source of confusion when identifying attack chaining, as they are very similar to combination moves in combat systems. Examples of attack chains in games can be found in almost any game with real time combat where you control a single character at a time from a close camera range. Traditional fighting games and hack-and-slash style games use attack chains liberally. Across the Super Smash Bros. games there are no set combinations of moves, rather every character has individual combat actions that are tied directly to a single set of button inputs. During gameplay, players are free to string combat actions together in any order they want, as every attack action deals a set amount of damage regardless of when its used in relation to other moves. The only thing dictating move input speed is the players ability to react to the combat animations of the other player. Attack chains are also found in action oriented role playing games like Dark Souls, Kingdom Hearts, and Final Fantasy XV, and massively multiplayer online role-playing games like Black Desert Online. In these types of games, often the chain is limited to just a string of attack inputs, blocking, or dodging without being hit.

Part II

Motor Abilities

0.1 Modeling the Player

In order to understand the relationship between video game challenges and the player we need both an extensive catalogue of challenges and a well-developed model of the player. The previous section focused on exploring the former, and so this section focuses on the latter.

Modeling software users has been a topic of interest for many different academic disciplines; as such there are many models. Artificial intelligence (AI), human-computer interaction (HCI), and cognitive psychology (CP) have used models in various gaming scenarios [113, 211, 33, 45, 239, 114, 116, 115]. The focus of using models like this is to create believable human-like behaviour by modeling decision making. These models work exclusively in the cognitive domain, and often are not focused on accurate representation of the underlying processes. Games studies as a field has also generated and used player models to explain and predict human behaviour in a gaming context. The most common types of player models in game studies are player taxonomies which classify gamers into groups based on play styles and preferences. Examples include Bartle's Taxonomy [24], and Chris Bateman's DGD1 Model [25]. These types of classification are also unhelpful for our purposes because they focus entirely on the player's preferences and personality, and not on the challenge of the game or how the player interacts with the game.

From our perspective, a well-developed model of the player encompasses their cognitive and motor capabilities. The model we need should facilitate examining the relationship between the player and the game. It should describe the abilities available to the player when interacting with the game. We have been unable to find any existing model that covers both the cognitive and motor domains. Therefore, we decided we needed to create one.

0.2 Scoping

In order to develop a model which examines the cognitive and motor abilities used in gameplay, we need to know what abilities exist. Unfortunately, there is no comprehensive list of these abilities. Contemporary understanding of these processes is constantly changing as more research is done into the brain and cognition [38]. Completing our player model would require extensive research and understanding of cognitive psychology and motor development. Developing a model like this could be the undertaking of a thesis by itself, therefore we decided to limit ourselves. At this point in time, we focused on creating a limited model which captured the motor abilities of a player. This limited model is constructed with the understanding that it will be modified in the future to accommodate cognitive abilities. Having this understanding allows us to make informed decisions regarding how this model will be structured. The choice to focus on motor abilities was necessitated by partially by space limitations for this work, and partially because it was the easier of the two player components to explain and visualize for the reader. As we are presenting a novel approach to analysing both players and games, it seemed appropriate to start with the easier sections to explain. It also allowed for us to spend more time explaining and elaborate on game challenges.

This choice to focus on motor abilities instead of cognitive abilities creates some limitations in the framework presented here. Primarily this limits the types of challenges that we can discuss, as some of the listed challenges are more heavily focused on cognitive abilities. This is fine at the moment as the challenges we have focused on explaining in the previous sections are more geared towards motor abilities being the limiting factor in their play.

Chapter 1

Motor Abilities and Processes

The following chapter outlines the motor abilities and processes that we believe our player model needs to capture. The abilities and processes that we are presenting have been compiled from reading several undergraduate level textbooks on cognitive and developmental psychology. As we are not experts in these domains, our understanding is grounded in these books. Specifically we draw our knowledge from Developmental Psychology: Childhood to Adolescence, 8th edition [191], Galotti's Cognitive Development: Infancy through Adolescence [73], and Human Development: A Lifespan Approach, 8th edition [162].

We initially examined kinesiology models of motor ability, which model motion through the contraction and expansion of muscle groupings [240, 88]. Individual muscles can belong to multiple muscle groupings. Applying this muscle group model to our study of video game challenges would result in the same muscle groups being activated most of the time. This is because many different actions are controlled by the same muscles. For example, imagine a person tracing lines and shapes with a stylus on a touchscreen. This person can draw straight lines by bending their wrist, and can draw circles by rotating their wrists. Though these two actions (bending, and rotating) are different, the muscles that control the wrist's movements are not. This distinction between actions rather than muscles is extremely important for video game challenges. So we concluded that this model of motor abilities did not suit our needs. We then turned to developmental psychology because it concerns itself with normative development (typical patterns of development in human beings).

As we are trying to model the set of human abilities, we assume that there must be a normal set that the average human will acquire over their lifetime. This aligns with developmental psychology's underlying assumption, so we are able to leverage their research to compile a list of motor abilities available at adulthood. Since developmental psychology studies normative development, we could similarly compile a list of the available motor abilities of any age group based on the skills acquired up to that age and factoring in skill degradation for ages past peak physical prime. This allows us to see which challenges bar certain age groups from playing based on motor abilities. For example, a game that requires a player to trace a shape with a stylus will be incredibly difficult for a preschool aged child who does not have the manual dexterity to hold a pencil steadily. Similarly a game that requires quick arm and leg movements may be too strenuous for an elderly individual.

Developmental psychology has 6 major theoretical viewpoints (psychoanalytic, learning, cognitive-developmental, information-processing, evolutionary, ecological systems) [191] of which we are focusing on cognitive-developmental (CD). This is because the CD viewpoint is the most concerned with motor skills. All theories from the CD view distinguish between fine and gross motor skills [191, 162]. This distinction is a widely agreed upon categorization of motor skills and will enable us to analyse different games based on the types of skills used to interact with it (e.g. button pressing vs arm movements). Though it is easy to distinguish between fine and gross motor skills, no comprehensive list exists. This makes sense because the space of possible human actions is virtually infinite. At this stage we are attempting to make a working list of motor abilities for a video game context. This space of possible actions is significantly smaller due to limitations on existing input devices.

In order to begin compiling a list of motor skills for a video game context, we examined various motor assessment tools used in developmental psychology. These were the Bayley Infant Development Scale, 3rd edition (BIDS), the Peabody Motor Development Scale, 2nd edition (PMDS), and Noller and Ingrisano's comprehensive assessment tool (Noller). BIDS and PMDS were selected due to their widespread use and popularity. Noller was selected because it is the result of an extensive study of all available assessment tools (including BIDS and PMDS) which compared the skills listed and extrapolated the commonalities. The motor skills listed for each tool are outlined in Table 1.1.

Motor Skills	**BIDS**	**PMDS**	**Noller**
	Movement of limbs and torso	Reflexes	Stacking tower of cubes
	Static positioning	Stationary gross motor skills	Transfers cubes between locations
Gross	Dynamic movement	Locomotion	Reach and grasp of cube
	Balance	Object manipulation	Copies cube structures
	Motor planning		
	Prehension	Grasping	Tracking across mid-line
	Perceptual motor integration	Visual-motor integration	Tracking in 180
	Motor planning and speed		Turning towards sound
	Visual tracking		Radial digital grasp of cube
Fine	Reaching		Pincer grasp
	Object grasping		Copies a drawing of a square
	Object manipulation		Static tripod grasp on crayon when copying square
	Functional hand skills		Placing shapes on a form board
	Responses to tactile information		Finger position

Table 1.1: Comparison of Motor Skills Listed by Assessment Tools

We found from these tools that motor skills are exclusively discussed as tasks (e.g. stacking blocks). This is familiar to those in HCI research as the major predictive models (Fitt's Law, Hick's Law, Steering Law) explain interactions using similar language (e.g. pointing tasks, text-entry tasks, selection tasks). Using tasks to describe interactions with video games lands us in a similar problem to using kinesiology models; that actions exist where the tasks are superficially identical, but measurably different. For example, the actions available on a standard controller (i.e. Playstation Dualshock 4) include pressing face buttons and pulling trigger buttons. In a purely computer science context these inputs are identical (both are button presses), however the movements made by the player's hands are measurably different due to how the controller is held (pressing with the pad of your thumb versus pressing with the middle of your finger). We believe that these differences are important to capture in our framework, as it potentially impacts the difficulty of a game. For example, mashing a face button requires much less movement than rapidly pulling a trigger this could contribute to the difficulty of a button mashing challenge depending on which button is meant to be pressed. Though we believe these differences are important, at our current stage of research the size difference between some movements is small enough that we are not concerned by them. For example, when a player is pressing a face button on a standard controller their thumb needs to extend to reach the top button in comparison to pressing the bottom button. We have decided that differences of that type are not going to be individually examined at this time. Rather, we propose that the motor ability information presented in our final framework be viewed with a to be specified margin of error to accommodate for these potential differences.

Since focusing on tasks does not suit our needs, we would like to find the movements made by players when completing a task. Focusing on movements allows us to capture the differences between superficially similar tasks, and see whether tasks that are superficially different use the same underlying abilities. Defining motor abilities by discrete movements also allows us to easily determine the accessibility of a game to different demographics based on physical ability. Isolating why a player failed a task is more complicated than isolating why a player failed to complete a discrete movement. This is because tasks involve more than just the motor response of the player; they include levels of cognitive processing and planning. Even a task like button pressing relies on perception and attention to successfully complete it. We want to be able to isolate motor responses as the point of failure, and so examining movements seems the best way to go.

In order to begin cataloguing the types of movements specific to interacting with video games, it would be best to figure out the various tasks for that context. We can analyse those tasks to see which movements they use, and from that generate our list of interaction actions. In order to begin listing video game controller interaction tasks, we need to explore the different types of controllers that exist.

Chapter 2

Input Methods

There are many forms of video game inputs available; a number that will only continue to grow as new technology emerges. If we were to attempt to list them all, we would undoubtedly miss some form of controller. Instead, we choose to focus on the most common forms of inputs for video games. These are: the standard controller (e.g. Xbox One controller, Playstation controller), the handheld motion controller (e.g. Wii Remote, Playstation Move), the full body motion controller (e.g. Kinect), smartphones, handheld consoles (e.g. Nintendo 3DS, Playstation Vita), keyboards, mice, fight sticks (arcade style controllers made for fighting games), mat controllers (e.g. DDR mats, Wii Fit Board), and specialty simulator controllers (e.g. driving simulator controller, flight cockpit controller, instrument controllers). By covering these inputs, we can generate a list of actions that will apply to a very broad set of games. The following sections will outline the types of actions available on each type of input.

As we discuss controllers, we need to account for how the player holds or positions them. Though two controllers may have the same features (e.g. face buttons, trigger buttons, thumbsticks), the way that they are held and positioned dictates how the player will interact with them. To account for this we will assume that players are holding controllers in the intended way. For many controllers the intended way is implied by its form (e.g. placement of handles and buttons). However, we choose to cite the available patents for our examples to formally explain the designer's intentions. We acknowledge that a number of players will not adhere to the intended holding style and yet may still be successful in completing and competing in video games [71, 91]. For our purposes we cannot catalogue all possible controller techniques, so we choose to focus on the method that the majority of players will use.

2.1 Standard Controller

The *standard controller* describes the type of controller used with most video game consoles. It is intended to be held in the player's hands with their thumbs resting on the joysticks [131, 32]. Controller design is fairly standardized between modern consoles with all having four face buttons, two joysticks, a d-pad, two trigger buttons, and two shoulder bumper buttons. Figure 2.1 shows the Xbox One and Playstation 4 Dualshock controllers side-by-side to see the similarities. This standardization between controllers allows for us to describe the motions used to interact with one controller and extrapolate it as general motions for all standard controllers. As well, controllers have only added functionality through the generations. This means that by examining the current generation of console controllers we know that any previous controllers will just use a subset of the actions we have listed here. We base our listing of

Figure 2.1: Xbox One and Playstation 4 controller

possible actions on the Playstation 4 Dualshock controller (Figure 2.2). This is because it has the most interaction features incorporated into its design (adding a touchpad and motion sensor LED into the controller on top of the usual components). The actions available on the Playstation 4 controller are: pressing buttons, bumping the shoulder buttons, pulling the trigger buttons, moving joysticks, pressing joysticks, swiping on the touchpad, pinch-to-zoom on touchpad, pressing on touchpad, and moving the controller.

2.2 Handheld Motion Controller

Motion control describes an input technique used by several forms of input device, wherein the orientation and acceleration of the input device is monitored by a fixed sensor so its movement (and by proxy the player's movement) can be tracked. The *handheld motion controller* is the most common form of motion control input device. It is typically shaped like a wand, and held in the dominant hand with the thumb resting on the front of the controller and the rest of the fingers wrapping around it. It can also be held horizontally

Figure 2.2: Playstation 4 controller showing off all points of interaction

Figure 2.3: Nintendo's Wii Remote and Playstation Move Controllers

between the player's hands with their left thumb resting on the directional pad, and their right thumb on the buttons. Commonly available examples are the Nintendo Wii Remote and Playstation Move controllers seen in Figure 2.3.

We base our list of possible actions on the Wii Remote (Figure 2.4). The possible actions available on the Wii Remote when held vertically in one hand are: pressing a face button, pulling the trigger button, swinging the controller, shaking the controller, pointing the controller, drawing shapes with the controller, thrusting the controller, tilting the controller, and flicking the controller. When held horizontally between the player's hands, the possible actions available are: pressing face buttons, shaking the controller, and tilting the controller.

Orientation is a recurring issue with many controller types. Though only briefly touched on here, Chapter 3 examines more closely how the orientation of these devices affects their possible actions, and how these actions are performed. We believe from our understanding of the underlying motor abilities used when completing these actions, that the defining characteristic in actions is not actually the orientation of the controller — but whether it is used with one or two hands. This point will become clearer as this paper continues.

Most handheld motion controllers have accompanying accessories that add additional actions. The most common accessory are ones that add a joystick to the controller. Figure 2.5 shows the Wii Remote and Playstation move with their associated accessories. These accessories can only be used when

Figure 2.4: Nintendo's Wii Remote

the handheld motion controller is in the vertical orientation. Though these attachments may also feature more buttons, the only new action is in moving the joystick.

2.3 Full Body Motion Controller

Full body motion controllers use the player's body as the input. This is done through external camera-like sensors which capture the player's movements. The most well known foray into this technology is the Microsoft Kinect for Xbox One and Xbox 360 2.6, as such we use it as our reference for understanding this controller type. The Kinect measures motion through skeleton-based pose estimation based on joint position [177].

Though this type of input theoretically covers the entire range of human motion, technological limitations restrict it to coarse movements (i.e. gross motor abilities). Based on the Kinect, possible detectable motions include arm movements (bending at the elbow, waving, positioning), leg movements (e.g. bending at the knee, kneeling, positioning), and body positioning (e.g. standing, bending at the waist, turning). This is an important point regarding full body motion controllers, as they appear to have higher dimensionality (can accommodate more inputs), but in effect are extremely limited. Generally, we see they pattern match well (are you making the same pose as an on screen

Figure 2.5: Nintendo's Wii Remote and Playstation Move Controllers with Joystick accessories

character), but cannot accommodate common actions that often require multi-dimensional input like moving a character. The Kinect has never successfully allowed the player to control the movements of their avatar, often relying on games being "on-rails" to move the avatar between areas of action automatically. When they attempted to allow the player to control the avatar's motion, they attempted to add "movement actions" such as moving the player's arms behind them to make their avatar sprint in Kinect Star Wars [212]. This type of motion just accelerated the movement on the rails and did not permit the player actual control.

Though this hardware is becoming more popular for use in other industries, it has not been successful in a video game context [267].

Figure 2.6: Microsoft Kinect for Xbox 360 and Xbox One

2.4 Smartphones

Smartphones and tablets are multipurpose computing devices that do not have any external input device attached (e.g. keyboard, mouse). As the available hardware has advanced, there has been a rise of mobile games. This has made smartphones and tablets the most common form of gaming device [126]. The multipurpose nature of mobile devices creates a problem when trying to identify possible actions. Specifically, that any potential input action (e.g. tapping on the screen) may use different motions depending on how the device is being held and oriented. To illustrate this issue, take a look at the game Two Dots. This game can only be played with your phone oriented vertically (as indicated by the text on the GUI), and the only interaction is connecting dots by drawing on the screen. Even still there are at least three ways in which the player can hold their device (Figures 2.7, 2.8, 2.9). Though all three holding positions are viable, the manner in which the player completes the interaction task is different (single thumb, both thumbs, single finger). These different methods are subject to their own limitations (e.g. range of motion of thumb while holding an object), which may inevitably affect the player's ability to complete certain tasks.

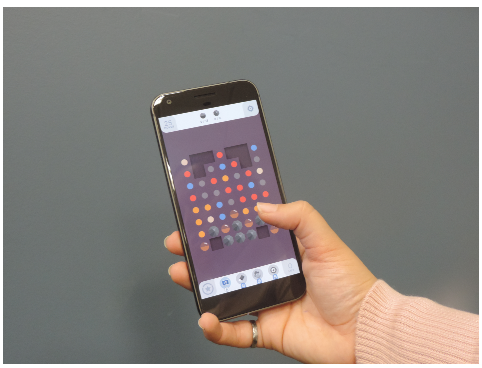

Figure 2.7: Two Dots played with one hand supporting the phone and interactions made by the same hand's thumb

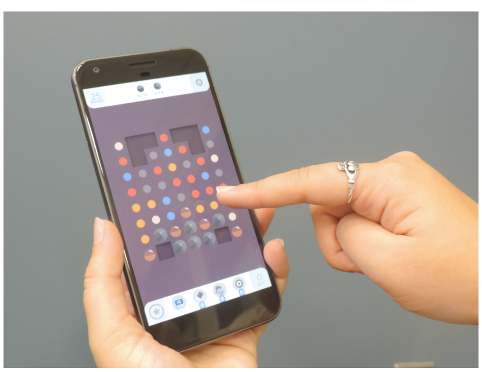

Figure 2.8: Two Dots played with one hand supporting the phone and interactions made by the opposite hand's finger

For the purposes of our research, we have decided to analyze only the intended method of holding (see subsection 2), but deciding on what that is for smartphones is extremely difficult. Choosing a single way to hold it, or

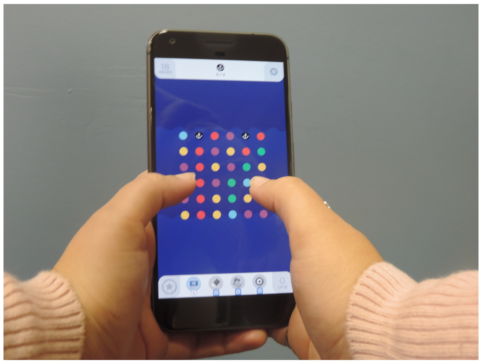

Figure 2.9: Two Dots played with two hands supporting the phone and interactions made by the thumbs

orient it would be largely arbitrary; however, because we will not be able to cover all holding methods, we choose to explore two of these methods with the belief that they will give us a representative sample of the types of motions used to interact with smartphones. The two methods we will be looking at are: supporting vertically oriented phone with one hand (Figure 2.11), and holding a horizontally oriented phone with both hands (Figure 2.12). Using the Google Pixel XL (Figure 2.10) as a base, we begin to examine possible actions for each of these orientations. We chose the Pixel XL because it has a wide range of features common to most smartphone and tablet devices; therefore the actions that we find for this device should be applicable to most others on the market. Though the form factor between different models of phones and tablets are becoming more streamlined, the existing differences could potentially impact how the user interacts with their device. For example, a user will use a different grip method to support a 12.9" iPad Pro versus the 9.7" version. Even between the Pixel and Pixel XL, the larger screen size causes the range of motion of a player's thumb to reach different parts of the screen. However, an in depth analysis of the measurable effects of form would be out of scope for our current research. We therefore save a comparison such as this for future research, with the hope that quantifying these differences can improve our understanding of the interaction limitations of smartphones and tablets.

In the vertically oriented position, we find the following actions: shaking the phone, tilting the phone, tapping the screen, drawing on the screen (includes short drawing actions like swiping and flicking, as well as long drawing actions like tracing), and speaking into the microphone. The first two actions (shaking, tilting) are performed by the hand supporting the device, while the tapping and drawing actions are performed by the free hand. Speaking does not require hand motions, but the player may choose to bring the device closer to their mouth to have the input be captured more precisely.

In the horizontally oriented position, we find the following actions: shaking

Figure 2.10: Google Pixel XL Smartphone

Figure 2.11: Google Pixel XL held in vertical orientation with one hand

the phone, tilting the phone, tapping the screen, drawing on the screen (short drawing actions like swiping and flicking, tracing), and speaking into the microphone. The shaking and tilting actions are performed with both hands, while the tapping and drawing actions are performed exclusively with the player's thumbs. This difference in the tapping and drawing actions (thumb use versus finger use) is significant due to the difference in range of motion between the thumb and the finger. As such, the actions are less exaggerated, and cannot cover the whole screen.

As previously discussed in Section 2.2, the defining feature here is whether the phone is held in one hand, or between the player's hands. The differences in the actions available in these two orientations is more extensively discussed in Chapter 3.

One of the issues that can arise when playing games using mobile device is the fatigue and stress on the player over long play sessions. As smartphones exist in many different sizes, they have significantly different weights. With the way that a player is meant to hold the phone, extended play sessions could potentially impact the player's ability to comfortably interact with the device.

Figure 2.12: Google Pixel XL held in horizontal orientation with two hands

Though this is possible, we have not incorporated it into our framework due to the limitations we are imposing on play sessions (see Chapter I section 0.3). Most mobile games have a short play session time [44, 192, 2, 164]. As such, the fatigue and stress factors should not significantly impact player performance at the level we are examining it.

We have only discussed here the movements and actions associated with the most common form of mobile game. As the hardware to support them becomes more accessible, VR and AR mobile gaming is becoming more apparent in the mobile gaming market. Though these types of games are becoming more common, we have decided to bypass examining them for reasons stated in Chapter I, Section 0.3.

2.5 Handheld Consoles

Figure 2.13: Nintendo Switch, Nintendo New 3DSXL, and Sony Playstation Vita

Handheld consoles are mobile devices whose primary function is gaming. Examples of these range from the Nintendo 3DS and Playstation Vita, to larger units like the Nintendo WiiU and Switch. The defining characteristic of these devices are their built-in screens and physical controls, which allow for portability. In this sense they are almost a hybridization of smartphones and standard controllers. An important distinction between handheld consoles and smartphones is that the former can only be used in a single orientation, and therefore only has one intended method for holding. Handheld consoles are intended to be supported by both hands, with the player's thumbs resting on the face of the device. Occasionally, the player may be required to support the device with one hand, while the other hand manipulates a stylus. Using the Nintendo 3DS XL as our guide, we find the following possible interactions: pressing face buttons, bumping shoulder buttons, moving joysticks, shaking the console, tilting the console, tapping on the screen, drawing on the screen, speaking into the microphone, and making facial expressions at the camera.

As with smartphones, we have chosen not to explore the effects of weight and fatigue over long play sessions. Unlike smartphones, games designed for handheld consoles are intended for longer play sessions. That means the effects of the consoles' weight and resulting player fatigue would be more pronounced. However, since we are taking such a rudimentary look at comparing gameplay challenges and gameplay motor abilities, it would be out of scope to address this issue at present. Instead, we assume that the player is only interacting with the device for the length of the individual gameplay challenge; thus, limiting the effects of this issue for most challenge types. We believe that future work should take a closer look at the effect of device weight over longer play sessions for different challenge types to better refine the work we are presenting here.

2.6 Keyboards and Mice

The keyboard and mouse pairing is the most ubiquitous form of input device. The majority of PC games use this setup as the default control scheme as it is the one form of input developers can guarantee a PC gamer will have. Keyboard and mouse use has even been an area of debate in online gaming communities, with popular opinion favouring this control method over standard controllers for a variety of game scenarios [249, 253, 3]. Unlike the other devices on this list, the fact that keyboards and mice are so closely related means that we need to analyse their interaction capabilities individually and as a unit.

Keyboards are a device designed for text entry, but whose gaming purposes range from controlling avatar movement, to executing in-game actions. Though there are many regional key mappings (e.g. QWERTY, AZERTY, Dvorak), the physical form is extremely standardized. It is important at this point to clarify that we are only discussing physical keyboards (not on-screen soft keyboards). Within the realm of physical keyboards there are several different types (e.g. mechanical, rubber-dome, membrance). Though consumers may find a difference in performance due to elements like feedback [129, 57, 209], there is no motion difference between pressing keys on any type of physical keyboard. Similarly, there is no motion difference in pressing a key on keyboards with different regional key mappings (e.g. QWERTY, AZERTY). Using the Corsair K70 LUX Rapidfire mechanical keyboard as a reference (Figure 2.14), we find only one form of input: pressing buttons.

Figure 2.14: Corsair K70 LUX Rapidfire Mechanical Keyboard

Mice are pointing devices which have a wide variety of uses in games. Depending on the challenges present, mice can be used for actions like controlling the camera, directing characters, navigating menus, using weapons/abilities, changing weapons, interacting with the in-game environment, and many more. Though at first glance mice seem to be as standardized as keyboards, contemporary mouse designs differ enough that we feel a distinction should be made between a few categories. For the purpose of our research, we limit ourselves to discussing optical mice, and modern trackball mice (where the trackball is on the side of the mouse). We acknowledge other mice designs, such as the vertical ergonomic mice, however we limit ourselves to these two forms due to their widespread acceptance in the public sphere. The optical mouse, for which we are using the Logitech G400s optical gaming mouse (Figure 2.15) as our standard, allows for the following possible actions: moving the mouse, clicking the buttons, scrolling the scroll wheel, and pressing the additional function buttons. The modern trackball mouse, for which we use the Logitech M570 wireless trackball mouse (Figure 2.16) as our example, has the following possible actions: moving the mouse, clicking on buttons, and scrolling the scroll wheel. Though the actions between the two seem identical, the design differences cause them to be performed differently. In examining ”moving the mouse” as an action, we see that when using an optical mouse this is accomplished through a combination of wrist and arm movements. In comparison, the same action on a modern trackball mouse is performed exclusively through thumb movements.

As we have analysed these inputs individually, we now need to discuss them together. It’s apparent that since nothing changes in the design of the

Figure 2.15: Logitech G400s Optical Gaming Mouse

Figure 2.16: Logitech M570 Wireless Trackball Mouse

individual devices, the interaction actions (e.g. pressing a key, moving the mouse) do not change. However, by using these devices in tandem there is a larger cognitive load on the individual, and greater demands on their motor coordination. Depending on the challenge type, this could be a limiting factor for a games' accessibility to a wider audience. Consider the reaction-time based challenge aiming for combat, which is the central challenge in all first person shooter games. Succesfully completing this challenge using a keyboard and mouse relies on significant amounts of manual coordination. Even games whose challenges rely on strategy, tactics, and logistics (e.g. League of Legends, Defense of the Ancients 2) require significant amounts of manual dexterity, especially at high level play. Expert level players of games like Starcraft II measure their actions per minute (APM), with most professional level players averaging numbers of 500-600 APM [117]. As discussed in Section 0.3, we choose to focus our research on average level play, not expert level play. Therefore future discussion regarding the tandem use of a keyboard and mouse will be limited to only instances where its effects are obvious in normal play.

2.7 Specialty Controllers

Specialty controller is the term we are using to describe a variety of non-standard input devices. Specialty controllers are designed to be used in specific gaming contexts, and so are limited in their use. Examples of specialty controllers include: racing wheels, instruments (e.g. Rock Band guitar and drums, DK Bongos), dance mats (e.g. Dance Dance Revolution pads), cockpit simulators, and many more. Though there exist significant markets for these types of controllers, they are mainly used for enthusiast and expert level player inputs. As such, they are currently out of the scope of our research. However, for the sake of cataloguing some basic information regarding these forms of control, we will briefly touch on some of the most common forms. These are: fight sticks, mat controllers, and simulation controllers.

2.7.1 Fight sticks

Fight sticks are a type of controller optimized for the fighting game community. They are meant to emulate old fighting game arcade cabinet interfaces. Fight sticks are extremely large, and so must rest on the player's lap or on a table. The player's hands rest on the joystick and the face buttons. To analyse fight pads we chose to use the Mad Catz TE2+ as our reference point (Figure 2.17). From this we see only two possible actions: joystick movement and button pressing.

2.7.2 Mat controllers

Mat controllers allow players to use their feet as an interaction method. Though originally only used for rhythm and dance games in arcades (e.g. Dance Dance Revolution), home versions have broadened their use for gaming. There are two types of mat controllers: *pressure sensitive* versions, like the Wii Balance Board (Figure 2.18), and *standard* ones, like the hard and soft DDR mats (Figure 2.19). Pressure sensitive mats are able to be used in more gaming contexts than the standard version. For example, the Wii Balance Board has been used in sports simluation games to simulate snowbaord and skateboard riding [217, 150], as well as in exercise games as a yoga mat to monitor how well the player is performing their poses [150]. In contrast, standard mats are exclusively intended for rhythm and dance games. An important thing to

Figure 2.17: Mad Catz TE2+ Fight stick

Figure 2.18: Wii Balance Board

note is some players have rigged standard mats to use as input for other game types [70, 27]. This use of mat controllers is exclusive to expert level players looking to increase the difficulty of a game. As such we are not considering these instances of using mat controllers as input, since it would be out of our scope.

2.7.3 Simulation Controllers

Simulation controllers are input devices designed to simulate the actions of a specific task. As such, the size and shapes of these controllers range from small instruments like the bongo controllers for Donkey Konga (Figure 2.20), to extremely large and detailed recreations of airplane cockpits. The motions made to interact with simulation controllers are virtually identical to interacting with their non-video game counterparts. The arm and hand movements used to interact with the DK bongo controllers are the same as those used to interact with regular bongos. Though the motions are identical, use of these

Figure 2.19: Dance Dance Revolution Arcade Machine with Hard Mat Controller

controllers would not necessarily make you proficient with their real world counterparts. The DK Bongos operate as a set of buttons (meaning interaction is only registered in certain places), real world bongos are sensitive to interaction all over. However, playing a flight simulator game with an accurate simulation controller have been shown to make a person more proficient with flying a similar plane [90]. The numerous simulation scenarios that we would need to cover in order to adequately explain these inputs is unrealistic to explore at this time in our research. If we were to attempt to do this, we would inevitably overlook some form of simulation controller as new technology constant emerges to make more simulation scenarios possible. Therefore we will not be discussing simulation controllers in any further detail.

Figure 2.20: Donkey Kong Bongos

Chapter 3

Motor Abilities Compiled

From the previous chapter, we now have a list of possible actions for different interaction devices (see Tables 3.1, 3.2). The purpose of compiling this list was to begin examining the motor abilities used when interacting with video games. However, the list we have is both task-oriented and hardware dependent. We would prefer to have a hardware independent list of abilities such that our framework then focuses exclusively on games, and not on the specific implementation of games. As such, we begin to analyse the listed actions for their movement components. We do this by eliminating references to the hardware from the actions. We then end up with the following list of motions (see Table 3.3).

Standard	**Handheld**	**Full Body Motion**	**Smartphones / Tablets**
Pressing buttons	Pressing buttons	Arm movements	Shaking the phone
Bumping the shoulder buttons	Pulling trigger buttons	Leg movements	Tilting tp
Pulling the trigger buttons	Swinging tc	Body positioning	Tapping on the screen
Moving thumbsticks	Pointing tc		Drawing on the screen
Pressing thumbsticks	Shaking tc		Swiping on the screen
Swiping on touchpad	Drawing shapes with tc		Flicking on the screen
Pinch-to-zoom on touchpad	Thrusting tc		Speaking into the microphone
Pressing on touchpad	Tilting tc		
Moving tc	Flicking tc		
	Moving thumbsticks		

Table 3.1: Compiled list of video game interaction devices (controllers) and the actions possible on each of them. The following abbreviations are used: tp = the phone; tc = the controller

Handheld Consoles	**Keyboards and Mice**	**Fight Sticks**	**Mat**
Pressing buttons	Pressing buttons	Move joystick	Pressing buttons
Bumping shoulder buttons	Clicking buttons	Press buttons	
Moving thumbsticks	Moving the mouse		
Shaking the consoles	Scrolling the scroll wheel		
Tilting the console	Pressing additional functional buttons		
Tapping on the screen			
Drawing on the screen			
Speaking into the microphone			
Making facial expressions at the camera			

Table 3.2: Compiled list of video game interaction devices (controllers) and the actions possible on each of them.

Motor Actions	Hardware Context
Pressing	Standard controllers, handheld motion controllers, handheld consoles, keyboards, fight sticks, mat controllers
Bumping	Standard controllers, handheld consoles
Pulling	Standard controllers, handheld motion controllers
Moving	Standard controller, handheld motion controller, full body motion controller, handheld console, fight sticks, mice
Swiping	Standard controller, smartphones/tablets
Pinch-to-zoom	Standard controllers, smartphones/tablets
Swinging	Handheld motion controller
Pointing	Handheld motion controller
Shaking	Handheld motion controller, smartphones/tablets, handheld consoles
Drawing	Handheld motion controller, smartphones/tablets, handheld consoles
Thrusting	Handheld motion controller
Tilting	Handheld motion controller, smartphones/tablets, handheld consoles
Flicking	Handheld motion controller, smartphones/tablets
Positioning	Full body motion controller
Tapping	Smartphones/tablets, handheld consoles
Speaking	Smartphones/tablets, handheld consoles
Making facial expressions	Handheld consoles
Clicking	Mice
Scrolling	Mice

Table 3.3: List of possible actions a player can make to interact with a video game and the controller hardware these interactions are found on. Legend: Blue = fine motor, Yellow = Gross motor, Green = both

It is at this point that we return to the concept of fine and gross motor abilities. Fine motor skills are actions which require delicate muscular control and primarily involve small motor groups, focusing on movements made by the hands, feet, and head [162]. Gross motor skills are actions which require coordination of large muscle groups, with focus on movements of the arms, back, torso, and legs [162]. All of the motions compiled in our new list (Table 3.3) can exist at both a fine and gross motor level. For example, the pressing action is a fine motor ability when completed on a standard controller as it requires finger motions, but is a gross motor ability when completed on a mat controller as it requires leg movement and coordination. We choose to divide the motions in Table 3.3 along these lines, so as to allow us to easily examine the types of movements made when interacting with a game. It is important to note that, though we can separate these motions into fine and gross motor abilities, actions exist on a fine to gross motor spectrum and use both small and large muscle groups in conjunction to complete tasks. Our categorization of these motions as either fine or gross motor abilities coarsely reflects which side of the spectrum they fall on. We found that the majority of motions made to interact with video games are fine motor abilities. This seems to be due to the controllers available for use, the majority of which are meant to be manipulated using a player's hands. Though there was this overwhelming majority, it became quickly apparent that many of the listed motions existed as both fine and gross motor abilities. We chose to colour-code Table 3.3 to represent our categorization of these actions, with blue indicating the action is a fine motor skill, yellow indicating it is a gross motor skill, and green indicating it exists as both.

This list of motions is large, and from a naive look, seems to overlap in places. In an effort to eliminate redundant information, we began to examine these actions and determine whether any could be amalgamated. To begin we separated the full list of motions into a fine motor list and a gross motor list (see Tables 3.4 and 3.11 respectively). The following sections will examine each of these lists to determine whether any of the abilities could be considered similar enough to be condensed.

3.1 Fine Motor Abilities

Fine Motor Actions	Hardware Context
Pressing	Standard controllers, handheld motion controllers, handheld consoles, keyboards, fight sticks, mat controllers
Bumping	Standard controllers, handheld consoles
Pulling	Standard controllers, handheld motion controllers
Moving	Standard controller, handheld motion controller, handheld console, fight sticks, mice
Swiping	Standard controller, smartphones/tablets
Pinch-to-zoom	Standard controllers, smartphones/tablets
Swinging	Handheld motion controller
Pointing	Handheld motion controller
Shaking	Handheld motion controller, smartphones/tablets, handheld consoles
Drawing	Handheld motion controller, smartphones/tablets, handheld consoles
Tilting	Handheld motion controller, smartphones/tablets, handheld consoles
Flicking	Handheld motion controller, smartphones/tablets
Tapping	Smartphones/tablets, handheld consoles
Speaking	Smartphones/tablets, handheld consoles
Making facial expressions	Handheld consoles
Clicking	Mice
Scrolling	Mice

Table 3.4: Fine Motor Actions (subset of Table 3.3) with their hardware contexts

Fine motor abilities are extremely important in most game playing scenarios, as many of the input devices available are manipulated by the players hands. We believe that this reliance on hand manipulation, ends up creating a limited number of interactions possible with a game due to the limited motions that hands can perform. Therefore, we believe that the existing fine motor list could be further condensed. We begin by grouping the existing list in regards to the part of the body it uses (see Table 3.5). Here we see that multiple body parts can perform the same motion, though the results and context will be different. From here we can begin to eliminate redundancy by examining the actions available to each individual body part, and determining whether they are the same.

3.1.1 Fingers

We start by examining the actions possible by fingers. Our first impressions were to see whether pressing, bumping, pulling, tapping and clicking constituted the same action. In order to determine whether these actions are the same it is important to reintroduce their hardware context. Pressing as fine motor ability can be found on standard controllers, handheld motion controllers, handheld consoles, keyboards, fight sticks. In all of these cases, pressing is the action used to interact with buttons on the controller. Pressing these buttons is completed by bending the thumb and/or fingers to depress a physical button. Clicking is an action exclusive to mice, and is performed by depressing a physical button by bending fingers. The orientation of a players hand and wrist in the clicking scenario is identical to those found when using a keyboard. Therefore, it seems like clicking mouse buttons is a specific instance of the larger pressing action. Tapping is an action found in the context of smartphones/tablets, and handheld consoles, wherein the player uses their finger to touch a designated spot on a touchscreen. The motion made when touching the touchscreen is identical to that of pressing a button (i.e. finger bending). The only difference between the two actions is the associated hardware; where pressing involves physical buttons (i.e. hard controls), tapping involves on-screen buttons (i.e. soft controls). This difference in types of controls affects the feedback from the motion, but does not affect the motion itself. As we are solely concerned with isolating motor abilities, if the only difference is feedback then we can consider the difference non-existent. It is important to note that there may be a measurable difference in the force used by a player when tapping a touchscreen versus pressing a physical button. At this point in our research, we consider this difference to be negligible — this is because we are concerned solely with motor abilities in relation to the intended interaction. It seems apparent that in the context of a game challenge, a player would be likely to use more force when interacting with a

Hands	Fingers	Pressing Bumping Pulling Swiping Pinch-to-zoom Flicking Tapping Clicking Scrolling
	Wrist	Shaking Flicking Pointing Swinging Drawing Moving Tilting
Head	Neck	Moving
	Face	Speaking Making facial expressions
Feet	Ankle	Moving
	Foot	Pressing

Table 3.5: Fine motor actions divided by body parts

physical button than when touching a flat touchscreen. This decision of force application does not change the movement in a significant way for our coarse listing of motor abilities. As such, we conclude that tapping is another specific instance of pressing. Pulling exists solely in the context of trigger buttons, as found on standard controllers, and handheld motion controllers. Compared to pressing, pulling seems different due to hand orientation and finger position. From the way that these controllers are held, one of the players fingers should be resting on the trigger button(s) at all times. In pulling the trigger button, the player bends their finger to depress the button. This motion, though oriented differently, seems identical to the pressing action. The question then becomes whether this orientation change creates a difference that is noticeable to a player. It is important to note that at some level these actions can be seen as measurably different. The real question is whether these differences are actually important. Though we are trying to isolate motor abilities, these actions do not exist in a vacuum apart from the cognitive abilities previously discussed. We believe that the only way to truly know whether actions can be considered identical is by comparing how the brain responds to each of these actions. In measuring how the brain interprets these actions, we could undoubtedly conclude whether they are different enough that it would impact game results. However, to be able to do that we would need to run several neurological experiments, a feat that would be solidly out of our field of expertise, and out of our scope. Therefore, at a coarse level, the fact that pressing and pulling are executed in the same way (finger bending to depress a button) leads us to believe they are the same action. Bumping exists solely in the context of shoulder buttons, as they exist on standard controllers, and handheld consoles. Like trigger buttons, the players fingers naturally rest on these buttons when holding the controller and the buttons are depressed through bending fingers. The orientation of the hand when using shoulder buttons is identical to that of the trigger buttons; as such the analysis for pulling actions applies as well to bumping actions. Therefore bumping and pulling are the same motion, which in turn we have shown to be the same as clicking and pressing motions. For the purposes of our research, we choose to encapsulate these terms under the Pressing name.

Our attention then turned to the actions of swiping, flicking, and scrolling, which we believe to be the same action presented in different contexts. Swiping as an action exists in the context of standard controllers, and smartphones/tablets. It is a motion wherein the player moves their finger across an area of a touch-sensitive surface (usually a screen). Often the finger is held straight, while the motion is controlled by the movement from the first knuckle (metacarpophalgeal joint – where the finger meets the hand) or wrist. We count stylus pens as a substitute finger, and as such use the word finger to represent both stylus pens and actual fingers. Flicking as a finger action exists in the context of smartphones/tablets. Flicking is virtually identical to swiping (player moves their finger across a touch-sensitive surface), with the sole difference being speed. Flicking is a motion that is done rapidly, whereas swiping can be done at any pace. For our purposes the exact speed of an action is not important, rather its the relative time it takes to complete the action which is important. We refer to Newells Time Scale of Human Action (see Figure 3.1) [123] to determine what is considered reasonably similar times. Newell groups various actions into different bands (social, rational, cognitive, biological) in order to discuss the speeds at which each action should be completed [123]. Planned motor actions, like the ones we are discussing, fall under the cognitive band. This categorization shows that any actions operating on the scale of hundreds of milliseconds to tens of seconds, could reasonably be considered equivalent from a processing standpoint. Therefore, when not discriminating for speed, we can conclude that flicking and swiping are the same. Scrolling as we are discussing it, is an action unique to mice. We purposefully omit discussion about scrolling on touchscreen devices as their implementation of scrolling is an adapted swiping manoeuvre. Scrolling on mice is different in that there is a physical scroll wheel with which the player interacts. Scrolling in the mouse context is completed by bending their finger so as to move the scroll wheel forwards or backwards. In examining the hand orientation and the motion made when using a scroll wheel, it would initially seem that scrolling may fit in more with pressing, as both use finger bending motions to complete the action. However, upon closer examination, we can see that pressing as an action is exclusively an adductive movement (your finger is always moving inwards/towards your body), whereas scrolling can be performed as either an adductive or abductive movement (you can move the scroll wheel backward - towards your body - or forward - away from your body). In this regard, scrolling fits in better with swiping which is also possible as both an adductive or abductive movement. The major difference between scrolling and swiping is where the movement happens. For swiping movement tends to occur at the first knuckle, whereas for scrolling movement is more common at the second and third knuckles. Which knuckle the finger bends at does not affect the time taken to complete the action (still in the cognitive band), and there is no obviously measurable difference in performance when bending at the different knuckles. Therefore at this time, we believe that scrolling actions can be captured with swiping actions. We are aware that further investigation needs to be done into whether there is a significant difference between these actions, however as we are currently only taking a rudimentary look at motor abilities, we choose to leave this for future researchers. At this time we decide to label the swiping, flicking, and scrolling actions as “Swiping”.

Having now eliminated redundancy in the finger actions, we can turn our attention to the remaining actions. For fingers, the only remaining action is

Scale (sec)	Time Units	System	World (theory)
10^7	Months		SOCIAL BAND
10^6	Weeks		
10^5	Days		
10^4	Hours	Task	RATIONAL BAND
10^3	10 min	Task	
10^2	Minutes	Task	
10^1	10 sec	Unit task	COGNITIVE BAND
10^0	1 sec	Operations	
10^{-1}	100 ms	Deliberate act	
10^{-2}	10 ms	Neural circuit	BIOLOGICAL BAND
10^{-3}	1 ms	Neuron	
10^{-4}	100 µs	Organelle	

Figure 3.1: Newell's Time Scale of Human Motion

"pinch-to-zoom". Pinch-to-zoom is an action exclusively available on touch-sensitive devices, in which the player coordinates the movement of their thumb and index finger to create a pincer grip/pinching motion. This action is significantly different from those previously discussed because of its demand for finger coordination. Pinch-to-zoom cannot exist without the player coordinating movement of two fingers. Though this separates it from movements like swiping at the most basic level, motor coordination at this level does not significantly impact the average player. This is because cognitive load measurements focus exclusively on the allocation of working memory resources, and motor control activities like finger movement are processed by an entirely different section of the cognitive system. However, performing coordinated activities has been shown to increase cognitive load for older adults [124, 190, 161, 121, 77].

It is important to distinguish single task coordinated actions like pinch-to-zoom, from multi-task coordinated actions like pressing multiple buttons. Single task coordinated actions require movement coordination to accomplish a single specified goal. In comparison, multi-task coordinated actions are when an individual performs two non-coordinated single task actions at the same time. For example, in Kingdom Hearts 2 and its re-releases the player controls the avatar's movements with the left thumbstick and the camera movement with the right thumbstick. The player can complete both of these actions independently (non-coordinated single task actions), but to effectively play the game the player must do both simultaneously (multi-task coordinated action) to succesfully navigate the game world. Many game challenges require players to use multi-task coordinated actions. We believe from our understanding of cognitive and motor control, that this affects the cognitive load (and thereby perceived difficulty) of the challenges but may not affect the motor difficulty. This is because multi-task coordinated actions are asking players to simultaneously achieve two different sub-goals (e.g. moving the player and moving the camera in Kingdom Hearts). However, the level of motor difficulty is fixed with each interaction (i.e. pressing a button is always the same level of intrinsic difficulty). We state this because we are not focusing on the effects of fatigue over long play sessions wherein the level of motor difficulty may increase over time.

3.1.2 Wrist and Forearm

Wrist actions are arguably the second most important set of movements in regards to video game interaction. Our first impression of wrist movements is that most of the listed actions are the same with different speed requirements. The reason we believe this is from the range of motion available to the wrist (because it is an ellipsoidal joint). As with finger actions, we need to examine each wrist action in their hardware contexts to determine whether this is true.

Pointing as an action exists in the context of handheld motion controllers, and involves the player positioning the on-screen cursor. Handheld motion controllers in this context are intended to be held in one hand (similar to T.V. remotes). When held like this the wrist movements are limited to lateral (wrist flexion and extension - see Figure 3.2) and vertical movements (radial and ulnar deviation - see Figure 3.3). Lateral movements are the type of movement made when waving as a greeting, while the vertical movements are those made when fanning oneself with their hands. It is possible that when pointing, users will also incorporate forearm movements into the motion so as to increase their range of motion. Regardless, the movements made are extremely controlled as often the goal of pointing tasks is accuracy; however, this does not mean it is inherently done slowly. The speed at which a user enacts a pointing action is determined by the context of the action and the cost of error. For example, navigating a menu in a game has no time limit or cost associated with missing the target. As such, the user can move at their own speed. However, if the pointing task was being done in a game context with both a time limit and an error cost (e.g. shooting ducks in Duck Hunt) then movement speed is important.

Flicking as a wrist action exists in the context of handheld motion controllers when the controller is held in one hand. In this context flicking is the rapid movement of the controller from one point to another. The motion used to complete this action is also lateral movements of the wrist. As discussed

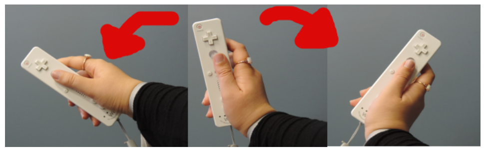

Figure 3.2: Wrist Flexion and Extension while holding a handheld motion controller

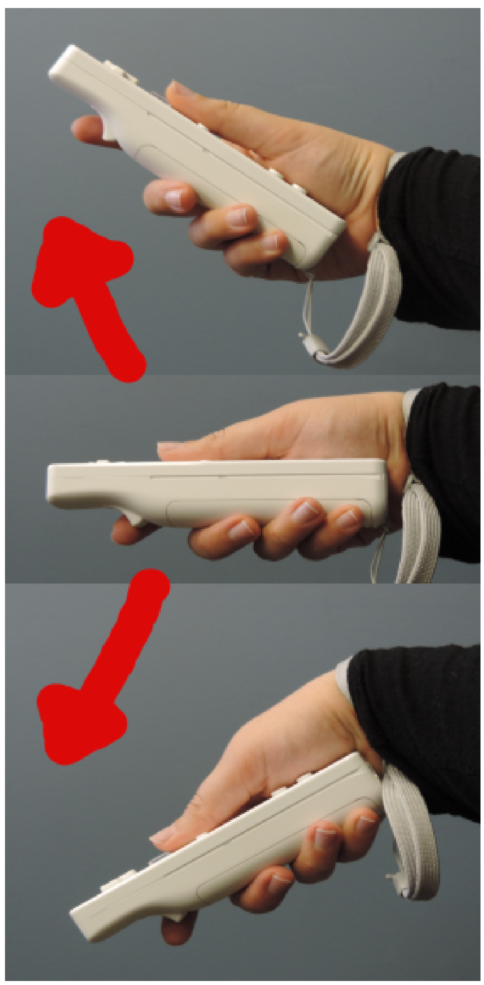

Figure 3.3: Wrist Deviation while holding a handheld motion controller

in the previous section (Section 3.1.1), flicking is an inherently quick movement. Though both of these actions take place in Newells cognitive band, and as such can be considered to be of the same relative speed, they should still be considered separate actions. This is because flicking is a discrete action, while pointing is continuous. It is important to note at this time that while we are discussing an isolated instance of pointing as an action to determine wrist movements, it is actually a continuous action. Handheld motion controllers rely on the user pointing the controller appropriately to interact with a game, and as such users are continuously moving the on-screen cursor. Over extended periods of time this repetitive motion may cause fatigue and stress in the user. However, as the effects of fatigue and stress are out of our scope, we will not examine the effects of this in detail. Still, this distinction between continuous and discrete directly affects the average completion speeds of these actions. As such, this difference in average speed and continuous versus discrete actions affects how and where these actions appear in a game context. Pointing as an action tends to exist in the context of accuracy tasks, an example being the archery mini-game in Wii Sports Resort [148]. It can support an entire challenge on its own, and may often appear alongside pressing actions. In comparison, flicking in game contexts usually appears alongside a variety of different motions. An example being the table tennis mini-game in Wii Sports Resort where it is the motion used to serve the ball [148]. Since it is inherently quicker and less accurate than pointing, flicking appears less frequently. Therefore we acknowledge that the wrist movements underlying these two actions are identical when not accounting for speed and fatigue effects, but we decide to keep them separate. We do this with the understanding that these two actions appear in different in-game contexts, and as such we need the distinction between the two for the purposes of future analysis.

Having explored the differences between pointing and flicking, we revise our initial impressions. Though all of these actions use the same movements, we believe that there is enough difference in game context to keep them largely separate. We realise that in order to support this assumption we need to further examine the other identified wrist actions. Having previously discussed pointing, we move on to discussing drawing and tilting as fine motor abilities. We begin with these two because they are similar to pointing in average speed, and in game context.

Tilting as an action is found in the contexts of handheld motion controllers, smartphones/tablets, and handheld consoles. Though all of these contexts rely on wrist movements, the way in which each device is held affords different degrees of movement. For handheld motion controllers when held in a single hand, tilting laterally involves the player twisting their wrist and forearm to angle their controller in the same motion as turning a doorknob (wrist supination and pronation - see Figure 3.4). Tilting vertically in this context is the

Figure 3.4: Wrist Supination and Pronation while holding a handheld motion controller

Figure 3.5: Forearm flexion and extension while holding a handheld motion controller

same movement as vertical pointing movements (radial and ulnar deviation). Smartphones/tablets can also be held in a single hand when in portrait mode. In this case tilting is done in the same manner as for handheld motion controllers. In comparison, handheld motion controllers held horizontally, smartphones/tablets in landscape mode and handheld consoles are held between the hands. In tilting up and down, the motion remains the same as vertical tilting for smartphones/tablets in portrait mode and hand held motion controllers (radial and ulnar deviation). However, when tilting laterally the wrist's main function is stability and the tilting motion is performed by coordinated movement of the forearms (forearm flexion and extension - see Figure 3.5). For example, when holding the Nintendo Wii U gamepad, tilting the device laterally to the left requires the player's right forearm to move up (flexion) while their left forearm simultaneously moves downward (extension). The player's wrists remain stable in order to hold the controller so it is not dropped. An example of tilting in game is steering the flying beetle item in The Legend of Zelda: Skyward Sword which is done by tilting the handheld motion controller. Like pointing, tilting is a continuous action whose completion speed is subject to the in-game context and error cost. Therefore these actions seem extremely similar in properties. However, because of the additional twisting (supination and pronation) movement, we believe that there is sufficient difference in the actions to keep them separate.

Drawing as an action is found in the context of handheld motion controllers held in a single hand, smartphones/tablets, and handheld consoles. Handheld motion controllers act as a paintbrush proxy with the previously described wrist movements controlling the tracing path of the cursor. Drawing on smartphones/tablets and handheld consoles involves interacting with touch-sensitive screens using a finger or a stylus. The motion used in tracing on these screens is primarily controlled by the previously described wrist movements, with the occasional forearm movement for large screens (those that extend past the range of motion of the wrist). In all of these environments the motion of drawing is controlled primarily by the wrist. An important characteristic of drawing as an action is that it is controlled; like pointing, accuracy is an important factor and so speed is variable. For all of its similarities to both tilting and pointing, drawing is sufficiently different in using the fullest range of motion. Unlike pointing and tilting, drawing also takes into account the ability of the wrist to rotate in actions like tracing circles and curves. This additional form of movement is a combination of the wrist movements we have already discussed, as the wrist is unable to properly rotate on its own because of its joint type (ellipsoidal instead of ball and socket). As such, this extra required movement makes it sufficiently distinct from the other two actions.

We now move on to examining shaking, and swinging as motions. These motions are similar in movement, speed, and in game context to flicking. Our initial understanding is that unlike flicking, shaking and swinging seem to imply a repetitive action. The examination of these actions will clarify whether repetition is inherent in these actions, and if so, whether it is sufficient to differentiate them.

Swinging as an action exists in the context of handheld motion controllers held in a single hand. Swinging is virtually identical to flicking with the exception that it is repetitive. In swinging the player moves their wrist laterally back and forth (flexion and extension). Swinging as a wrist motion exists in multiple game contexts. Examples include using tools like the fishing rod and net in Animal Crossing: City Folk [147], cracking an egg in Cooking Mama: Cook Off [54], and sword actions in The Legend of Zelda: Skyward Sword [144]. In all of these contexts repetition is not always necessary to complete the goal of these tasks. For example, casting the fishing rod in Animal Crossing: City Folk only takes one swing of the controller, while fighting an enemy in Skyward Sword takes several swings of the sword. Therefore repetition does not seem to separate swinging from flicking, this would seem to imply they are the same action. However, unlike flicking, swinging as a motion is not inherently quick; this built-in speed limits how these actions appear in game contexts. Flicking is used in less accurate situations, and often in single instance uses (e.g. casting the fishing rod). Swinging can be used in more accurate situations and multiple instance uses (e.g. swinging the sword along specific paths). These differences leads us to believe that these actions should remain separate.

Shaking as an action exists in the context of handheld motion controllers (both orientations), smartphones/tablets, and handheld consoles. It involves movement of the wrist and forearms to quickly and repeatedly move the controller in a back and forth motion in some direction. For handheld motion controllers held in one hand and smartphones/tablets in portrait mode, shaking

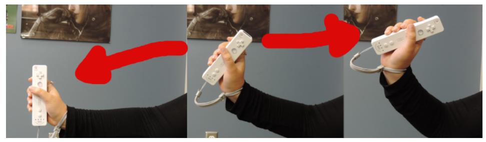

Figure 3.6: Forearm flexion and extension while holding a handheld motion controller

exists as either a vertical wrist motion (radial and ulnar deviation) mimicking the motion of a drumstick tapping on a drum, or as a jerking forearm movement (forearm rotation - see Figure 3.6) similar to the motion of shaking a drink to mix the context. Examples of this type of shaking in games include: ground pound in Donkey Kong Country Returns when using a handheld motion controller [173], and asteroid in SpaceTeam on smartphones/tablets [194]. For handheld motion controllers held horizontally, smartphones/tablets in landscape mode and handheld consoles which are held between the hands, shaking is exclusively the result of forearm movement (forearm flexion and extension). Though shaking actions are possible for these hardwares in this orientation, they are most common for handheld motion controllers. Examples of shaking in this way include: ground pound in Donkey Kong Country: Tropical Freeze [174], performing wheelies in Mario Kart 8 [146], and performing the homing hat throw in Super Mario Odyssey [151]. At this time we are not able to find in game examples of shaking for landscape smartphones/tablets and handheld consoles. This is not to say that they do not exist, but that they are incredibly few and far between. This is possibly because these hardwares have the screen attached to the controls. As such, shaking the controls would involve shaking the screen as well, potentially making the game extremely difficult to play. Like swinging, shaking does not inherently mean repetitive motions, or quick motions. The movements required for all shaking contexts are also sufficiently different from both flicking and swinging. Therefore we believe that shaking actions should remain separate as well.

3.1.3 Neck and Face

It is our understanding that there is no overlap in the motions made by the neck or face. This section will be used instead to clarify the motions of both body parts and to contextualize them in gaming environments.

The necks only motion is moving. This action reflects the necks purpose in creating head movements such as tilting, nodding, or shaking. These actions are becoming more important for AR and VR games, which use headsets and monitor head movements as input. As these types of games are currently out of our scope, we refrain from further examining neck movement actions.

The face handles two actions: making facial expressions, and speaking.

Though uncommon, both of these exist in currently available games and so need to be further explored. Making facial expressions exists in the hardware context of handheld consoles, and is performed by making faces at the front facing camera. An example of this is the Pokemon Amie mini-game in Pokemon X and Y for the Nintendo 3DS [74, 75]. Speaking as an action exists in the hardware context of smartphones/tablets, and handheld consoles, and is performed by making noise directed at the devices microphone. Speaking as we describe it here is not to be confused with natural language processing, rather the microphones are only detecting whether a noise is made and at what intensity. Examples of speaking used this way in games include Puzzle 138 in Professor Layton and the Diabolical Box which requires players to blow into their microphone simulating a gust of wind [119], and Chicken Scream on smartphones which allows the user to control how the chicken avatar moves by making sounds [165].

3.1.4 Ankle and Feet

As with neck and face, we do not believe there is any overlap in the actions made by the ankle and feet. This is because of current controller limitations; existing controllers that use foot input (mat controllers) only allow for pressing as an action. Therefore, even though there are many potential movements for ankles and feet, we are limited to considering the two as a single unit and to condense all possible actions to just ”pressing”. Examples of this in a game context includes Dance Dance Revolution [111], Shaun White Skateboarding [222], and Mario and Sonic at the Winter Olympic Games [187].

Having explored all our listed fine motor abilities, we are left with the following list (Table 3.6).

3.2 Gross Motor Abilities

With the increase of motion controls, gross motor abilities are becoming more important as an interaction method with video games. However, we believe that due to elements like fatigue, stress, and other physical limitations, gross motor abilities as a main control method are still less important than fine motor abilities. Still, the list of gross motor abilities that we have generated (Table 3.7) can be further refined. As with fine motor abilities, we begin by grouping the existing list in regards to the part of the body it uses (see Table 3.8). From this grouping we see that most gross motor abilities for video games focus on arm movements. This is possibly because the existing interaction hardware, such as handheld motion controllers, still focuses on being operated with the players hands. As with fine motor abilities, the listed actions appear across different body parts. The context of these actions and underlying motion

Hands	Fingers	Pressing Swiping Pinching
	Wrist	Shaking Flicking Pointing Swinging Drawing Tilting
Head	Neck	Moving
	Face	Speaking Making facial expressions
Feet	Ankle and Foot	Pressing

Table 3.6: Fine motor actions divided by body parts

Gross Motor Actions	**Hardware Context**
Moving	Handheld motion controllers, full body motion controls
Swinging	Handheld motion controller
Pointing	Handheld motion controller
Shaking	Handheld motion controller, smartphones/tablets, handheld consoles
Drawing	Handheld motion controller, smartphones/tablets, handheld consoles
Thrusting	Handheld motion controller
Flicking	Handheld motion controller, smartphones/tablets
Positioning	Full body motion controls

Table 3.7: Gross Motor Actions (subset of Table 3.3) with their hardware contexts

Arms	Pressing Moving Swinging Pointing Shaking Drawing Thrusting Flicking Positioning
Legs	Moving Positioning
Torso	Positioning

Table 3.8: Gross motor skills by body part

differences will explain how this is possible and why they are different. Many of the movements that are listed exist as fine motor abilities as well. We understand that though the labels may be the same in describing the action, the underlying motions will be different due to the size of the movements. At this point we attempt to eliminate redundancy by examining each body part and their actions.

3.2.1 Arms

We start by examining the actions possible using the arms. Though arms are comprised of different sections (upper arm and forearm), in the context of gross motor abilities we consider them to be a single unit. The rationale for this decision is two fold: firstly, because motions of the upper arm automatically affect the forearm, and secondly, because of limitations in current controller technology.

In an attempt to reduce overlap and remove redundancy in this list, we begin by examining ”moving” as an arm action. Moving is an extremely broad term, and encompasses all the other arm actions in Table 3.8. Since arm movements are the most common form of gross motor ability when interacting with video games, it is important to be precise in our explanations and understanding of them. In order to begin to examine “moving” as an action we need to reintroduce its hardware contexts. “Moving” as an arm action was derived from three distinct contexts: full body motion controllers, mouse, and fight sticks. In the context of fight sticks, moving applied to moving the joystick. From understanding this context, we see that the action is controlled by a combination of shoulder and elbow movements. Lateral joystick movements are controlled by the medial and lateral rotation of the shoulder. We

Arms	Pressing Pushing Rotating Swinging Pointing Shaking Drawing Thrusting Flicking Positioning

Table 3.9: Gross motor skills for Arms

choose to call this type of movement ”rotating” because the shoulder rotation is what causes the arm movement. As well, it describes the rotation the arm is making around the torso. Forward and backward movements of the joystick are controlled by shoulder and elbow flexion and extension, wherein the shoulder and elbow are making opposite actions (i.e. shoulder flexion/elbow extension, or shoulder extension/elbow flexion). These movements are more easily described as “pushing” and “pulling” as the arm is being used to move something away or towards the body. To simplify the table we will call this motion “pushing”. In the context of mouse movements, moving encompasses the left/right mouse movements (medial and lateral shoulder rotation) and forward/backwards mouse movements (shoulder/elbow flexion and extension). These movements are identical to those of the joystick, and so we keep the same motions. In the context of full body motion controls, moving applied to arm movements. As previously discussed, these devices are limited to measuring coarse movements through joint-based skeleton pose estimation (section 2.3). Therefore all possible arm movements are measurable. There are many possible descriptors for arm movements, but it would be out of scope for this project to attempt to cover them all. Therefore we limit the movements that we will consider for this thesis to those found in other gaming contexts (positioning, pointing, pressing, shaking, thrusting, swinging, drawing, and flicking). We encourage future work to be done on this section to identifying more possible arm movements for gaming contexts as they begin to appear. After considering the hardware context of “moving” we believe that it should be deleted from this list. We opt instead to add the motions of “rotating” and “pushing” to the existing list of movements, thus more descriptively covering the range of arm movements used in a gaming context (see Table 3.9). We now move on to examining whether any listed actions ought to be combined or eliminated.

We begin by examining the actions of “pressing” and “pushing” to see whether they can be considered the same action. We believe this may be the case due to their use in game contexts; both are used to describe interactions with movable physical controls (e.g. push a button, or press a button). In regards to arm actions, we find pushing and pressing in the context of specialty controllers like fight sticks, and simulation controllers. As we had previously described pushing, we focus now on pressing to see whether they are similar enough to amalgamate. We focus on pressing in the context of simulation controllers, particularly the Donkey Kong Bongos, and the Rockband Guitar. The Rockband Guitar facilitates interactions that use both fine and gross motor abilities depending on the part of the controller being used. The Rockband Guitar controller employs pressing as a gross motor ability when interacting with the whammy bar. The player interacts with this bar by pressing it with the palm of their hand to the body of the guitar (shoulder/elbow flexion and extension). This movement is identical to the pushing action described when using a joystick. In the case of the Donkey Kong Bongos, which can be interacted with using both fine and gross motor abilities depending on player preference, hitting the bongos with an open palm requires arm movements that also fall under the category of shoulder flexion and extension. As such, we find that anywhere a gross motor action can be described as "pressing" it is actually using the same movements as "pushing". Therefore we believe these two actions can be combined to eliminate redundancy in the table.

We turn our attention to thrusting, an action we believe is an instance of pushing. We believe this because of the similar motion used for the two actions despite different contexts. Thrusting is found in the context of handheld motion controllers, and is used to perform in game actions like thrusting a sword into an enemy in Legend of Zelda: Skyward Sword [144], or punching in Wii Sports Boxing [143]. This motion is a quick forward movement of the arm, which we have previously explained to be shoulder flexion and extension. Due to the controller, thrusting is measured exclusively by the arm movement through the accelerometer in the control, and not whether the movement was aimed at a particular target. The thrusting movement is a therefore a quicker, less accurate version of pushing. As previously discussed in the finger section (Section 3.1.1), the exact speed of an action is not important enough at this point to differentiate them. Since we are looking at the movements being made, and thrusting and pushing use the same movements, we choose to combine them under the "pushing" label.

We now focus on rotating, swinging, shaking, and flicking as actions we believe can be combined. Rotating, as previously discussed, is the action caused by medial and lateral shoulder rotations that moves the arm across the torso. We provided the example of moving a joystick or mouse laterally, but other examples include moving handheld motion controllers from left to right. Swinging as an action is found in the context of handheld motion controllers. It involves large sweeping movements of the arm, and can be performed in any direction. Swinging is not an inherently quick or precise action, as it can be performed both quickly/non-precisely and slowly/precisely. Examples of swinging in game contexts include swinging the tennis racket or ping pong paddle in Wii Sports [143], and swinging a sword in Legend of Zelda: Skyward Sword [144]. Swinging when moving laterally or diagonally is the same motion as rotating (medial and lateral shoulder rotation). However, when performed as a vertical chopping motion, it is exclusively controlled by shoulder flexion and extension (the elbow is not used to apply force to anything, and so is not relevant to this motion). Though there are two different motions involved in swinging, we will not separate it into two actions. However, we also believe that we cannot combine it into rotating because this understanding of multiple motions would be obfuscated if we did. Therefore we leave swinging as its own action and move on to the next action. Shaking as a gross motor action is found in the context of handheld motion controllers. As previously explained in the wrist and forearm section (see Section 3.1.2), controller orientation determines how certain actions are performed. From the previous description of how controllers are held when oriented differently (vertically being single handed, and horizontally being between two hands), we see that only one of these orientations lends itself to gross motor abilities. Shaking a horizontally oriented handheld motion controller is predominantly a wrist and forearm motion, which we have categorized as a fine motor ability. Though vertically oriented handheld motion controller shaking exists as a fine motor ability, it can also be considered a gross motor ability depending on how it is performed by the player. When using a vertically oriented handheld motion controller in a single hand, gross motor shaking is performed by moving the arm up and down as if vigorously shaking a person's hand. Like swinging vertically, shaking in this manner is exclusively a shoulder motion (shoulder flexion/extension). An example of this kind of shaking is the Wii Sports Resort Cycling minigame [148], which requires the player to alternately shake the Wii remote and nunchuck to simulate pedaling.

Gross motor shaking actions like this are not common in gaming contexts, likely due to the fatigue that would be caused from this kind of repetitive motion. Shaking as a gross motor action is mechanically identical to vertical swinging. As such, we choose to combine these actions under the term "swinging". Flicking as a gross motor action is found in the context of handheld motion controllers and is a quick motion that moves the arm across the body. As such it is a faster instance of swinging. Examples of flicking in game contexts could not be found, as every instance that could potentially be identified as a flicking action could also be categorized as a swinging action. It appears from our survey of games that flicking exists predominantly as a fine motor ability, and should be removed from the gross motor ability list. Therefore we decide to eliminate flicking as its motion and in game appearances are

Figure 3.7: Just Dance 4 Gameplay showing both arm positioning and pointing

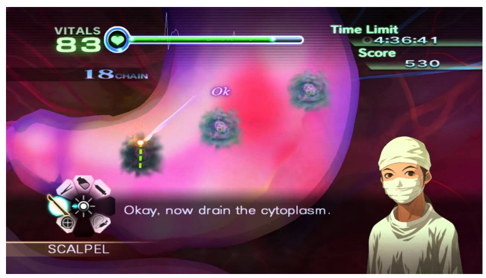

Figure 3.8: Trauma Center: New Blood showing lines for the player to trace

all identical to swinging.

Positioning and pointing as arm movements seem redundant. Both involve controlled arm movements. Positioning is an action where the player flexes, extends, rotates and/or moves their arm into a specific shape. Pointing is an action where the player extends their arm into a straight line in some direction. From the natural language description of these two actions, pointing seems like an instance of positioning. We can see this to be the case in games like Just Dance!, where players are tasked with mimicking the movements of on screen dancers. Figure 3.7 shows an instance of gameplay with four players, where two are tasked with arm positioning and two with pointing. From the in game instances of these actions, we conclude that they are the same and will label them as "positioning" in our table.

Drawing is the final arm motion to be examined. Drawing is a complicated movement that many different motions to trace a large image. The gross motor gaming context of drawing is through handheld motion controllers, which are used in several games to trace lines and images on screen. Examples of games using drawing as a mechanic include the Trauma Center series for Wii (Second Opinion [18], New Blood [19], and Trauma Team [20]). All of these games require the player to trace lines and shapes, and draw stars in a surgery simulation environment (see Figure 3.8). Drawing actions in games combine the motions of previously discussed actions in a more controlled manner. As such, though the underlying mechanics are similar, we choose to keep it as a distinct action.

3.2.2 Legs

As with arms, we consider legs to be a single unit as opposed to individual components (thigh and calf). Unlike arms, legs are less articulated in movement in a gaming context. Legs have two forms of motion: moving, and positioning.

Positioning as a leg action is the same as an arm action. The difference being the smaller potential positions a leg can be in compared to an arm because they balance the body when standing and moving around. Therefore any game that the smaller potential positions a leg can be in compared to an arm because they balance the body when standing and move it around. Therefore any game that uses leg positioning as an action is limited by the possible movements of the leg, the capabilities of existing technology, and the fact that legs play a major role in player stability when standing and so one must always be on the ground. Examples of leg positioning in games exist in Wii Fit's yoga games [149], and Kinect Adventures! 20, 000 leaks mini-game [78].

Moving as a leg action encompasses larger actions such as rotating, swinging, and kicking. It is obvious that moving is too broad of a term; however there are too many possible leg actions to account for them all. Therefore, we focus on ones which we can provide examples of their use in a video game setting (summarized in Table 3.10).

Leg action	Controller context	Game Example
Kicking	Full body motion controller	Soccer in Kinect Sports
Jumping	Full body motion controller Mat controller	River Rush in Kinect Adventures Pressing two buttons simultaneously in Dance Dance Revolution
Stepping	Full Body Motion Controller Mat controller	River Rush in Kinect Adventures Pressing button in Dance Dance Revolution
Jogging	Full Body Motion Controller	Sprint (Track and Field) in Kinect Sports

Table 3.10: Leg actions with in game examples

Though there are examples of leg movements used in video games, they are not widespread. We believe there are two reasons for this. Firstly, the limitations and unpopularity of the input technology affects how frequently these movements are used to complete challenges. Mat controllers and full body motion controllers are not commonly used as input for traditional video games. As previously discussed, full body motion controllers are not popular in a gaming context (see Section 2.3). One reason is because they require very specific conditions to be able to track movement effectively. Conditions like lighting, size and shape of the room where the sensor is in, position of the sensor, and even the clothes the player is wearing could influence the ability of the sensor to track the player. Mat controllers, while more popular, are only used in dance games and simulations (see Section 2.7.2). These technologies do not translate well as input types to challenges that are traditionally found in games.

Secondly, the physical limitations of the player and restriction of leg movements affects how these movements are used to complete challenges. Large leg movements (e.g. kicking, jumping, running) are exhausting when done repetitively. This limits when and for how long a player could be asked to make these movements. According to the 2017 Nielsen 360 Gaming Report, the average gamer (aged 13 and older) spends about 40% of their total gaming time on consoles (of which full body motion controllers can be used as input) [76]. The 2017 ESA report found players average 3+ hours per week gaming [17], meaning about 1.2 hours potentially playing a game that requires leg movement. Consider applying full body motion controls to a game such as Assassin's Creed. In Assassin's Creed the player is constantly in motion; running away from guards, climbing buildings, and tailing their targets. Running in this example would require the player to actually jog in place for as long as they need their character moving. The physical exertion then required to play this game quickly becomes taxing on the player. In turn this limits the amount of time they could reasonably spend playing this game in a single session. This conforms to how games using these control schemes are currently made - short mini-game sessions with an indefinite rest period between them. The player can then control how many of these mini-game sessions they choose to undertake based on their own physical fitness level.

At this point it is important to explain the difference between "traditional" games and "exer-games". Traditional games (e.g. Super Mario Bros., Assassin's Creed, Tetris, Call of Duty), aim to engage the player in "play". As such, the player's expectation of the experience is to be engaged with the content, and to enjoy the experience. They therefore tailor the length of their play session and gameplay expectations to match this goal. Exer-games (e.g. Wii Fit, Wii Fit Plus Yoga, workout mode in DDRMAX!) aim to be exercise. As such their gameplay is meant to be strenuous and physically engaging.

Arms	Pushing Swinging Drawing Rotating Positioning
Legs	Moving Positioning
Torso	Positioning

Table 3.11: Gross motor skills by body part

Players understand that this is the goal when they choose to engage with an exer-game, and as such tailor their play session and gameplay expectations to match. Therefore, a person choosing to play the workout mode in Dance Dance Revolution will choose to play an hour of songs back-to-back, whereas a person looking for a more traditional gaming experience will not.

Therefore, due to the lack of popularity in contemporary gaming, we choose to not further decompose the "moving" action. We believe that further research should be done into leg actions in video games as the technology available becomes more popular and widespread.

3.2.3 Torso

The torso movements encompasses movements of the central part of the body (chest, back, abdomen). We include in our understanding of torso, the hips. Though this is not common as hips tend to be considered part of the legs, we believe that the limitations of hip movements align more closely with our understanding of torso movements. Torso movements tend to be a by-product of other gross motor movements, such as leg movements, in order to stabilize the body. For example, in kicking a leg back and forth a player's hips and torso naturally move to compensate for being off balance. Deliberate and isolated torso movements, such as bending or twisting are often used for stretching or posing the body, such as in yoga or dance. Movements of this part of the body are limited in the context of video game interactions. We have labeled movements of this type as "positioning", which captures the movements of twisting and bending. These types of movements are usually found in dance games, like Dance Central, and simulation games like Wii Fit Plus Yoga.

Having now explored all of our gross motor abilities, we are left with the following list (Table 3.11).

Part III

Analysis of Challenges and Motor Abilities

Having outlined the challenges and motor abilities, we can now attempt to analyze gameplay through the context of player motor abilities. We do this by creating a table where the row headers are the gameplay challenges and the column headers are motor abilities. We then fill the cells with a number between 0 and 100 which indicates how important an ability is to successfully completing the associated challenge. We have broken this range down into 4 intervals, each named to represent the degree to which the abilities are used (see Table 1). We chose the range of 0 to 100 to allow for more granularity in understanding how an ability is used.

Level	**Range**	**Interpretation**
1	1 - 25	Used but not important
2	26 - 50	Noticeably used
3	51 - 80	Important, but not limiting
4	80 - 100	Limiting factor

Table 1: Range of cell numbers and what they represent

The cell numbers we have provided are derived from our experience playing various games and challenges, and so represent our best educated guess as to how important individual abilities are. We provide in the following sections simple analyses of challenge types through representative examples. We use these analyses to explain how we arrived at our numbers. We believe that these numbers should be viewed not as exact information, but as existing in a range of error of ± 10 to account for our uncertainty. We did not have time to verify the numbers in our table, but we believe that they provide a good approximate from which future researchers can build. Verifying all the numbers in our table would require an experiment for potentially every cell presented; ergo potentially 21 experiments per challenge, totaling 378 experiments for the information presented here. Future work should attempt to find more concrete numbers to fill the table with by conducting a series of experiments isolating the motor abilities and the challenges.

The purpose of this analysis is to allow us to find existing trends in gameplay challenges. So long as the relative ordering is somewhat correct, we will be able to derive some general conclusions from the data in our table.

0.1 Analysis Methodology

In order to derive numbers for our table we need to analyze instances of each challenge type. To conduct an analysis we first identify a gameplay instance that fits the challenge definition. At that time we note down the controller type and commercial genre of the game. We include genre in this analysis to show we have attempted to examine challenges across a variety of games where possible, and not from single families of games. We note down the game instructions provided to the player and then proceed to play the challenge multiple times. During these playthroughs our goal as player is to "win". After each play instance we record the motor abilities we used (as per our nomenclature), the general cognitive abilities, and any strategies we employed to be competitive. Following this we establish an ordering based on the relative importance of the abilities used. By establishing a relative importance we can assign these abilities numbers in a more educated way, knowing that though the final numbers may not be precise, the general ordering should be more correct. This would allow our trend analysis and conclusions to be more credible as data changes should only affect the finer details of the trends, not their general appearance.

For every challenge we have attempted to compile at least 4 examples from a variety of games. This has proved difficult for some challenges, and so some analyses are sparser than others. This does not mean more instances of any individual challenges do not exist, merely that we could not find any suitable examples. An issue to note with the current form of analysis is outlier examples. Though challenge types are somewhat tied to implementation (controller type) there are still cases where singular challenges can have different implementations (control schemes). For example *Rapid Analog Stick Rotation* can be implemented for thumbsticks on standard controllers (e.g. Lulu's Overdrive [205]) or for larger joysticks (e.g. cycling in Summer Games 2 for the Commodore 64). This difference in implementation may seem minimal, but it drastically affects the motor abilities used (the former focuses on finger/wrist movements, while the latter focuses on wrist/shoulder movements). A more drastic example include *Rapid Shape Drawing* which exists in the controller context of smartphones/tablets (e.g. Do or Dry [106]), handheld console with stylus (e.g. Wash Rice [53]), and handheld motion controller (e.g. Healing Touch [18]). In our analyses we capture these gameplay instances; however, as this thesis is preliminary work, we choose to focus on the most common implementation and associated abilities of any challenge.

0.1.1 Cognitive Abilities

We acknowledge that every challenge requires at least basic cognitive abilities to be completed successfully. As previously discussed (section 0.2) we have chosen to focus on motor abilities at the moment. We hope to explore cognitive abilities in the context of challenges in the future, and integrate them with our current work. In the meantime, we assume all challenges require the fundamental cognitive abilities of perception (to process what they are seeing and hearing in the game), attention (to focus on the challenge), and memory (to remember what is being done).

Chapter 1

Speed Challenges

This chapter explores the analysis of motor abilities used in speed challenges. The analysis of each challenge is summarized in tabular form at the beginning of each section. The content of these sections will deal with explaining how the cell numbers are derived from the tabular analysis.

1.1 Single Input Button Mashing Challenges

Genre	**Game**	**Instance**	**Controller**	**Motor**	**Cognitive**
Party	Mario Party 5 [96]	Manic Mallets	Standard (Gamecube)	Finger pressing and wrist pointing	Perception (minimal)
RPG	South Park: The Stick of Truth [157]	Dragon's Breath	Mouse (left button)	Finger pressing and wrist pointing	Perception (minimal)
Platformer	Donkey Kong Country: Tropical Freeze [174]	Boss Knockouts	HM (WiiRemote-horizontal)	Finger pressing and wrist pointing	Perception (minimal)
Action-adventure-stealth	Metal Gear Solid 2: Sons of Liberty [112]	Holding breath in Solidus Snake fight	Standard (Dualshock 2)	Finger pressing, wrist tilting, and forearm shaking	Attention (minimal) and perception (minimal)

Table 1.1: Case studies for single input button mashing

HM = Handheld Motion

Through these examples we arrived at 3 motor and 2 cognitive abilities. We order these in terms of importance as:

1. Finger pressing,

2. Wrist/Forearm shaking,
3. Wrist pointing,
4. Perception, and
5. Attention

Depending on the context of the challenge (absolute difficulty, single vs. multiplayer, chance of loss/game over, etc) we employ different strategies to "win". Examples 2 and 3 are single player instances where there is no penalty for not maxing out the score; therefore, there is no external pressure on the player to come up with strategies to max out the score. In these cases we found that finger pressing most frequently determines success. In addition, we notice that we naturally locked our wrists (wrist pointing), which allowed for us to press the button faster.

In contrast, examples 1 and 4 held the potential for loss and "game over" respectively if the player did not max out their button mashing potential. So we employ different strategies to ensure success (e.g. holding the controller differently, pressing with different fingers, etc). We found that some strategies still result in the same motor abilities being used. To be competitive in example 1, we hold the controller in one hand and push the button with the index finger on our opposite hand — still a finger pressing, and wrist pointing combination.

Other strategies emphasized using a combination of abilities to succeed. For example 4 we hold the controller in one hand, perpendicular to our torso while bracing the controller against it, and tilt and shake our wrist/forearm to quickly rub our finger over the button in order to "press" it — a wrist/forearm shaking, and wrist pointing combination. We believe that this strategy was only necessary because of the extreme difficulty and associated penalty of example 4.

Regardless of context and strategy, we never actively noticed the use of cognitive abilities. Perception and attention were used in the most minimal sense to know what we were supposed to be doing in the challenge, and to maintain our active engagement with the game.

From these examples we see that the crux of this challenge is taxing the player's motor abilities. We say finger pressing is the limiting factor (level 4), as it appeared to be most important regardless of context. In relation to finger pressing, wrist pointing is noticeably used (level 2) but we do not believe it would ever impede a player from winning. These motions are a natural consequence of how the player holds the controller, with their finger resting on buttons. We place wrist/forearm shaking as noticeably used (level 2) because we find it becomes more important as the difficulty of the challenge increases. Between wrist/forearm shaking and wrist pointing we believe that

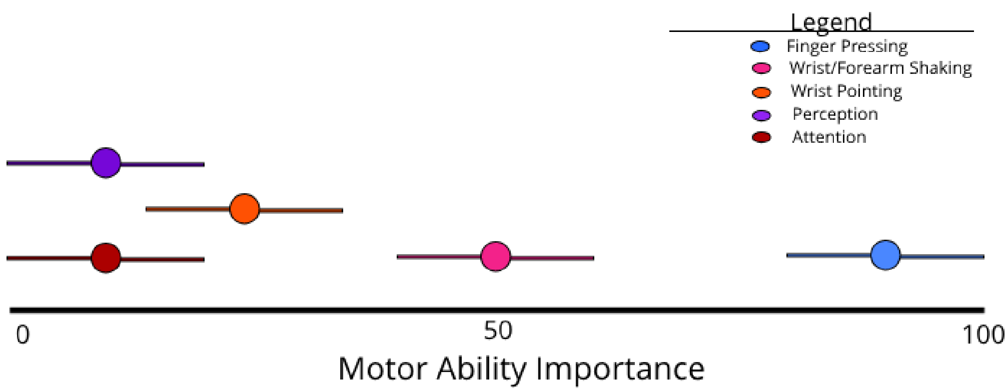

Figure 1.1: Summarized analysis for Single Input Button Mashing with error bars of ±10

the former is more important during play, and so list it higher relative to the latter. Perception and attention were both used, but not important (level 1), we believe that neither is more important than the other at this level.

In summary, the single input button mashing challenge uses the following abilities: finger pressing (90), wrist/forearm shaking (50), wrist pointing (26), perception (10), attention (10).

1.2 Multipe Input Button Mashing Challenges

Genre	Game	Instance	Controller	Motor	Cognitive
Party	Mario Party 2 [93]	Mecha-Marathon	Standard (N64)	Finger pressing, wrist pointing, wrist/forearm tilting	Perception (minimal), attention (minimal)
Party	Wii Party [155]	Chin-up Champ	HM (WiiRemote - vertical)	Finger pressing	Perception (minimal), attention (minimal)

Table 1.2: Examples of single input button mashing in commercial games

HM = Handheld Motion

Through these examples we found 3 motor and 2 cognitive abilities. We order them in terms of importance as:

1. Finger pressing,

2. Wrist/forearm tilting,
3. Attention,
4. Wrist pointing, and
5. Perception

Both examples exist in a multiplayer context where "winning" is achieved by out scoring your opponents. This put external pressure on us as the players to come up with ways to be more efficient at playing. For example 1 this resulted in us holding the controller in a non-standard way. Due to the shape of the N64 controller and position of the buttons that needed to be pressed (A and B), being competitive meant switching to hold the controller in one hand and resting our thumb on the opposite hand across the buttons. This meant that pressing the buttons came from a combination of wrist/forearm tilting and wrist pointing. When playing on easier difficulty settings against NPCs using two fingers to press the buttons (finger pressing) was sufficient to win.

For example 2 being competitive did not require any changes in how we interacted with the controller. Due to the shape of the WiiRemote and position of the buttons, pressing the two at the same time resulted in a natural pinching like motion (finger pressing). The ergonomic nature of the controller and motion translated to this instance of multiple input button mashing feeling easier than example 1. It also seemed to allow us to get higher scores (more button presses).

For both examples we found attention was noticeably used, as we focused more closely on the coordination of our movements specifically when we felt that we were lagging behind in performance. Perception was also used in making sure we knew which buttons to press and whether we had pressed both buttons fully, but overall was not as important.

From these examples the difficulty of this challenge is taxing the player's motor abilities. We say finger pressing is the limiting factor (level 4) since it is used in both examples. In relation to finger pressing, we find wrist/forearm tilting to be important, but not limiting (level 3). This is because it became the dominant motion as the challenge became harder. Though that may have been due to the button placement, most standard controllers have face buttons that are close together and would therefore need the same technique of button pressing. Attention and wrist pointing were both noticeably used (level 2). Perception was used, but not important (level 1).

The numbers and ordering we present is obviously limited by the small number of examples. It is likely that other instances of multiple input button mashing exist in games, and we are just not familiar with them. It is possible that this challenge is not easily found because of how conceptually similar it is

Figure 1.2: Comparison of buttons to be pressed on a WiiRemote vs. N64 Controller (not to scale)

to single input button mashing; designers may opt for the more accessible and less taxing single input variation instead of including a multiple input version.

One thing we noticed from the examples presented here is that they seem to be popular in party games. This is possibly because party games are collections of mini-games; each mini-game can be taxing on the player because it will be over quickly and so there can be breaks in the action between them. It's also possible that party games reach a wider audience, and so designers are more inclined to include more physically oriented challenges to keep multiple types of players engaged.

In summary, the multiple input button mashing challenge uses the following abilities: finger pressing (90), wrist/forearm shaking (50), attention (35), wrist pointing (30), and perception (15).

1.3 Alternating Input Button Mashing Challenges

Through these examples we find 2 motor, and 2 cognitive abilities. We order these in terms of importance as:

1. Finger pressing,
2. Attention,

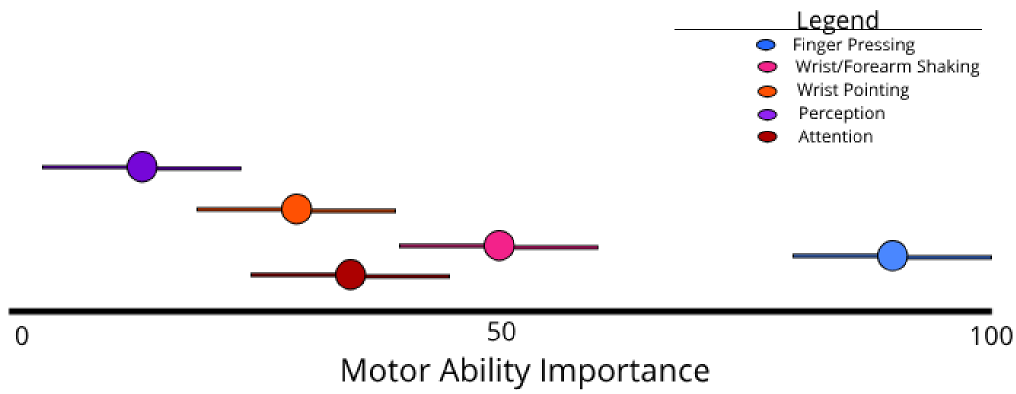

Figure 1.3: Summarized analysis for Multiple Input Button Mashing with error bars of ±10

3. Wrist pointing,
4. Perception

Examples 1 and 3 exist in a multi-player context, therefore putting external pressure on us as players to be more efficient. To be competitive in example 1 we switched to holding the controller in one hand, and resting the index and middle finger of our other hand on the buttons we needed to push (A, B). This meant that pressing the buttons was a combination of finger pressing and wrist pointing (for keeping our wrist locked in position). Example 3 required pressing of the shoulder buttons (L and R) on the Nintendo DS. We were able to hold the device in the intended manner while still being competitive in pressing.

Example 2 exists in a single player context as a small part in a larger boss fight. Due to the shape of the controller, and the buttons that needed to be pressed (L1 and R1), holding the controller in the intended manner already allowed us to be competitive. As well, the difficulty of the challenge seemed significantly lower than the other examples and the punishment of failure was not "game over" but just some health loss in the fight.

For all examples we found that attention was noticeably used, as we focused on the switching pattern of button presses. Perception was used minimally in knowing which buttons to press, and overall was not important. From these examples this challenge's difficulty is taxing the player's motor abilities. Finger pressing is the limiting factor (level 4) since it is the most actively used ability across all examples. We find wrist pointing and attention to be noticeably used (level 2), as we feel both played an easily identifiable role but not enough to deter winning. Wrist pointing was important in relation to finger pressing, as we felt that we were able to press the buttons faster when our wrist was

Genre	Game	Instance	Controller	Motor	Cognitive
Party	Mario Party 2 [93]	Psychic Safari	Standard (N64)	Finger pressing, wrist pointing	Perception (minimal), attention (noticeable)
Action-adventure	God of War 2 [180]	Colossus of Rhodes Fight	Standard (Dualshock 2)	Finger pressing, wrist pointing	Perception (minimal), attention (noticeable)
Sports/Party	Mario and Sonic at the Olympic Games [183]	Cycling	HC (Nintendo DS)	Finger pressing, wrist pointing	Perception (minimal), attention (noticeable)

Table 1.3: Examples of alternating input button mashing in commercial games

HC = Handheld Console

locked versus loose. Attention was important in maintaining a rhythm, as if we got out of sync (i.e. pressing the same button twice) it would throw off the whole score. As such we list them both as equally important (interchangeable in ranking). Perception was used, but not important (level 1).

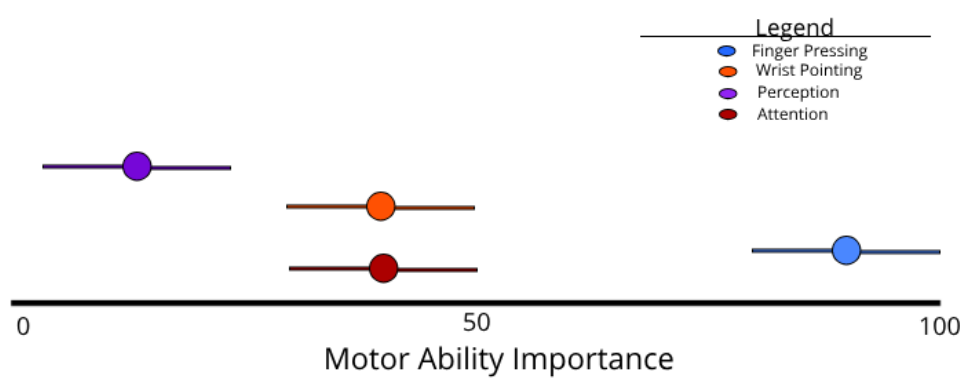

Figure 1.4: Summarized analysis for Alternating Input Button Mashing with error bars of ±10

In summary, the alternating input button challenge uses the following abilities: finger pressing (90), attention (40), wrist pointing (40), and perception (10).

Genre	Game	Instance	Controller	Motor	Cognitive
RPG	Final Fantasy X [205]	Lulu's Overdrive	Standard (Dualshock 2)	Finger swiping, wrist pointing, arms pushing	Perception (minimal)
RPG	South Park: The Stick of Truth [157]	Whirling Doom	Standard (Xbox 360)	Finger swiping, wrist pointing, arms pushing	Perception (minimal)
Party	Mario Party [92]	Pedal Power	Standard (N64)	Wrist pointing, Arms pushing	Perception (minimal)
Party	Summer Games 2 [58]	Cycling	Joystick	Wrist pointing, Arms pushing	Perception (minimal)

Table 1.4: Examples of single input button mashing in commercial games

1.4 Rapid Analog Stick Rotation Challenges

Through these examples we found 3 motor and 1 cognitive abilities. We list them in order of importance as:

1. Arm pushing
2. Finger swiping
3. Wrist pointing
4. Perception

In these examples the most important context to consider is the input. Examples 1, 2, and 3 all use thumbsticks on standard controllers, which by design should be operated with your thumb. Example 4 uses a full joystick, meaning that the player is intended to grip the stick with their whole hand and make adjusting movements with their arm/shoulder. This difference immediately separates the potential motor abilities used; example 4 uses arm pushing to move the joystick around and wrist pointing to keep a grasp on the controller.

Examples 1 and 2 exists in a single player thumbstick context. Both instances are combat moves in a turn-based combat system meaning that enacting the challenge (rotating the analog stick) is an isolated instance that does not affect the time for the rest of combat. In both instances failure to perform means a lower damage score to the opponent, but it does not imply a "game over". In terms of difficulty example 1 is more difficult; in example 1 the degree that we needed to turn the thumbstick to count as a single "rotation" changed with the difficulty level, topping out at 720 ° degrees per rotation, while the timer was constant through the difficulty. With the highest number of counted rotations being 16 [52], that meant we'd have to do 32 rotations in 4 seconds, making it physically impossible to max out the counter. In order to perform even mediocrely in this instance we had to rely on non-standard methods of gameplay. We resorted to holding the controller in one hand and resting the palm of our opposite hand on the thumbstick. When the timer started, we started making quick back and forth motions with our shoulders (arm pushing) while adjusting the movement of our hand by pivoting at the elbow (wrist/forearm drawing) to get the rotation motion. Even with this technique the highest we could achieve was 10 counted rotations.

In comparison example 2 was much easier, as every rotation of the thumbstick counted as a single rotation. We attempted this instance while holding the controller in the intended way (therefore using finger swiping to turn the thumbstick with our thumb), and the non-standard way used for example 1. We found that while we seemed to turn the thumbstick faster with the non-standard method, it didn't translate to significant gains in the damage dealt with the attack.

Example 3 is also a single player standard controller context; however, it is presented as a mini-game in which the player is given 10 seconds to "pedal" their way to success. To be successful in this challenge we employed the same technique we used for example 1. When we attempted this instance using just finger swiping (standard holding technique) we found it impossible to succeed at the challenge.

Regarding the cognitive ability used, we found perception was used in the most minimal sense (to know what to do, and how well we were doing) but did not affect how we mechanically interacted with the game.

For all examples arm pushing was the most sure-fire way to succeed at the challenge, as such we list it as the limiting factor (level 4). In relation to arm pushing, we found wrist pointing was important in maintaining control over the movement of the analog stick. We therefore list wrist pointing as noticeably used (level 2). Finger swiping was used in the easiest form of this challenge (example 2) and conforms to the intended method of interaction based on the design of the controller. We therefore list it as important, but not limiting (level 3); we believe that it is on the lower end of that interval since it can be used but will often be overlooked in favour of the arm pushing strategy. Finally we list perception as used, but not important (level 1).

In summary, the rapid analog stick rotation challenge uses the following abilities: arm pushing (90), finger swiping (65), wrist pointing (30), and perception (10).

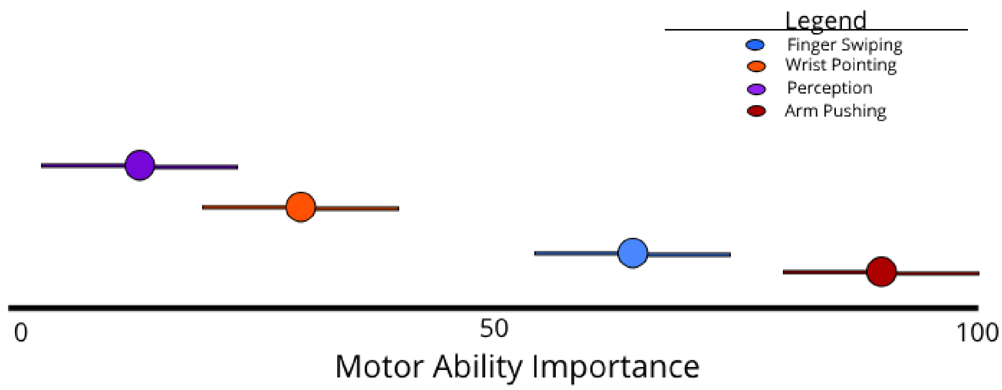

Figure 1.5: Summarized analysis for Rapid Analog Stick Rotation with error bars of ±10

Genre	Game	Instance	Controller	Motor	Cognitive
Idle	Cookie Clicker [107]	—	Mobile (Pixel XL)	Finger pressing	Perception (minimal), attention (minimal)
Action/Puzzle	Dumb Ways to Die [106]	Hot Head	Mobile (Pixel XL)	Finger pressing	Perception (minimal), attention (minimal)
Idle	Almost a Hero [26]	Adventure	Mobile (Pixel XL)	Finger pressing	Perception (minimal), attention (minimal)

Table 1.5: Examples of indiscriminate rapid tapping in commercial games

HC = Handheld Console, Mobile = Smartphone/Tablet

1.5 Indiscriminate Rapid Tapping Challenges

Through these examples we found 2 motor and 2 cognitive ability. We order these in terms of importance as:

1. Finger pressing,
2. Wrist Tilting,
3. Perception,
4. Attention

Example 1 and 3 exist in a single player context with a vertical screen orientation. We played these instances on a Pixel XL (5.5 inches, 168 g [109]), and so our discussion is limited to how they play on this particular device. Since smartphones and tablets come in a variety of sizes, the motor abilities that we have observed might differ for larger or smaller screens. For example, if a player was using an iPad Pro (12.9 inches, 1.49 lbs [11]) how they hold and interact with the device would be different due to the size and weight. In both of these examples performing well meant that our score would be higher, and sub-optimal performances were not penalized by the game. In playing optimally we held the phone in our non-dominant hand and used multiple fingers on our dominant hand to tap around the screen as if drumming our fingers against a desk (finger pressing).

Example 2 is a single player mini-game presented in a series of mini-games, and is played with the phone oriented horizontally. Losing at this mini-game results in losing a life, with "game over" after losing 3 lives. When playing, we held the phone horizontally in our hands and tapped on the screen with our thumbs (finger pressing). We also attempted playing the game with the phone lying on a table and using multiple fingers across both hands to tap the screen (finger pressing). Playing the game in this way did not seem to improve our chances of success.

For all examples we found perception was only minimally used. Attention was also only minimally used in keeping focused on touching the screen.

Therefore we list finger pressing as the limiting factor (level 4), perception and attention as used, but not important (level 1), and we drop wrist tilting from the list.

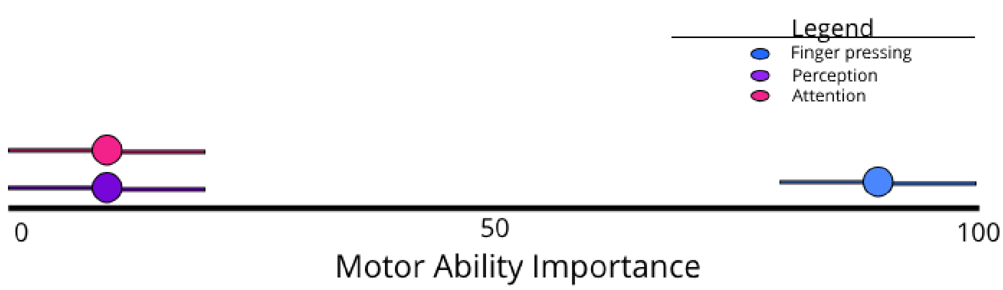

Figure 1.6: Summarized analysis for Indiscriminate Rapid Tapping with error bars of ±10

In summary, the indiscriminate rapid tapping challenge uses the following abilities: finger pressing (90), perception (10), and attention (10).

Genre	Game	Instance	Controller	Motor	Cognitive
Sports/Party Game	Mario and Sonic at the London 2012 Olympic Games [189]	100m Freestyle/"Paddle Fingers"	HC (3DS)	Finger pressing, wrist pointing	Perception (minimal), attention (minimal)
Simulation/Minigame	Cooking Mama: Let's Cook [158]	Make Shape	Mobile (Pixel XL)	Finger pressing, wrist pointing	Perception (minimal), attention (minimal)

Table 1.6: Examples of alternate rapid tapping in commercial games

HC = Handheld Console, Mobile = Smartphone/Tablet

1.6 Alternating Rapid Tapping Challenges

From these examples we found 2 motor and 2 cognitive abilities used. We order them in terms of importance as:

1. Finger pressing
2. Attention
3. Perception
4. Wrist pointing

Example 1 exists in the context of a handheld console (Nintendo 3DS). The game instructions ask us to place the 3DS on a flat horizontal surface (e.g. table) before starting the challenge. This means that, since were not holding the device while playing, we have both hands free to interact with the touchscreen. As well, the way were holding the device cannot influence our strategy in playing the game. The game then instructs us to tap the left and right halves of the screen in an alternating pattern to swim the race, with the instructional images showing both hands being used (left to tap left, and right to tap right). We found that these instructions were an effective way to play the game. This meant that we were using finger pressing to interact with the console, and wrist pointing to keep our hands steady as we tapped. We attempted to play the game using one hand and two fingers and found similar win rates in the challenge. When playing with one hand the motor abilities used did not change.

Example 1 exists in both a single and multi-player context. The mechanics of the challenge are consistent across both contexts, and by extension the motor abilities. However, we found there was a slight difference in the importance of cognitive abilities. In both contexts we found attention to be important as we focused on the alternating pattern and making sure we were tapping in the designated area. In the single player context, we found perception was used, but not important. We needed perception to assist in making sure we were tapping the right area; though we ought to have used perception to keep track of our place in the competition, we found that the NPCs did not pose a significant threat to our lead and so didnt bother. It is possible that on higher difficulty settings a player would need to divide their attention between the game and how they are placed in the event. In the multiplayer context, we found that we ended up checking our place in the competition more and split our attention between the gameplay and watching the other players. We believe this difference in attention and perception was due to the increased difficulty and competition of playing against other people. For this challenge we base our information on the single player experience. Though we noticed a difference in multiplayer, we do not believe it was outside of our ± 10 error bars, and so the relative ordering would be the same. We find that unquestioningly finger pressing was the limiting factor (level 4), and the associated wrist pointing was used, but not important (level 1). In regards to cognitive abilities we find both attention and perception to be noticeably used (level 2), but attention was undoubtedly more important than perception in this instance and so places higher in that interval.

Example 2 exists in a single player smartphone/tablet context. To play this game we hold the phone in a landscape position between our two hands (wrist pointing used to keep phone stable), with our thumbs hovering above the screen. The instructions that we're given to "make the shape" is to alternately tap the right and left hands on screen to toss a ball of ground meat between them. We found that this was easily done with the phone in this position as our thumbs were placed at the right position to naturally touch the different points on the screen (finger pressing). The cognitive abilities used for this instance are identical to those of the single player context of example 1.

Since we were only able to find two examples of challenge, its quite possible this information is too specific to these instances and not generalizable to the whole category. While we were unable to find other examples of this type of challenge, there are many games we are not familiar with which may feature gameplay that fits this category. We believe that our inability to find more instances of this challenge could be due lack of popularity; this challenge mimics the alternating button mashing challenge, and so developers may choose to use that version instead as it provides more haptic feedback to the player through the physical buttons.

In summary, the abilities of alternating rapid tapping are: finger pressing (90), attention (50), perception (30), and wrist pointing (10).

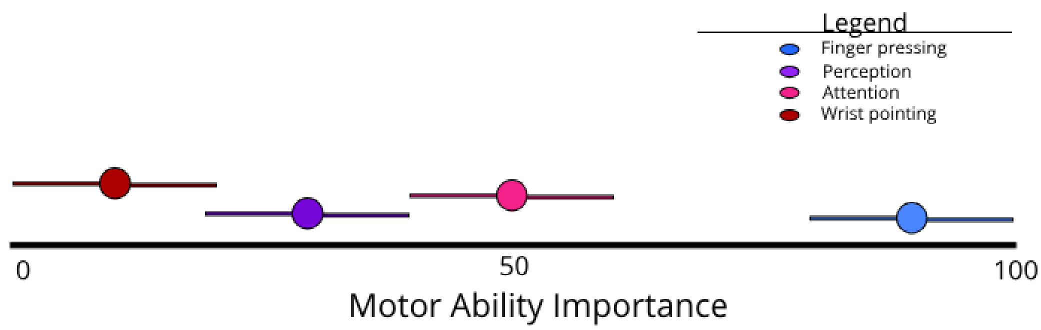

Figure 1.7: Summarized analysis for Alternating Rapid Tapping with error bars of ±10

1.7 Rapid Line Drawing Challenges

We find in these examples 4 motor (wrist flicking, wrist tilting, forearm shaking, arm swinging) and 2 cognitive (perception, attention) abilities.

Example 5 is an outlier, using motion controls and focusing on gross motor abilities like arm swinging. We could not find other examples of RLD challenges that use motion controls or incorporate gross motor abilities. We believe this is likely due to the fatigue that quickly swinging an arm would cause, potentially making it an unappealing challenge for both players and designers. Since this is the only example we could find of RLD challenges in this context, we decide to eliminate it from the analysis.

We therefore have 3 motor and 2 cognitive abilities. We order them in terms of importance as follows:

1. Wrist flicking
2. Wrist tilting
3. Forearm shaking
4. Attention
5. Perception

For all examples we found that both perception and attention were minimally used. Perception was necessary to make sure we were drawing on the touchscreen, and attention kept us focused on the task.

Examples 1, 2, and 3 exist in the context of handheld consoles, and so players can interact with the touchscreen using either the stylus or their finger. We notice that when holding a stylus, we tended towards using wrist tilting movements — mimicking scribbling with a pen. However, when we played the

Genre	Game	Instance	Controller	Motor	Cognitive
Simulation and mini-games	Cooking Mama [53]	Make shape	HC (DS)	Wrist tilting (stylus); Wrist flicking (finger) Forearm shaking	Perception (minimal), attention (minimal)
Action and rhythm game	WarioWare: Touched! [101]	Impressionism	HC (DS)	Wrist tilting (stylus); Wrist flicking (finger) Forearm shaking	Perception (minimal), attention (minimal)
Sports/Party game	Mario and Sonic at the London 2012 Olympic Games [189]	4x100m Relay	HC (3DS)	Wrist tilting (stylus); Wrist flicking (finger) Forearm shaking	Perception (minimal), attention (minimal)
Action and puzzle game	Dumb Ways to Die [106]	Throw Up, Clean Up	Mobile (Pixel XL)	Wrist flicking (finger) Forearm shaking	Perception (minimal), attention (minimal)
Party	Mario Party 8 [97]	At the Chomp Wash	HMC (Wii Remote)	Wrist flicking, Arms swinging	Perception (minimal), attention minimal

Table 1.7: Examples of rapid line drawing in commercial games

HC = Handheld Console, Mobile = Smartphone/Tablet, HMC = Handheld Motion Controller

same instances and used our finger we used a combination of wrist flicking and forearm shaking to move our finger across the screen. We believe this is due to the different position that our hand is in when holding a stylus versus when were just extending a finger.

Examples 1, and 3 exist in both a single player and local multiplayer context. Though the multiplayer context of these challenges seemed more competitive, we found it didnt impact the motor abilities we used. For cognitive abilities, we found there was a marginal increase in attention for the multiplayer context as we focused on moving faster to outperform the other players. However, we didnt find this to be marginal increase to be taxing enough to move it out of the used (level 1) category.

Example 4 exists in the context of smartphones/tablets. Since we played it on a device that didnt come with a stylus we found we used wrist flicking and forearm shaking. It is possible that on devices that come equipped with styluses we would see wrist tilting emerge as the most used motor ability.

In ordering these abilities, we decided to rate the finger-touchscreen interaction as more useful/important than the stylus-touchscreen interaction. This is because the most intuitive touchscreen interaction for the user is using their finger; in the case that a device is missing a stylus, or one is not present, most devices can still be interacted with through the users fingers. We therefore list wrist flicking as the limiting factor (level 4) of this challenge. We list wrist tilting as important, but not limiting (level 3), however we feel that it should be at the high end of this interval to express that it could potentially take the place of limiting factor. Forearm shaking is also important, but not limiting (level 3) as it contributes significantly to efficiently moving our finger across the screen. However, we believe that it belongs closer to the lower-middle half of the interval to reflect that it really is a complementary ability to wrist flicking. Perception and attention are then both used, but not important (level 1). We list attention as slightly higher ranked than perception to reflect the potential for its importance in a multiplayer context.

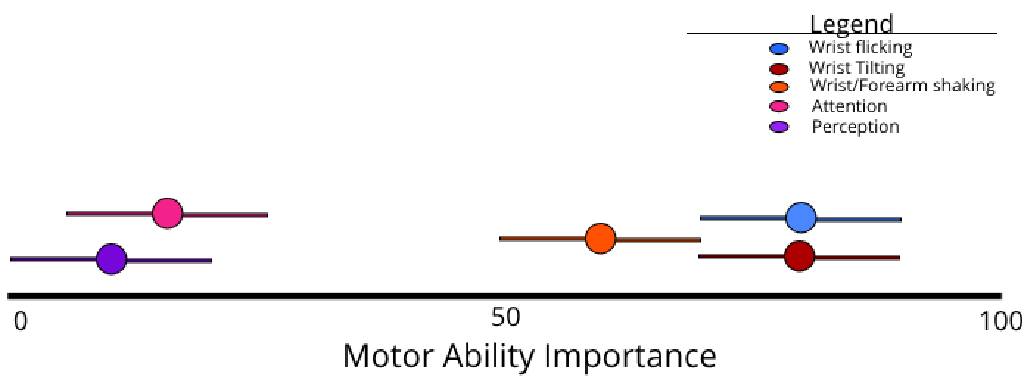

Figure 1.8: Summarized analysis for Rapid Line Drawing with error bars of ± 10

In summary, rapid line drawing uses the following abilities: wrist flicking (81), wrist tilting (80), forearm shaking (60), perception (10), and attention (15).

1.8 Rapid Shape Drawing Challenges

Through these examples we arrived at 3 motor and 2 cognitive abilities. We order them in terms of importance as:

1. Wrist drawing
2. Arms drawing
3. Perception
4. Wrist pointing

Genre	Game	Instance	Controller	Motor	Cognitive
Action and puzzle game	Dumb Ways to Die 2 [128]	Hammer Throw	Mobile (Pixel XL)	Wrist drawing	Perception (used), attention (minimal)
Simulation and mini-games	Cooking Mama [53]	Wash Rice	HC (DS)	Wrist drawing	Perception (used), attention (minimal)
Simulation	Trauma Center: Second Opinion [18]	Healing Touch	HMC (Wii Remote)	Arm drawing, Wrist pointing	Perception (used), attention (minimal)
Action and rhythm game	WarioWare: Touched! [101]	Engine Trouble	HC (DS)	Wrist drawing	Perception (used), attention (minimal)
Simulation and mini-games	Cooking Mama: Let's Cook! [158]	Mix It!	Mobile (Pixel XL)	Wrist drawing	Perception (used), attention (minimal)

Table 1.8: Examples of rapid shape drawing in commercial games

HC = Handheld Console, HMC = Handheld motion controller, Mobile = Smartphone/Tablet

5. Attention

Example 3 is an outlier case, as it exists in a single player handheld motion controller context. In this example, players must hold down the Z button on the Wii Remote and quickly draw a star to enable their Healing Touch [18]. In this context we hold the remote in our dominant hand as if its a TV remote (portrait mode, single hand) and point it towards the TV. Drawing the star means making quick large arm movements something that causes fatigue when done often. This is potentially why we do not often see this type of challenge used with handheld motion controllers. The cognitive abilities used for this instance are identical to the other examples. Though on its surface it seems to fit the description of this challenge type, we choose to omit it from our understanding of these challenges due to its difference in motor abilities, and its lack of popularity. We keep it in this analysis to show that a large motor version of this challenge can exist but does not constitute a separate challenge.

We therefore reorder the motor and cognitive abilities as follows:

1. Wrist drawing

2. Perception

3. Attention

Examples 1, and 5 exist in a single player smartphone/tablet context. We used our finger as the main method of interaction, as the device we played it on does not use a stylus. In both instances we found that we kept our finger stiff and rotated our wrist to draw the shape (wrist drawing). We didnt find any differences in how we approached these instances when we raised their difficulty levels.

Example 4 is a single player handheld console instance, and so can be interacted with using either our finger or the DSs stylus. We didnt find a difference between using our finger or a stylus. We believe this is because the motion were making is entirely controlled by our wrist; because we are drawing shapes (often curved or circular), there is no significant experiential difference between using our finger or a stylus. This lack of noticeable difference might be because of the screen is relatively small, and so the turning motion of our wrist can cover the whole screen. It is possible that on a larger touchscreen like the iPad, we would see different motions to cover the size, and therefore a potential difference in abilities when using our fingers versus a stylus.

Examples 2 exists on handheld console in both a single and multiplayer context. The single player context played identically to example 5. We did not find the multiplayer context to significantly change the abilities used in the challenge, or their relative importance. We were more aware of our own performance, and the performance of others, in the multiplayer context — but this did not seem to translate to significant extra load on our attentional abilities.

For all examples wrist drawing was the limiting factor at play (level 4) as it determined our ability to quickly draw the shape. Perception was noticeably used (level 2) as we needed it to know what shape we were drawing. Attention was used, but not important (level 1).

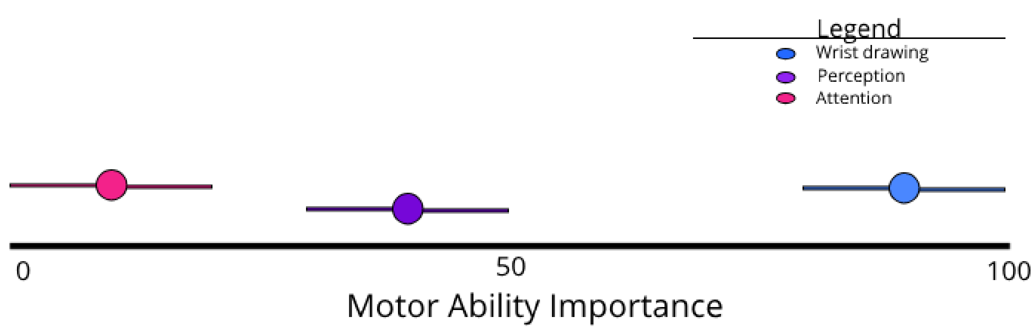

Figure 1.9: Summarized analysis for Rapid Shape Drawing with error bars of ±10

In summary, rapid shape drawing used the following abilities: wrist drawing (90), perception (40), and attention (10).

1.9 Rapid Controller Rotation Challenges

Genre	Game	Instance	Controller	Motor	Cognitive
Sports and Party game	Mario and Sonic at the Rio 2016 Olympic Games [185]	Hammer Throw	HMC (Wii Remote)	Wrist drawing, Arms rotating, Arms pushing	Perception (minimal), Attention (minimal)
Sports and Party game	Mario and Sonic at the Rio 2016 Olympic Games [189]	Hammer Throw	HC (3DS)	Wrist tilting	Perception (minimal), Attention (minimal)
Action and rhythm game	WarioWare: Smooth Moves [102]	Stir Crazy	HMC (Wii Remote)	Wrist drawing, Arms rotating, Arms pushing	Perception (minimal), Attention (minimal)
Action and rhythm game	WarioWare: Smooth Moves [102]	Wiggle Room	HMC (Wii Remote)	Wrist drawing, Arms rotating, Arms pushing	Perception (minimal), Attention (minimal)

Table 1.9: Examples of rapid controller rotation in commercial games

HC = Handheld Console, HMC = Handheld motion controller

Through these examples we found 3 motor and 2 cognitive abilities. We order them in terms of importance as:

1. Wrist drawing
2. Arms rotating
3. Arms pushing
4. Perception
5. Attention

Example 2 exists in a handheld console context. Since we have to hold the controller between two hands, we were extremely limited in our potential movements. We used wrist/forearm tilting movements to rotate the entire console. In general, this action was very cumbersome as the shape and intended hold for the 3DS did not make it easy to rotate. Holding the device in one hand would not have been effective, as its too heavy and doesn't feel secure when rotating it with one hand.

Though example 2 differs in motor abilities from the other examples, we are hesitant to say it belongs in its own category. It seems as though this instance was an experimental use of the mechanic for handhelds, and hasnt been applied since — likely due to the awkwardness of the movement for that grip. Since it did not seem to merit its own category, it seemed important to catalogue its existence here as it technically meets all the definitions of the challenge. However, since its motor abilities are not congruent with the other examples, we omit it from this analysis.

Examples 1, 3, and 4 [1] exist in a handheld motion controller context. These instances are all played with the controller oriented vertically and held in a single hand. For these instances we found that there were two ways in which we rotated the controller; exclusively with our wrists (wrist drawing), or through a combination of arm movements (arm rotating and pushing). We found that when we naturally started to play these instances our default was to use our wrist as the pivot point for rotating the controller. This allowed us to make a lot of rotations very quickly, but we experienced a lot of wrist fatigue when repetitively playing this challenge. As our wrist became fatigued we found that we started to off load the rotation work onto our arms/shoulders, making larger and less efficient rotations but still interacting with the game. This movement is a combination of the arms rotating and arms pushing motion, as it incorporates both shoulder rotations and flexion/extension in order to make a circular motion.

All examples used perception and attention minimally. Perception was used to be aware of the arm movements, and may have subconsciously factored into motor adjustments due to fatigue. Attention was used to focus on the game.

In general we did not find many examples of this challenge, hence multiple examples from the same game. This type of challenge seems more common with handheld motion controllers, which makes sense as they seem to be more comfortable and easy to rotate quickly. As such we feel wrist drawing is the limiting factor (level 4) for this challenge, since it is the default movement when asked to rotate a handheld motion controller in a single hand. We list arms rotating and pushing as important, but not limiting (level 3) since they act almost like a backup function to completing the task. We also believe that arm movements are likely to be limiting factor movements for children playing this challenge type, as their fine motor abilities develop slower than their gross motor and so they rely more heavily on gross motor interactions. Perception and attention are both used, but not important abilities (level 1) with perception ranking slightly higher in terms of importance.

In summary, rapid controller rotation uses the following abilities: wrist

[1] Wiggle Room is intended to be played using the “Big Cheese” form (Wii Remote held at your hips as if standing akimbo), the rotation would then come from rotating your pelvis as if hula hooping. However, the game also allows you to just rotate the controller with your hand as full body motion is not being examined by the system.

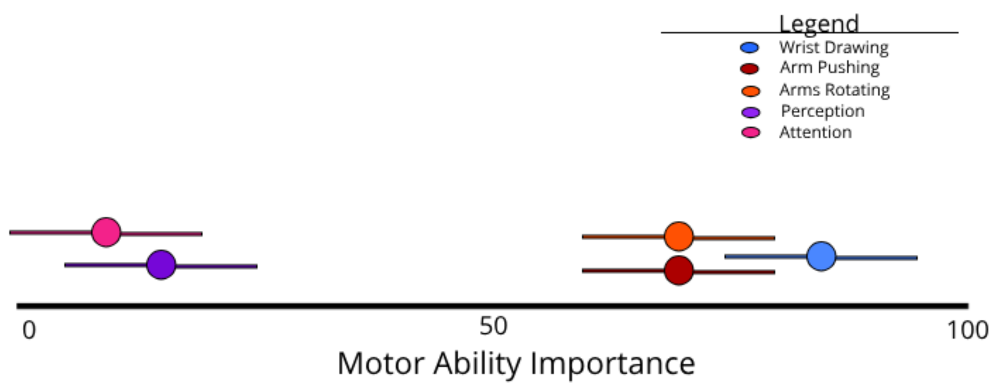

Figure 1.10: Summarized analysis for Rapid Controller Rotation with error bars of ±10

drawing (85), arms rotating (70), arms pushing (70), perception (15), attention (10).

1.10 Rapid Controller Shaking Challenges

Genre	Game	Instance	Controller	Motor	Cognitive
Action and rhythm game	WarioWare: Smooth Moves [102]	Shakedown	HMC (Wii Remote)	Wrist/forearm shaking, Arms swinging	Perception (minimal), Attention (minimal)
Platformer	Donkey Kong Country: Tropical Freeze [174]	Roll attacks	HMC (Wii Remote)	Wrist/forearm shaking, Arms swinging	Perception (minimal), Attention (minimal)
Action-adventure/Survival	The Last of Us [134]	Flashlight	Standard (Dualshock 3)	Wrist/forearm shaking, Arms swinging	Perception (minimal), Attention (minimal)

Table 1.10: Examples of rapid controller shaking in commercial games

HMC = Handheld motion controller

Through these examples we found 2 motor and 2 cognitive abilities. We order them in terms of importance as:

1. Wrist/forearm shaking
2. Arms swinging
3. Perception

4. Attention

Example 1 exists in a single player vertical handheld motion controller context. In this instance we found that quick up and down movements using the elbow as the pivot (wrist/forearm shaking) were the most effective way to play. As we fatigued, we incorporated more movement from the shoulder (arms swinging) to compensate for the speed.

Example 2 exists in a single player horizontal handheld motion controller context. In this instance we held the controller between our two hands and used simultaneous up and down movements pivoting at the elbow (wrist/forearm shaking) to shake the controller. Like example 1, we found we incorporated more shoulder/arm movements as we fatigued. Example 2 can exist in a multiplayer context; however, this does not change the abilities used in the example as the multiplayer mode is cooperative not competitive.

Example 3 exists in a single player standard controller context, which resulted in an identical experience to example 2.

All examples used minimal amounts of perception and attention. Perception was used to understand what was being done, while attention was used to focus on the task.

From these examples we can see that this challenge exists in many different contexts and genres. Though we had a hard time finding more challenges, this is likely due to our own limitations in game familiarity. It seems likely that this challenge would also exist in a handheld console, and smartphone/tablet context as those devices often come equipped with accelerometers to recognize device motion. Though these instances were single player, it is likely that this challenge exists in competitive multiplayer arenas as well.

We find that wrist/forearm shaking is the limiting factor (level 4) in this challenge, since it is the most common interaction method between examples. Arms swinging is an important, but not limiting factor (level 3) as it becomes more important as we become fatigued. It is likely that arms swinging would be the limiting factor if the player is a younger child, as their fine motor abilities (wrist/forearm shaking) are not as well developed as their gross motor abilities. Perception and attention are both used, but not important (level 1).

In summary, rapid controller shaking uses the abilities: wrist/forearm shaking (85), arms swinging (70), perception (10), attention (10).

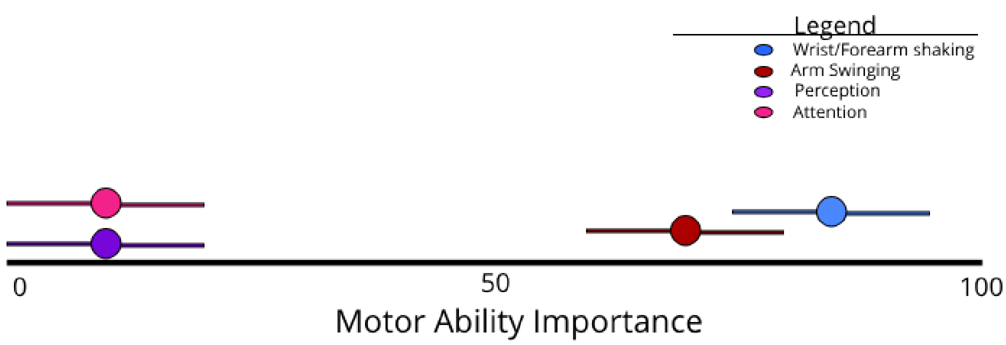

Figure 1.11: Summarized analysis for Rapid Controller Shaking with error bars of ±10

Chapter 2

Reaction Time Challenges

This chapter explores the analysis of motor abilities used in reaction time challenges. The analysis of each challenge is summarized in tabular form at the beginning of each section. The content of these sections will deal with explaining how the cell numbers are derived from the tabular analysis.

Before moving on to individual analyses, we need to address a meta issue regarding cognitive abilities. The current challenge division is because they use different cognitive and motor abilities, or use the same abilities in differing amounts. This was obvious for the speed challenges where motor abilities were the limiting factor, but becomes harder to illustrate as we move into challenges whose limits are cognitive. Due to our scope (Sections 0.3 0.2) we only outlined motor abilities in detail, leaving the cognitive aspects as broad strokes. This creates an issue with cognitively limited challenges, like Reaction Time, as it appears they all have the same ability breakdown (high use of perception and attention, low use of motor abilities).

In reality "perception", "attention", and "memory" are much more nuanced and comprised of several distinct but interconnected abilities. To give a generally accepted example of this, we know memory has two major components: short term and long term memory. We originally explored and catalogued all of these nuanced abilities, as we had with the motor abilities chapter. With this understanding of cognitive abilities we separated the Reaction Time challenges; however, we soon realised that outlining a gameplay, motor, and cognitive model as well as explaining how they would be used together to analyse games created a much too large project (or more specifically, a much too long paper for anyone to read). Therefore, in the interest of having a readable thesis, we decided to cut the cognitive abilities for future work. Though we scoped cognitive abilities out of our current work, we did not think it wise to combine all the reaction time challenges that we knew would be separated once we got to discussing the cognitive abilities. So, moving forward, if two challenges have an identical ability graph remember that it is because different aspects of those cognitive abilities are being used.

2.1 Simple Reaction Time Challenges

Genre	Game	Instance	Controller	Motor	Cognitive
RPG	Final Fantasy X [205]	Swordplay	Standard (Dualshock 2)	Finger pressing	Perception (limiting), Attention (important)
RPG	Final Fantasy XII [201]	Quickenings	Standard (Dualshock 2)	Finger pressing	Perception (limiting), Attention (important)
Action, stealth, adventure	Resident Evil 4 [36]	Shaking off villager	Standard (Gamecube)	Finger swiping	Perception (limiting), Attention (important)
Stealth, adventure	Silent Hill: Shattered Memories [48]	Escaping Raw Shocks	HM (Wii Remote)	Wrist/Forearm shaking, Arms swinging	Perception (limiting), Attention (important)
Party	WarioWare: Touched! [101]	Up for Grabs	HC (DS)	Wrist tilting (stylus), Finger pressing	Perception (limiting), Attention (important)

Table 2.1: Case studies for single input button mashing

HM = Handheld Motion, HC = Handheld Console, RPG = Roleplaying game

Through these examples we found 5 motor (finger pressing, finger swiping, wrist/forearm shaking, arms swinging, wrist tilting) and 2 cognitive (perception, attention) abilities.

Unlike the previous analysis sections we cannot begin with an ordering of these abilities in terms of importance. This is because, when examining the examples, we find that the motor abilities we have listed are not important to the success of this challenge. Example 1 mimics a standard reaction time carnival game, where the player is expected to press a single button when the indicator lines up with a certain marked spot on the screen. Example 2 requires players to quickly discern whether one of 3 button options are available to press before the timer expires, all options produce the same result (with different aesthetics) and so there is no element of choice analysis in the process. If no buttons are available the player must press the designated R2 button to "shuffle" their options. Example 3 tasks the player with wiggling their thumbstick left and right (finger swiping) to shake an attacking NPC off of them. Example 4 requires the player to shake their vertically held handheld motion controller to escape an attacking NPC. Example 5 requires the player to tap an object as it moves across the screen. We could find several more examples that fit the description of "simple reaction time" with completely different motor abilities used.

The common denominator between these tasks is the strain it places on the player's perceptional and attentional abilities. This challenge is inherently about simple actions needing to be performed quickly and without warning. As such, the hardware and player contexts of the examples are irrelevant in terms of difficulty or importance.

We therefore choose to list importance as follows:

1. Perception
2. Attention
3. Motor Ability

By leaving the third item generic, we indicate that it doesn't matter what the actual ability is, just that it falls at a significantly lower level of importance in comparison to perception and attention. We find that for our research we list the generic motor ability at noticeably used (level 2), in the lower-middle range of the interval to account for it not being a significant factor.

Regarding the cognitive abilities, it is hard to distinguish whether perception or attention is more taxed in this challenge. As reaction time is an internal process, we are not able to study its individual components — we can only measure reaction time, and assume that the perceptional and attentional processes are taking similar amounts of time to complete. For our research we list perception as the more taxed (limiting factor level 4) ability because it is only on stimuli that the player is moved to act. While they must be intently focused, therefore using a significant amount of attentional resources, if nothing appears they still can't complete the challenge. We therefore list attention as important, but not limiting (level 3), and make the assumption that it is at the higher end of this interval to account for it potentially being the limiting factor ability.

In summary, simple reaction time uses the following abilities: perception (90), attention (75), motor ability (25).

2.2 Combat-based Reaction Challenges

2.2.1 Single Input Attacks

Through these examples we found 1 motor (finger pressing) and 3 cognitive (attention, perception, memory) abilities. We order them in terms of importance as follows:

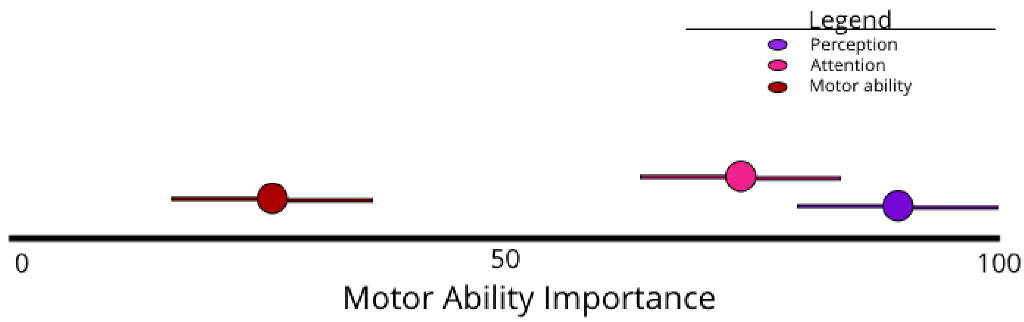

Figure 2.1: Summarized analysis for Simple Reaction Time with error bars of ±10

1. Attention,
2. Perception,
3. Memory,
4. Fine Motor Abilities

We previously argued for Simple Reaction Time challenges that the actual motor ability does not matter since it is not the defining trait of this challenge. In the case of Single Input Attack challenges, we believe that it does create a significant difference. Generally, we know that the reaction processing loop has 3 parts: recognize stimuli, plan response, enact plan; motor abilities are used in the third part of the loop. In the same way that having multiple options increases the planning stage of the loop, the type of motor reaction (fine vs. gross) affects the enacting stage as on average fine motor abilities take less time to enact that gross motor abilities.

Generally, we found examples that exclusively use button pressing (finger pressing) as the motor response. It is possible that there are games that use this challenge with alternate control schemes, but we are not currently aware of any. If this challenge appeared on smartphones/tablets, it is possible that on-screen controls would emulate this type of button pressing. This would open the door to using stylus accessories, which would result in different fine motor abilities being used to manipulate the stylus into pressing the soft controls (wrist tilting). This type of change in motor ability would not significantly affect the enacting time of the loop. In comparison, if this was implemented on a system with a handheld motion controller the button pressing may be replaced by a movement like arm swinging. As an action arm swinging takes more time and energy than button pressing, and so would significantly affect the enacting portion of the loop. Therefore, we choose to list the motor ability

Genre	Game	Instance	Controller	Motor	Cognitive
Party	Mario Party 2[93]	Hot Rope Jump	Standard (N64)	Finger pressing	Perception (important), Attention (limiting), Memory (important)
Fighting	Super Smash Bros. for 3DS [21]	Shielding and other attacks	HC (3DS)	Finger pressing	Perception (important), Attention (limiting), Memory (important)
Action-adventure	The Legend of Zelda: A Link Between Worlds [145]	Combat	HC (3DS)	Finger pressing	Perception (important), Attention (limiting), Memory (important)
Action-adventure	The Legend of Zelda [154]	Combat	Standard (NES)	Finger pressing	Perception (important), Attention (limiting), Memory (important)
Action-adventure	The Legend of Zelda: The Wind Waker [142]	Combat and Parry	Standard (Gamecube)	Finger pressing	Perception (important), Attention (limiting), Memory (important)

Table 2.2: Case studies for single input attack

HC = Handheld Console

for this challenge as “Fine Motor Ability instead of “finger pressing to show that any fine motor ability is usable for this challenge type.

We previously stated that the cognitive difficulty of this challenge increases with the number of “attack” options the player had to choose from (Section 2.2.1). This understanding of the difference in complexity between choosing an action, and the motor ability that enacts the action is how we determined “Fine Motor Ability” is the lowest ranked ability for this challenge.

For cognitive abilities, we believe that attention is the most important and limiting factor (level 4) for this challenge. Attention is used to pick up on the actions of the stimuli (often enemies) and to know when to make a move. Success in example 1 relies on the player to be giving the game their full attention, as the movement speed and size of the flaming jump rope can change at any time. Without paying attention to this, the player would not be able to time their jump correctly, or to choose whether they should be making a quick short jump or a longer higher jump. Example 5s parry option is similar, where the player must be paying attention for when the parry command becomes available and to react at the appropriate time (early parries may miss higher level enemies, while waiting too long will cause you to miss the parry option).

Perception is an important, but not limiting ability (level 3) because it is the skill used to recognize the type of stimuli. Without the information gained from perception, the player would not be able to come up with an appropriate response. Example 2 relies on perception as some incoming attacks cannot be blocked by the shield [263]. Therefore, the player would know that they should not use their shield and can instead choose a different option. We could argue that perception is the most important ability in examples 3, and 4; since both are presented in a top down perspective, the player can see the enemy placement across the whole screen. Therefore, it is perception, and recognizing which enemies are on screen and where they are that dictates success. However, we believe that attention also plays a non-trivial role in those instances as the player must constantly keep track of enemy movements and the distance between themselves and their enemies. We therefore reinforce that attention as the most important, limiting factor, but believe that perception is a highly important, but not limiting ability.

Memory is an important, but not limiting ability (level 3) because it is used in all examples when recognizing stimuli and planning responses. In relation to perception, memory is ranked lower because it is not constantly being taxed in the same way. Since this challenge can generally be described as a selection task, we can model the time selection will take using the Hicks-Hyman Law. A future experiment can test this challenge by modeling an experiment like those that explore the Hicks-Hyman Law.

In summary, single input attack challenges use the following abilities: attention (90), perception (80), memory (60), and fine motor abilities (40).

2.2.2 Multiple Input Attacks

Though we only list 4 examples, this challenge is extremely common in all games with sufficiently deep combat mechanics. All given examples are instances of larger franchises that use the same mechanics, and are also part of larger genres that use similar mechanics.

At first glance, it seems out of place to list fighting game moves as examples of this challenge since other challenges like “Learning combination moves” and “Attack chaining” fall more in line with what we think of as fighting game gameplay. As previously mentioned (section 0.3.3), combat generally is a composite challenge that focuses on cognitive and motor play at different levels. “Multiple Input Attacks” are common at the motor level of play, as

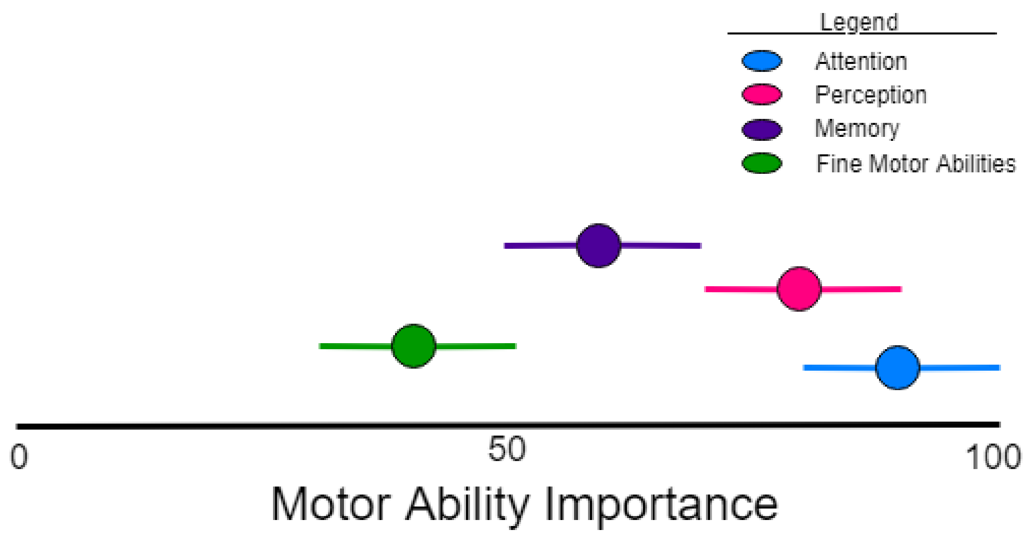

Figure 2.2: Summarized analysis for Single Input Attacks with error bars of ±10

this challenge only exists in the context of combat (section 2.2.2). Consider example 4, the "Slip Up" move performed as Phoenix Wright is executed by simultaneously pressing the thumbstick in the forward direction and medium punch button. The actual execution of this move is not excessively complex in either the motor or cognitive context, to the point that novice player can use it without significant practice. The difficulty in a competitive scenario is understanding the specifics of the move (range, damage, effectiveness in different situations) and having the strategic understanding and muscle memory to incorporate it into a larger attack chain. These are distinct challenges that, while often occurring together, can be separated and so we need to be careful not to conflate them.

Through these examples we find 2 motor (finger pressing, finger swiping) and 3 cognitive (perception, attention, memory) abilities. As with Single Input Attacks (section 2.2.1) we find that while the category of motor ability is important, the specific abilities used are not. Therefore, we replace these specific motor abilities with the more generic "fine motor abilities. We order the abilities as follows:

1. Perception
2. Attention
3. Fine Motor Abilities
4. Memory

Genre	Game	Instance	Controller	Motor	Cognitive
Fighting	Super Smash Bros. for 3DS [21]	Dodging	HC (3DS)	Finger pressing, finger swiping	Perception (limiting), Attention (important), Memory (noticeable)
Action-RPG	Bloodborne [68]	Rolling	Standard (Dualshock 4)	Finger pressing, finger swiping	Perception (limiting), Attention (important), Memory (noticeable)
Action-adventure	The Legend of Zelda: Ocarina of Time [141]	Jump Slash	Standard (N64)	Finger pressing, finger swiping	Perception (limiting), Attention (important), Memory (noticeable)
Fighting	Ultimate Marvel vs. Capcom 3 [35]	Phoenix Wright - Slip Up	Standard (Xbox 360)	Finger pressing, finger swiping	Perception (limiting), Attention (important), Memory (noticeable)

Table 2.3: Case studies for Multiple Input Attacks

HC = Handheld Console

Performance in this challenge relies on a decent amount of fine motor coordination. This is usually implemented as combining a button press with a directional push on the thumbstick, as seen from these examples. Other potential examples include simultaneously pressing two or more buttons, such as Phoenix Wrights "Investigate" move in Ultimate Marvel vs. Capcom 3 [35] which requires pressing a punch button and the special move button at the same time. Regardless of specific input, this coordination effort creates a measurable difference in motor and attentional load. Therefore, we list fine motor abilities at a high spot in the noticeably used interval (level 2). We keep it in the noticeably used category because most examples feature coordinating only two inputs; as more actions need to be coordinated the strain on the players motor abilities will increase, potentially moving it into an important, but not limiting role (level 3).

We find that memory is not as important in this section as it was in Single Input Attack challenges (section 2.2.1) as there are often a fewer selection of inputs to choose from. This is likely a game design decision — as the moves become more demanding, you dont want to overburden your player with choice. Therefore, less memory should be used remembering how to

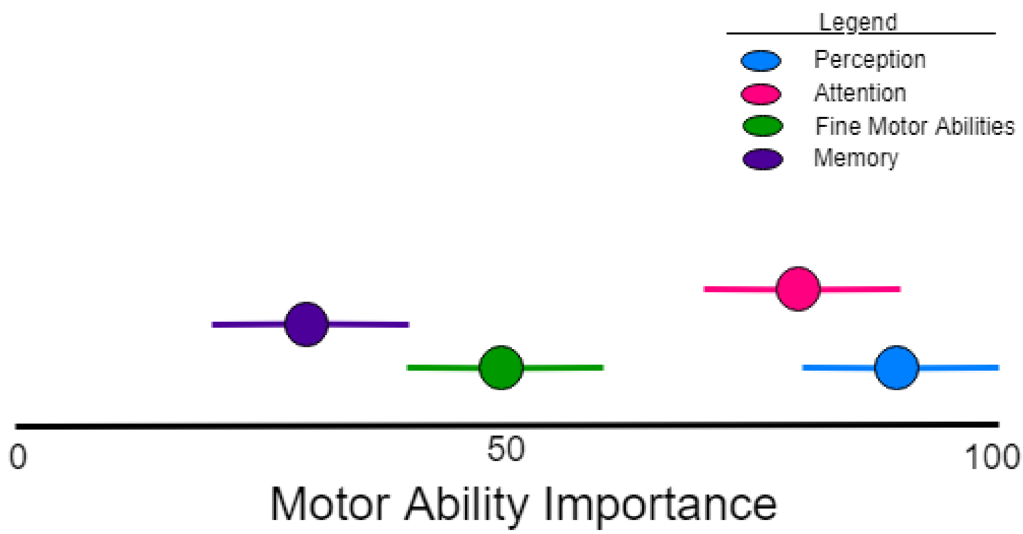

Figure 2.3: Summarized analysis for Multiple Input Attacks with error bars of ±10

perform this challenge. The apparent issue is that memory load has been off-loaded from this challenge to the Advanced Combat challenges that deal with the learning of moves and their strategic applications. It's possible that with further study we may find memory to play a larger part in this challenge. We therefore list memory at a low spot in the noticeably used, but not important (level 2) interval.

Perception and attention are still quite important for this challenge as many the moves for these challenges are often executed in response to an opponents movements (examples 1, 2, and 3). Example 1 requires the player to understand what move the opponent is using against them, its intricacies (range, damage, can it be blocked), and then choose how to dodge (spot dodge, movement dodge, etc). Example 2 works in the same way. Between these two abilities we believe that perception is the more important because the planning phase hinges on recognizing what movement the player should be responding to. Therefore, we list perception as the limiting factor (level 4), and attention as an important, but not limiting ability (level 3).

In summary, multiple input attack challenges use the following abilities: perception (90), attention (80), fine motor abilities (50), and memory (30).

2.2.3 Motion Control Attacks

Through these examples we found 4 motor (arm swinging, arm rotating, arm positioning, leg moving) and 3 cognitive (attention, perception, memory) abilities. We order them in terms of importance as:

Genre	Game	Instance	Controller	Motor	Cognitive
Action-adventure	The Legend of Zelda: Skyward Sword [144]	Combat	HM (Wii Remote)	Arm swinging	Perception (important), Attention (limiting), Memory (noticeable)
Action-adventure	No More Heroes [79]	Combat	HM (Wii Remote)	Arm swinging	Perception (important), Attention (limiting), Memory (noticeable)
Action-adventure	Wii Sports [143]	Tennis	HM (Wii Remote)	Arm swinging	Perception (important), Attention (limiting), Memory (noticeable)
Action-adventure	Kung Fu Panda 2 [80]	Combat	HM (Kinect)	Arm swinging, Arm rotating, Arm positioning, Leg moving	Perception (important), Attention (limiting), Memory (noticeable)

Table 2.4: Case studies for Motion Control Attacks

HM = Handheld Motion

1. Attention,
2. Perception,
3. Memory,
4. Gross motor ability

As with Single Input Attacks we believe that the type of motor ability used matters more than the specific ability in respect to the affect on the reaction processing loop. We see in all examples that arm swinging is used to interact with the challenge, however this is likely due to the limitations of the systems the games were made for. The Wii and Wii U systems only capture the position of the Wii Remote, and by proxy only capture the players arm movements. For games made on the Kinect (example 4) we see that the whole range of gross motor abilities can be used to handle this challenge. As such, we label the motor abilities we found as "Gross motor ability and understand that generally their use extends the enacting segment of the loop.

Motion Control Attacks are like Single Input Attacks from a cognitive perspective. Attention is the limiting factor (level 4), as it dictates the flow of

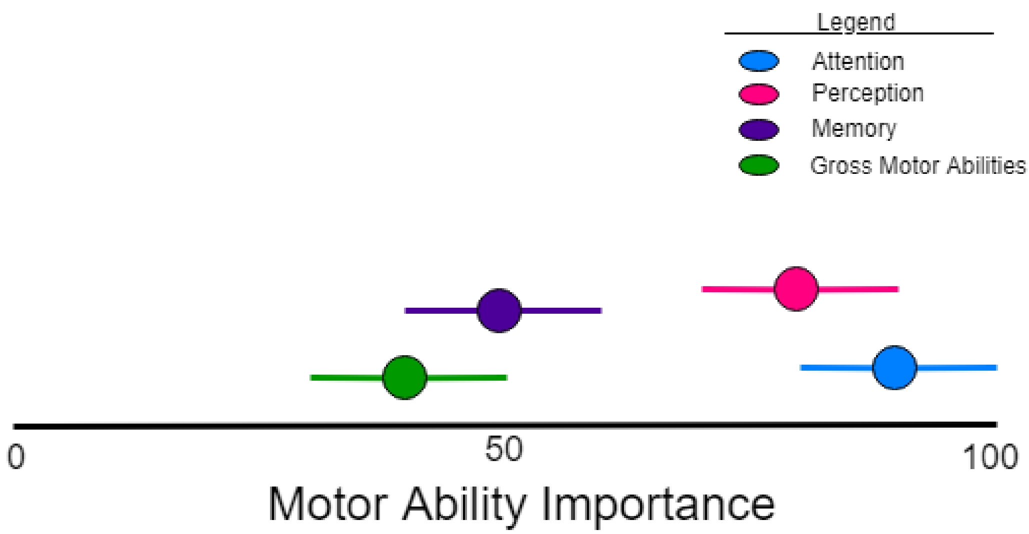

Figure 2.4: Summarized analysis for Motion Control Attacks with error bars of ±10

combat. The player is constantly waiting and responding to enemy movements (example 1, 2, 3, 4). Perception is important, but not limiting (level 3) as it is used in coming up with a response plan and determining timing for actions. Generally, memory is less important for this challenge than it is in Single Input Attacks. This is because there are fewer general options that the player can be asked to remember; we can have many buttons on a controller, each with their own action, but there are fewer full body actions a player can be asked to make. As such, memory is listed as noticeably used (level 2) but is placed higher on the intervals.

In summary, motion control attack challenges use the following abilities: attention (90), perception (80), memory (50), and gross motor abilities (40).

2.3 Movement Based Reaction Challenges

2.3.1 Fixed Time Movement

Through these examples we found 2 motor abilities (finger pressing, finger swiping) and 2 cognitive (perception, attention).

Though finger pressing is the most commonly used, we do not believe the motor ability is a defining trait of this challenge. Buttons are the most common form of interaction on a controller, so it is only natural that finger pressing is the most common ability. Since this challenge type focuses on reaction time, and not a specific input method, it is able to be implemented on any platform (as confirmed by the variety of controllers and by extension platforms we see

Genre	Game	Instance	Controller	Motor	Cognitive
Platformer	Runner 2 [72]	–	HC (Wii U)	Finger pressing	Perception (limiting), Attention (important)
Platformer	Donkey Kong Country: Tropical Freeze [174]	Auto-scroller levels	HC (Wii U)	Finger pressing	Perception (limiting), Attention (important)
Action-adventure	Kingdom Hearts 2 [204]	Gummi Missions	Standard (Dualshock 2)	Finger pressing	Perception (limiting), Attention (important)
Endless Runner	Temple Run [100]	–	Mobile (Pixel XL)	Finger swiping	Perception (limiting), Attention (important)
Endless Runner	Chrome Browser Game/T-Rex Runner [248]	–	Keyboard	Finger pressing	Perception (limiting), Attention (important)

Table 2.5: Case studies for Fixed Time Movement

HM = Handheld Motion, HC = Handheld Console, Mobile = Smartphone/Tablet

in the given examples). We therefore list these abilities in terms of importance as:

1. Perception,
2. Attention,
3. Motor ability

As previously stated (section 2.1), the cognitive elements of the reaction processing loops are not able to be studied separately. For this challenge, we understand that generally perception and attention are almost equally taxed. As previously stated (section 2.3.1), the absolute difficulty is determined by how many times the player must undergo the reaction processing loop in a short span of time (more obstacles in a shorter time makes for a harder level). From playing through these examples we determine perception to be the limiting factor (level 4), as the reaction processing loop begins at the appearance of a stimuli, and it is used by the player to estimate the time to contact with the obstacle. Attention is important, but not limiting (level 3) as our ability to maintain focus is taxed. Since our movement is controlled by the game, we do not get a break from the action during a level; any significant distraction

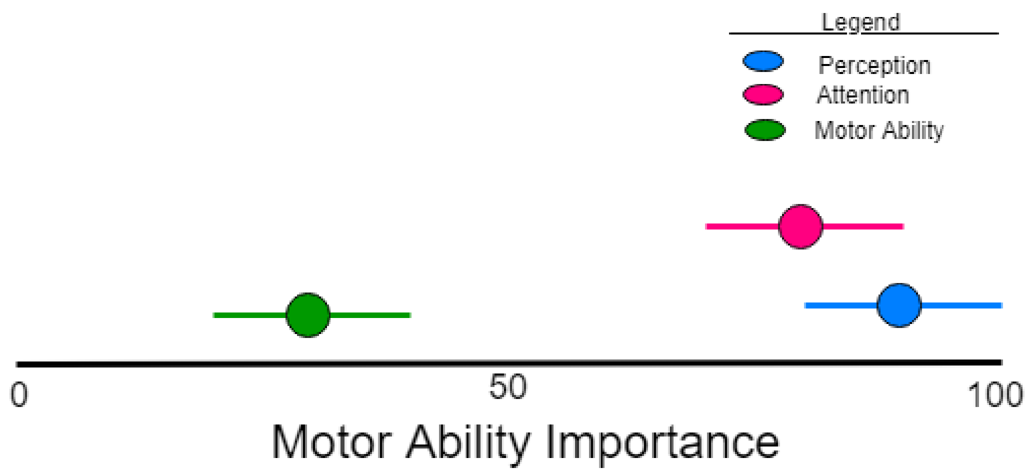

Figure 2.5: Summarized analysis for Fixed Time Movement with error bars of ±10

will result in a "game over" as we might then miss perceiving an obstacle in the game. Therefore, we place it at the high end of the interval to indicate that it may be a limiting factor upon further analysis and testing.

Though motor abilities are not the determining factor of this challenge, they are still somewhat important. Fixed Time Movement challenges often require a very quick movements in quick succession as the difficulty increases. Therefore, even though the movement itself is simple, and what movement it is doesn't really matter, the fact that its going to be done multiple times makes it a noticeably used ability (level 2).

In summary, Fixed Time Movement challenges uses the following abilities: perception (90), attention (80), and motor ability (30).

2.3.2 Variable Time Movement

Through these examples we found 2 motor (finger pressing, finger swiping) and 3 cognitive abilities (perception, attention, memory). We order these abilities in terms of importance as:

1. Perception,
2. Attention,
3. Memory,
4. Finger pressing,
5. Finger swiping

Genre	Game	Instance	Controller	Motor	Cognitive
Platformer	Super Mario Bros. [140]	–	Standard (NES)	Finger pressing	Perception (limiting), Attention (important), Memory (noticeable)
Platformer	Donkey Kong Country: Tropical Freeze [174]	Non-Auto-scroller levels	HC (Wii U)	Finger pressing, Finger swiping	Perception (limiting), Attention (important), Memory (noticeable)
Platformer	Sonic Mania [159]	–	Keyboard	Finger pressing	Perception (limiting), Attention (important), Memory (noticeable)
3D Platformer	Crash Bandicoot (part of N-Sane Trilogy) [264]	–	Standard (Dualshock 4)	Finger pressing, Finger swiping	Perception (limiting), Attention (important), Memory (noticeable)

Table 2.6: Case studies for Variable Time Movement

HC = Handheld Console

This challenge is most commonly found in the context of platformer games, where they make up the bulk of the gameplay. This is not to say that this challenge isn't used in other genres, but that as games grow more complex this challenge is likely to be found alongside other challenge types making it harder to identify. We believe that it any game that features world exploration will inevitably use some form of this challenge, when players are navigating through the world.

This challenge requires two main actions from the player: 1) move the avatar, 2) handle obstacle; how these are implemented is dependent on the available hardware. Examples 1 and 3 exist in a hardware context where only buttons exist. This meant that movement was done via finger pressing to input the direction of movement. Examples 2 and 4 exist on controllers that have a variety of input controls (analog thumbsticks, buttons, touch interface, gyroscope, accelerometer), and so avatar movement was handled by the analog thumbstick (finger swiping) while handling obstacles became tied to buttons (finger pressing). Thus, why we see a difference in the used motor abilities. It is possible that other examples with different hardware controllers would therefore use different motor abilities. We chose to leave finger pressing and finger swiping in place of a generalised motor ability because of their frequency as the used motor abilities.

Though we're not generalising the motor abilities used, we believe they are still somewhat important to the challenge. As with fixed reaction time, the speed at which the player must enter the correct input is an important factor in the success of this challenge. Therefore, we believe that generally the motor abilities used should fall in the range of noticeably used (level 2). Inside this range, we believe that generally the motor ability that causes the obstacle avoiding move is the more important ability; it is only in pushing the jump button that Mario can clear the Goomba. The motor ability that controls movement is marginally less important. We believe that a test should be devised to verify whether this is correct, as there are certain situations where it may not be true. For our work finger pressing is the more important of the motor abilities, and finger swiping is the lesser.

All examples used perception, attention, and memory. We believe perception is the limiting factor (level 4) as it is what begins the reaction-processing loop. Attention is important, but not limiting (level 3) as sustained attention is needed to handle all the obstacles over the length of the level. Memory is noticeably used (level 2) because the player must remember how to deal with different types of obstacles, and which input to use to interact with them. For example, in Donkey Kong Country: Tropical Freeze the low-level enemy Tuck comes in several varieties (Tuck, Pointy Tuck, Trench Tucks, Archies, Speedy Tucks, Painguin Tucks, etc). The regular Tuck can be defeated by jumping on them, rolling at them, or throwing something at them. Pointy Tucks wear a horned helmet, killing the player if they are jumped on. When running through a level, the player must be aware of what type of enemy is in front of them and what action are available to them. Since mistakes in the memory component of this challenge are more likely to result in death than mistakes in the motor component, we believe that memory is more important than the motor abilities. Though memory is a level 2 ability, we place it on the border of levels 2 and 3 to reflect our uncertainty in exactly how important this cognitive ability is.

In summary, variable time movement challenges use the following abilities: perception (90), attention (80), memory (50), finger pressing (35), and finger swiping (30).

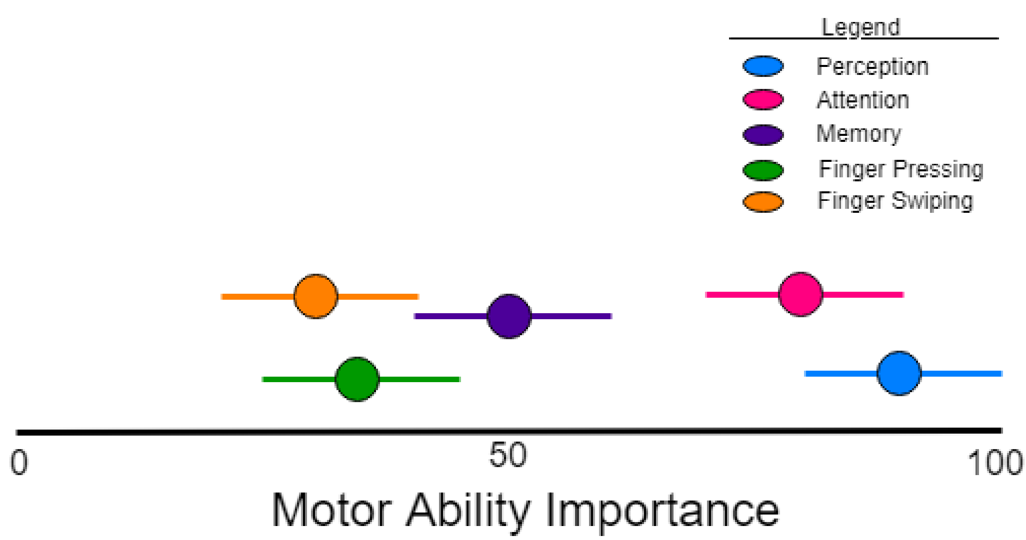

Figure 2.6: Summarized analysis for Variable Time Movement with error bars of ±10

Chapter 3

Advanced Combat Challenges

This chapter explores the analysis of motor abilities used in advanced combat challenges. The analysis of each challenge is summarized in tabular form at the beginning of each section. The content of these sections will deal with explaining how the cell numbers are derived from the tabular analysis.

3.1 Learning Combination Moves

Genre	**Game**	**Instance**	**Controller**	**Motor**	**Cognitive**
Fighting	Skullgirls [175]	Squigly - Shun Goku Saltsu	Standard (Xbox 360)	Finger pressing, Finger swiping	Memory (important), Perception (noticeable)
Fighting	Ultimate Marvel vs. Capcom 3 [35]	Vergil - Summoned Swords	Standard (Xbox 360)	Finger pressing, Finger swiping	Memory (important), Perception (noticeable)
Action-adventure, Hack and slash	DmC: Devil May Cry [136]	Cleaver	Standard (Xbox 360)	Finger pressing, Finger swiping	Memory (important), Perception (noticeable)
Action-adventure, Hack and slash	Bayonetta 2 [167]	Ground Combo	HC (Wii U)	Finger pressing, Finger swiping	Memory (important), Perception (noticeable)

Table 3.1: Case studies for learning complex combination moves

HC = Handheld Console

Through these examples we found 2 motor and 2 cognitive abilities. We order them in terms of importance as:

1. Finger pressing,

2. Finger swiping,
3. Memory,
4. Perception

Without question, the physical ability of executing these combos is the limiting factor of this challenge. As players get more advanced at the game executing combos becomes muscle memory, and sufficiently easier however, for the average player the precise motions and performance of the moves is the barrier to entry. From our examples finger pressing (press buttons) and finger swiping (maneuver thumbstick) are the foremost motor abilities.

This highlights a blindspot in our analysis; since weve focused our work on the "average" player using "common" controllers, this analysis has not captured the wrist and forearm movements that would be central to playing fighting games on a fight stick or arcade cabinet. As competitive gaming is becoming more popular, and retro-nostalgia is bringing back the popularity of arcades, fightsticks and arcade style controllers are becoming more accessible to the average player. As such, we believe that our ordering of our abilities can be made slightly more generic to pre-emptively handle future controller changes. Since finger swiping and any wrist and forearm movements are meant to accomplish the same input action (maneuvering the thumbstick or joystick), we abstract finger swiping to be "secondary fine motor ability". This makes our new listing:

1. Finger pressing,
2. Secondary Fine Motor Ability,
3. Memory,
4. Perception

Though we've solved the controller issue, we still must decide which ability is most important to the success of this challenge. We believe that the bulk of the motor difficulty for traditional fighting games (example 1, and 2) comes from precisely executing non-intuitive thumbstick motions such as the Z-shaped "Dragon punch" (forward, down, down-forward). When learning moves in the tutorial and practice areas, we found that memorizing which buttons to press was easy but making sure that the analog input was precise (e.g. making sure we didn't rotate it too much, or that the input direction was registering correctly) was our barrier to performing moves. From these examples the secondary fine motor ability would be the limiting factor. In comparison, for combat in examples 3, and 4 we didn't have to worry about the precision at all since our combat was completely controlled by buttons. From these examples finger pressing would be the limiting factor.

The obvious difference between the examples is length of combos. Theoretically a longer combo should be more difficulty both cognitively (more to remember) and physically (more inputs and quicker inputs to execute). Though we agree with that intuitively, we dont believe that length informs us as to which ability is more important. Examples 3 and 4 the longest combos in their games at 6 (Y, Y, pause, Y, Y, Y) and 7 (punch, punch, punch, kick, kick, pause, punch) steps respectively. However, example 2 requires only 5 steps (dragon punch, attack, attack, attack and special, attack and special) and is still significantly more difficult to execute than examples 3 and 4.

The larger issue of difficulty seems to be from input variation. Games in the vein of examples 3 and 4 often have combos that require repetitive pressing of one or two buttons with delays in between. In comparison fighting games rely on more variation in their combos such as multiple thumbstick movements and various button presses. Example 1 requires 4 thumbstick movements interspersed through 10 button pushes (quarter circle right, punch, quarter circle right, kick, light punch, light punch, left, light kick, heavy punch, light kick, light kick, left, light punch, heavy kick) for a total of 14 inputs. Since the variation comes from the addition of thumbstick movements the secondary fine motor ability that controls thumb/joystick movement seems more important. However, there are only a limited number of moves that incorporate thumbstick movement the degree of example 1, and so it would seem arbitrary to decide that thumbstick movement is more important than pressing the attack buttons. As well, example 1 is playable on a keyboard only set-up, meaning that the thumbstick maneuvers become key presses, negating the difference between the abilities.

Realistically at this point, it is impossible for us to decide which ability is more important than the other. We would need to conduct a more focused study of fighting games with players of different fine motor abilities to even begin understanding that. Since we believe there is only a single limiting factor per challenge, we choose to leave finger pressing as the listed limiting factor (level 4). This is because finger pressing is the most frequently used as it exists across all examples. We then list the secondary fine motor ability as important, but not limiting (level 3), but position it at the upper edge of the interval such that the error bars cross into limiting territory. It is entirely possible that future research could find that our assumption that there is only one limiting factor per challenge is entirely wrong — studying this challenge in depth would be ideal at testing that assumption.

Memory plays a large part in this challenge as well, as the player must memorize the steps to execute the move and recall them during the execution. From a naive viewpoint, it would be easy to assume that the limiting factor

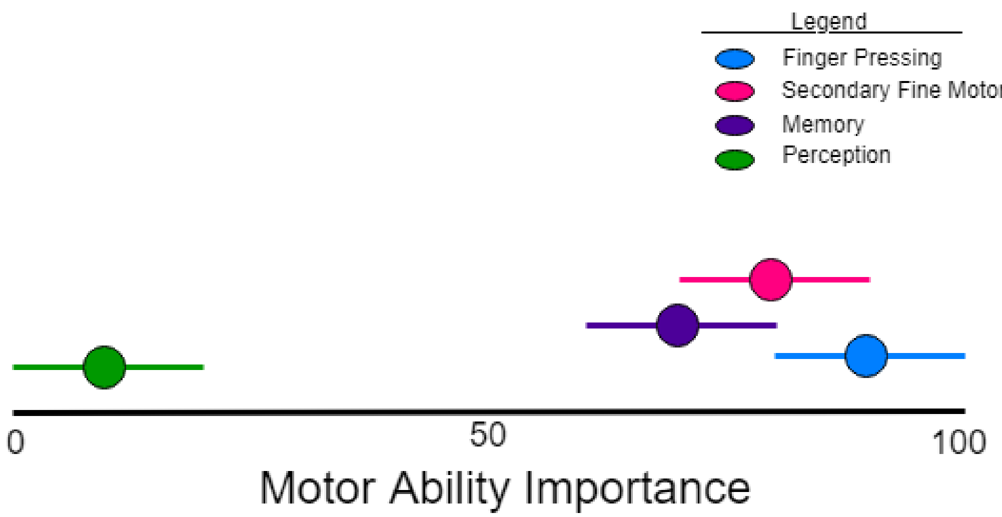

Figure 3.1: Summarized analysis for Learning Combination Moves with error bars of ±10

in "learning" would be memory, as the player must memorize long strings of input. Generally, when encountering this challenge for the first time the player relies very heavily on their memory as theyre still in the memorization stage of learning. With practice, the execution of these moves becomes muscle memory, and as such the cognitive load on the player decreases as they are not actively attempting to remember all the inputs. There is a measurable spot where the difficulty switches from cognitive to motor-focused as a function of practice. Generally, we find that this intersection point occurs very soon after practice starts. This switching point is what makes this a motor-focused challenge, as most players will find the physical skills to be limiting. Though memory becomes less important with time, we believe that it plays a significant enough role to be an important, but not limiting ability (level 3).

Perception generally is only used to recognize what inputs the player is being asked to make. As such, we believe it is used, but not important (level 1).

In summary, Learning Combination Moves uses the following abilities: Finger pressing (90), Secondary fine motor ability (80), memory (70), perception (10).

3.2 Attack Chaining

Through these examples we found 2 motor (finger pressing, finger swiping) and 3 cognitive abilities (memory, perception, attention). We order them in terms of importance as:

1. Finger pressing,
2. Finger swiping,
3. Perception,
4. Attention,
5. Memory

The variety of examples available indicates that this challenge type is common across genres and platforms. Though this challenge type exists on several different input types, we see that it relies exclusively on buttons (finger pressing) and thumbsticks (finger swiping). In cases such as example 6, even for touchscreen exclusive inputs we find that the game provides on screen controls to mimic holding a controller in your hand. Even controls with motion capabilities favour button and thumbstick inputs over larger motions (example 7). This is potentially due to the quick paced nature of combat that calls for multiple inputs being made simultaneously or nearly so. Alternatively, it could be due to the potential fatigue of using a different control scheme. It is entirely possible that there are games that use this challenge type that we have not catalogued here due to our own limited knowledge of existing games.

Regarding finger pressing and finger swiping, we run into the same issues discussed in the Learning Combination Moves section (section 3.1). Finger swiping is too specific, and limits discussion around fighting games that use fight sticks (wrist movements). We adopt the same solution from LCM (section 3.1) and instead abstract finger swiping to "secondary fine motor ability. In terms of ordering the importance of the different motor abilities, we choose to place finger pressing above the secondary fine motor ability. As we previously discussed (section 3.1), this decision is somewhat arbitrary, and the actual importance of each ability requires more extensive study. This means our ordering is now:

1. Finger pressing,
2. Secondary Fine Motor Ability,
3. Perception,
4. Attention,
5. Memory

Memory is trivially used to remember attack execution and basic information about how attacks combine. It is possible that this becomes more

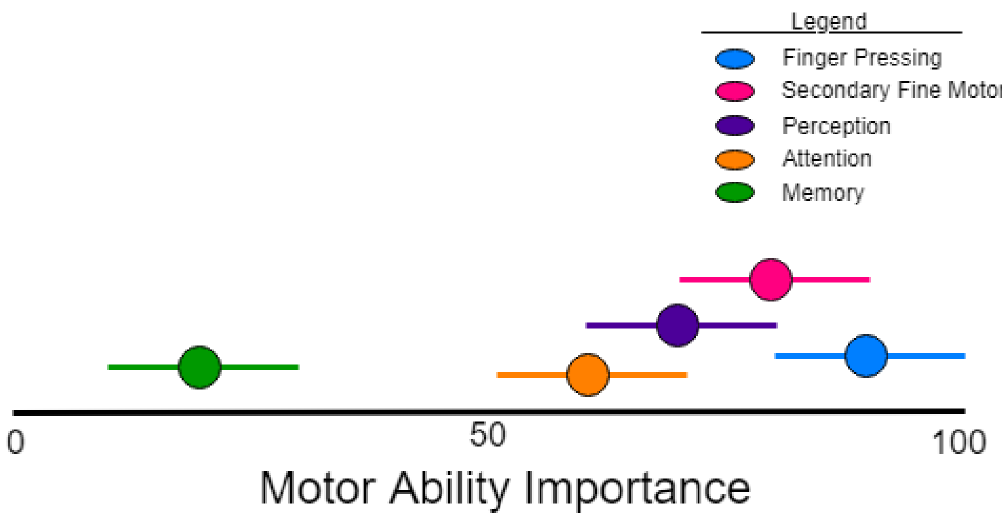

Figure 3.2: Summarized analysis for Attack Chaining with error bars of ±10

important for high level play once the idea of strategy, stun-locking, and mechanics exploitation comes into play. To understand that we need more information regarding the different cognitive abilities, and how they interact. For the moment, we list memory as being used (level 1).

Perception and attention are more important in this instance due to the reactionary nature of fluid combat. In LCM the player is focused on perfecting the execution of the move, not necessarily how to effectively apply it in combat. With attack chaining, the player needs to not only be able to perform the various combos, but also to read and react to what their opponent is doing. As with motor abilities, it is difficult to separate which of these cognitive abilities is more important. With work that explores the cognitive depth of these challenges we could understand how these interact and what other systems they use. Currently we list both abilities as being important, but not limiting (level 3), and settle with placing perception above attention to account for the fact that perception marks the beginning of the reaction processing loop.

In summary, attack chaining uses the following abilities: Finger pressing (90), Secondary fine motor ability (80), perception (70), attention (60), and memory (20).

Genre	Game	Instance	Controller	Motor	Cognitive
Fighting	Super Smash Bros. for 3DS [21]	Combat	HC (3DS)	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)
MMORPG	Black Desert Online [163]	Combat	Keyboard and Mouse	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)
Action-adventure	Kingdom Hearts 2 [204]	Combat	Standard (Dualshock 2)	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)
Action-adventure RPG	Final Fantasy XV [203]	Combat	Standard (Dualshock)	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)
Fighting	Ultimate Marvel vs. Capcom 3 [35]	Combat	Standard (Xbox 360)	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)
Action-adventure, Hack and slash	Devil May Cry 4: [34]	Cleaver	Standard (Xbox 360)	Finger pressing, Finger swiping	Memory (important), Perception (noticeable)
Fighting	Super Smash Bros. Brawl [87]	Combat	HMC (Wii Remote)	Finger pressing, Finger swiping	Memory (used), Perception (important), Attention (important)

Table 3.2: Case studies for learning complex combination moves

HC = Handheld Console, HMC = Handheld Motion Controller, MMORPG = Massively multiplayer online role playing game

Chapter 4

Compiled Analysis

To get a "big picture" look at the data, we organized it into a colour-coded table:

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.1: Hardware independent analysis of challenges with respect to basic cognitive skills.

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90			50		26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input					40				
Multiple input					50				
Motion control attacks									
Fixed time					30				
Variable time	35	30							
Simple Reaction Time					25				
Learning complex combos.	90				80				
Attack Chaining	90				80				

Table 4.2: Hardware independent analysis of challenges with respect to hand specific fine motor abilities

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input	40			
Multiple input	50			
Motion control attacks				
Fixed time	30			
Variable time				
Simple Reaction Time	25			
Learning complex combos.				
Attack Chaining				

Table 4.3: Hardware independent analysis of challenges with respect to the remaining fine motor abilities

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.4: Hardware independent analysis of challenges with respect to gross motor abilities

Each coloured cell represents a used ability, with the colours representing which level of importance it falls under. We read from each row the set of abilities used in conjunction to complete the row challenge (i.e. set of coloured cells). Multi-column cells represent that any of those motor abilities can be used in the completion of this challenge.

In doing this we realised that the multi-column cells, which are a result of our abstractions in the analysis, make it difficult to understand details of the relationship between challenges and motor abilities. For example, both Fixed Time Movement and Simple Reaction Time span all the motor abilities this obscures the actual details of motor challenges for these challenges by implying that every motor ability can or is being used in every instance of this challenge. What we instead want to convey is that some subset of these abilities is being used for any particular instance. Since understanding the details of the challenge-motor relationship is the purpose of this work, we needed to address this issue. We realized this issue is borne from trying to make the data controller-agnostic; controllers are built with specific functionality, and therefore specific motor interaction capabilities, which the game designers must consider when developing a game for a specific platform (and by association a specific controller type). We therefore decided to separate the data by controller type, with the hope that it would give us more detailed information.

4.1 Standard Controllers

Isolating the standard controller data, we end up with the following:

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80		60
Multiple input	80		30
Motion control attacks			
Fixed time		80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.5: Standard Controllers Analysis - Cognitive Systems

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90			50		26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50	50							
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90	80							
Attack Chaining	90	80							

Table 4.6: Standard Controllers Analysis - Fine Motor Abilities (Hands). Abbreviations: we use . to mean ‘ing’, so Press. means Pressing, etc.

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.7: Standard Controllers Analysis - Fine Motor Abilities (Head and Feet)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.8: Standard Controllers Analysis - Gross Motor Abilities

The greyed-out columns represent motor abilities that are not captured through interacting with this controller; any non-grey columns are abilities that the player may naturally use when using this controller type. For standard controllers, arm movements are still available because players can use arm movements to supplement other fine motor abilities, like wrist and forearm movements.

The dotted rows represent challenges that we could not find examples for on this controller. We see that there are 6 unexplored challenges: the rapid tapping challenges (indiscriminate, and alternating), the scribbling challenges (rapid line drawing, and rapid shape drawing), rapid controller rotation, and motion control attacks. For the rapid tapping and scribbling challenges this makes sense as the controller aren't equipped for those challenges both families of challenges require touchscreens which the standard controller does not have. It makes less sense for the latter two challenges as modern standard controllers come equipped with gyroscopes and accelerometers to measure these types of motions. For rapid controller rotation, it is possible this challenge does not appear because of how the controller is intended to be held; a player holding a controller between their two hands would have a hard time quickly trying to spin or rotate it because of the restrictions on their wrist movements. Attempting to coordinate the rotating wrist movements while holding a rigid object between them creates an unnatural, and awkward movement which in turn makes it hard to perform quickly. However, we know that handheld consoles, which are held the same way, have an example of this challenge as we discussed it in the analysis (see section 1.9). Therefore, it's possible for this challenge to be done, but designers have not explored it. Motion control attacks likely don't exist because this controller already has several buttons. Players come with pre-existing knowledge about using standard controllers; a large part of this knowledge is that buttons are the main form of interaction. The controller also communicates this with the number of buttons, the space they take up on the controller face, and the buttons affordances themselves. Therefore, they are unlikely to adopt new methods of interaction on a device that is extremely familiar and does not imply that it has motion sensing capabilities. Another potential reason why we don't find motion control attacks with this device is due to the "death" of motion controls. Many "hardcore" gamers find motion controls to be gimmicks and associate them with "casual gaming" — therefore there is little incentive for developers to pursue motion control-based challenges [267, 81, 261, 260, 245, 257]. Associated to this may be the issue of fatigue; standard controllers and the consoles that they work with imply a longer play session length, and over the course of a longer play session movements of any kind will begin to fatigue the player.

We see an obvious bias towards using fine motor abilities, with a focus on finger movements. This would make sense since buttons are the most obvious interaction method on this controller. We speculate that this bias may persist through the other controllers due to widespread availability of standard controllers. It is possible that designers consider it to be the "default" controller, and so when creating a generic gaming experience will gear it towards standard controller play experiences. If this is the case, it is possible that the underlying standard controller-centric experience is carried through when porting games or working on cross-platform experiences. More research would need to be done in this area to understand whether this hypothesis has any truth to it.

Though there is a fine motor bias, we do find 2 challenges that engage gross motor abilities: rapid analog stick rotation, and rapid controller shaking. The gross motor abilities used are both arm movements and are important in the completion of the challenges (level 3 and 4). These two challenges are the only ones that don't focus on button pressing as input. The use of gross motor abilities for both challenges seems to come from speed context. In the rapid analog stick rotation challenge (see Section 1.4) when using your thumb or fingers to spin the analog stick in a short time limit, the player can't reach their maximum potential because of limitations in how the thumb moves, therefore they switch the hold on their controller to maximize their score by moving their shoulders and arms instead. We find similar movement with shaking, where it can be accomplished using fine motor wrist movements, however players tend to use more gross motor abilities to optimize play. More study should be done into these outliers, as it's possible a more thorough study of this challenge in different context (e.g. multiplayer vs. single player) will show different results.

4.2 Handheld Motion Controllers

Isolating the handheld motion controller data, we end up with the following:

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.9: Handheld Motion Controller Analysis - Cognitive Systems

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	**Press.**	**Swiping**	**Pinch.**	**Shaking**	**Flick.**	**Point.**	**Swing.**	**Drawing**	**Tilt.**
Single button input	90					26			
Multiple button input	90					30			
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing					81				
Rapid shape drawing						50			
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40								
Multiple input	50	50							
Motion control attacks				40					
Fixed time	30			30					
Variable time	35	30							
Simple Reaction Time				25					
Learning complex combos	90	80							
Attack Chaining	90	80							

Table 4.10: Handheld Motion Controller Analysis - Fine Motor Abilities (Hands). Same abbrev. as before.

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.11: Handheld Motion Controller Analysis - Fine Motor Abilities (Head and Feet)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.12: Handheld Motion Controller Analysis - Gross Motor Abilities

Though our intent was to remove multi-column cells, we still have Motion Control Attacks covering 6 fine motor spaces and 5 gross motor spaces, Fixed Time Movement spanning 2 fine motor abilities, and Simple Reaction Time spanning 4 fine motor spaces. For this thesis it makes sense because it conveys that there are several ways a player can move themselves to execute a Motion Control Attack. This is likely to be the case with all motion control implementations of challenges. To break these cells down further, we would need multiple copies of each challenge with their own specific motor configuration. This would significantly increase the number of rows on our table without adding a comparable amount of new information. Though we are choosing to not follow this thread further, we believe that it should be undertaken as future work; further refining these tables would allow us to produce recommendations to designers about novel experiences and may lead to the creation of new challenges.

The greyed-out columns for this controller are the finger pinching, head, foot, legs, and torso abilities. This makes sense since the controller cannot track these types of motions. Handheld motion controllers can track arm movements as it measures the changes in position and distance from the receiver. However, it only measures the position of one arm, meaning that movements with the other one are not tracked. Rhythmic dance games, such as Just Dance [234], encourage the use of leg abilities but do not directly measure these movements (indirect measurements may be made through pattern matching of accelerometer/gyroscope readings). In the case of Just Dance for the Wii [234], only the Wii Remote position is tracked, therefore only effectively tracking the movement of one of the player's arms. As such, some players choose instead to just use their dominant arm to interact with these games, bypassing the intended play method. This opens a larger issue with motion controls, which is intended play patterns versus actual play patterns. If we were to look at the intended play patterns with these controllers and games, most of these grey columns would be available. With more precise full body tracking, it may be possible to streamline the experience to better fit the intended play. This merits a discussion about player fatigue, as fitting the intended play style would increase player fatigue since it incorporates more movement. It is possible that these unintended play styles were created to mitigate the fatigue players were feeling when playing these games "correctly".

We see there are both fine and gross motor abilities used in this controller type. However, there still seems to be a bias towards fine motor abilities, particularly finger and wrist movements. This is possibly due to several factors, including controller orientation, fatigue, and the state of the market.

Focusing on fine motor skills would allow for similar challenges in both orientations, as movements like tilting, thrusting, and twisting, can be accomplished by the wrist as opposed to using larger arm muscles. Fine motor abilities may also cause less fatigue than gross motor abilities over the same play session. Finally, it is possible that there is little development around the gross motor abilities due to feedback from “hardcore gamers” who view motion controls as too “casual”, “gimmicky”, frustrating, or just generally dislikable [81, 261, 260, 245, 257]. Since the consumer profile of a “hardcore” gamer is significantly different from the average gamer and from “casual” gamers, it is possible that their feedback skewed the market away from experimenting with more novel and gross motor ways to use this controller. It is likely that with the popularity of the Nintendo Switch this perception is changing, as new games like Fortnite, Splatoon 2, and Breath of the Wild feature motion control support. Future researchers should continue to explore this area to see how these motion controls can be further developed and integrated into other games.

This controller can be held in two orientations, vertically in one hand and horizontally between two hands; Tables 4.9, 4.10, 4.11, 4.12 are an attempt to capture both of those orientations. However, in the same way that trying to capture all the different controllers in one table obscures details we want to see, trying to understand a multi-faceted controller through an amalgamated table also would obscure information. We further separated this table into its vertical and horizontal orientations. It is important to note that most handheld motion controllers have peripheral attachments, such as the Wii Nunchuck, that add functionality and change the experience of these controllers. We omit discussing the experiences of these peripherals in this discussion as it is out of scope for our current work. Future work may want to investigate this.

4.2.1 Vertical Handheld Motion Controller

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.			
Attack Chaining			

Table 4.13: Handheld Motion Controller Analysis - Cognitive Systems

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing					81				
Rapid shape drawing						50			
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40								
Multiple input	50	50							
Motion control attacks				40					
Fixed time	30			30					
Variable time	35	30							
Simple reaction time				25					
Learning complex combos.									
Attack Chaining									

Table 4.14: Handheld Motion Controller Analysis - Fine Motor Abilities (Hands). Same abbr. as before.

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.15: Handheld Motion Controller Analysis - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.16: Handheld Motion Controller Analysis - Gross Motor Abilities

In its vertical orientation we see that this controller doesn't support the following challenges: rapid analog stick rotation, indiscriminate tapping, alternating tapping, learning combination moves, and attack chaining. For rapid analog stick rotation and the rapid tapping challenges (indiscriminate, and alternating) this makes sense as they require features that the controller doesn't have (analog stick, and a touchscreen). We hypothesize that generally, all Rapid Analog Stick Rotation challenges can be adapted into Rapid Controller Rotation challenges, thus implying that even though Handheld Motion Controllers don't have analog sticks, they can deliver a similar experience by having the player twirl their controller. This would require further investigation to see whether it is true, and to explore whether the player considers the challenges to be similar experiences.

Rapid line drawing is an outlier challenge where we were unable to find many examples of it in commercial games, the only one being the "At the Chomp Wash" minigame in Mario Party 8 [97]. This is odd as the controller has the capability of registering movements for drawing, as can be seen in the Rapid Shape Drawing challenges. This challenge's low appearance rate may be because developers thought rapidly swinging your arm to "scribble" in the air would be too tiring. The Mario Party 8 example works because it is a minigame and so exists in a limited time setting. Alternatively, it may be because they did not find it to be interesting; rapid line drawing would effectively just be swinging your arm back and forth very quickly, which could be hard to develop a "fun" context for. Examples of arm swinging with handheld motion controllers often come in contexts where speed is a secondary factor, such as in tennis or ping-pong styled games.

Advanced Combat challenges do not exist in this orientation. We speculate that this is due to intended play style of this orientation; when held vertically like a wand, motion controls become intuitive and tend to be prioritized. In an Advanced Combat context, there is not enough control to allow for complex combination moves and attack chains to be created. Super Smash Bros. Brawl [87] for the Wii offers a vertical orientation control scheme, but it is only available if the Wii Nunchuck is also attached. Since peripherals are out of our scope, this type of vertical control scheme is not captured by our table.

This orientation uses gross motor abilities for rapid shape drawing, rapid controller shaking, rapid controller rotation, motion control attacks, and simple reaction time challenges. In this context, rapid shape drawing tends to ask the player to hold the controller like a wand and draw something in the air. The motions used for drawing like that would require larger arm and shoulder movements, which would be more fatiguing to a player over a long play session. Future work should investigate the differences between rapid shape drawing for handheld motion controllers versus the same challenge implemented on a smartphone/tablet or handheld console context, as they vary along lines of fatigue (fine vs. gross motor abilities) and haptic feedback (drawing on a screen vs. drawing in the air). Understanding how these contexts create different play experiences would allow us to better create novel experiences for each controller.

Rapid controller shaking and rotation both require a combination of fine and gross motor abilities to work. Rapid controller shaking can be exclusively done through larger arm swinging movements or through a combination of arm and wrist movements. Rapid controller rotation requires the player to twirl the controller in their hand, which is done through the coordination of wrist and arm movements. Future work should explore the nature of this coordination; currently, we believe that the fine motor elements are more taxed than the gross, however, this comes from our own limited understanding of how these challenges are played. It is possible that the "average" player exclusively focuses on gross motor abilities.

Motion control attacks can use one to any number of fine or gross motor abilities. Motion control attacks, such as those in The Legend of Zelda: Skyward Sword [144] can be incredibly fatiguing over long play sessions when they focus on gross motor abilities like arm swinging and arm pushing (thrusting) to emulate sword combat. It is likely, since only the controller is being tracked, that fatigued players will move from using gross motor abilities to using fine motor abilities to complete the task. Future work should look into understanding and modeling how fatigue affects play style and by association challenge design. As well, many motion control attacks rely on there being a peripheral attached to the handheld motion controller. In The Legend of Zelda: Skyward Sword, Shield Bashing (where the player uses their shield item to push back an attacker) is exclusively done through shaking the Wii Nunchuck peripheral. Blocking and Dodging in the Wii Sports Boxing minigame [143], extends this by requiring coordinated movement of the peripheral and handheld motion controller for these moves to be executed. Future work should consider the differences in fatigue, experience, and the required levels of motor abilities between Motion Control Attacks that only require the main handheld motion controller, and those that require both the controller and peripherals.

Simple Reaction Time challenges can use one of several fine or gross motor abilities. Often, the challenge will require the player to respond with only a single motor ability. For example, to shake off Raw Shocks in Silent Hill: Shattered Memories, the player makes shoving motions (arm pushing) with the Wii Remote and Nunchuck [48]. The issue of fatigue is lessened for this challenge since, by definition, it is a short segment apart from normal gameplay. This would mean that even in longer play sessions, the player would have breaks from this challenge where they could recover from any fatigue. It

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input			
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks	80	90	50
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.17: Differences between Standard Controller and Vertical Handheld Motion Controller - Cognitive

may be interesting for future researchers to study the different types of Simple Reaction Time challenges that show up for this controller, and to analyse whether there is a focus on fine or gross motor abilities, and how having a peripheral attachment affects this challenge type.

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90			50		26			
Multiple button input	90			50		30			
Alternating button input									
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing					81				
Rapid shape drawing						50			
Rapid controller shaking									
Rapid controller rotation								85	
Single input									
Multiple input									
Motion control attacks				40					
Fixed time				30					
Variable time									
Simple Reaction Time	25			25					
Learning complex combos	90	80							
Attack Chaining	90	80							

Table 4.18: Differences between Standard Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Hands). Same abbrv. as before.

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.19: Differences between Standard Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.20: Differences between Standard Controller and Vertical Handheld Motion Controller - Gross Motor Abilities

When we compare this orientation to the colour-coded visualization of the Standard Controller data, we can compare how these controllers are experienced. We therefore created another table which compares the two sets of data. We interpret this comparison table as green cells indicating that the tables are identical at that point, light blue cells meaning that data was found in the handheld motion controller table and was not in the standard controller table, navy blue cells meaning we found it in the standard controller table and not the handheld motion controller, and yellow meaning that there was some other difference in the table. The yellow on this table is also filled with a grey dotted pattern, representing that those yellow cells did not exist in one of the source tables. We keep grey to separate the unmonitored abilities from the others.

From this comparison table we see that the vertically oriented handheld motion controller shares many challenges with the standard controller but varies slightly on which motor abilities are used. We see from this table that the Standard Controller can support Single Input Button Mashing, Multiple Input Button Mashing, Rapid Analog Stick Rotation, Learning Complex Combinations, and Attack Chaining while the Handheld Motion Controller can support Rapid Line Drawing, Rapid Shape Drawing, Rapid Controller Rotation, and Motion Control Attacks. Some of these make sense based on controller features (a handheld motion controller doesn't have a thumbstick, so it can't support a thumbstick based challenge). There are some instances where this doesn't make sense; for example, Standard Controllers come equipped with motion sensors and can track that information, but do not support Motion Control Attack challenges. This is more a case of missed design opportunity, where it is possible to create instances of that challenge on a Standard Controller but we are not aware of anyone who has attempted to do so. Similarly, Handheld Motion Controllers have several buttons, yet we did not find examples of button mashing in the vertical orientation. We speculate that there are two reasons for this; firstly, it is uninteresting design for this controller type. With the multitude of motion capabilities this controller offers, it would seem banal to resort to button mashing when that action can be replaced with something like shaking the controller. In comparing Mario Party 8 and 9 [97, 135], the only Wii-based installments of the Mario Party series, against their predecessors, we found that the Wii editions had no button mashing mini-games in comparison to the previous minimum of 2 [1]. From examining the mini-games in Mario Party 8 and 9, it seems that what previously may have been styled as button mashing challenges were replaced by more gross motor focused challenges like shaking, thrusting, or tilting the controller. Secondly, we speculate this is because of the way the controller is meant to be held; using the Wii

[1] Mario Party through to Mario Party 7 had 3, 5, 4, 2, 4, 3, and 2 Single Input Button Mashing challenges respectively.

Remote as an example, the main buttons on the controller are situated directly under the thumb and the index finger. This set up, while making it easy and comfortable to reach the buttons, creates an issue when attempting to mash them as the player's range of motion for their thumb is limited, making awkward to competitively button mash. This was the same problem we documented with the Standard Controller when player's attempt to button mash while holding it in its intended orientation (see Section 1.1,1.2,1.3).

These hypotheses would need to be further studied by future researchers to determine whether it has any merit. In comparing the challenges that were mostly identical (mostly green), we notice that handheld motion controllers use more wrist and arm abilities than standard controllers. We speculate that this is due to the controller design and orientation; it is hard to incorporate coordinated wrist movements when the controller is held between two hands. In contrast, the handheld motion controller designed to be used like a remote in one hand creates opportunities from wrist movement. We see in cases like Fixed Time Movement that the challenge is practically identical to Standard Controllers, only adding potential wrist abilities.

4.2.2 Horizontal Handheld Motion Controller

Challenges	**Cognitive Systems**		
	Perception	**Attention**	**Memory**
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks			
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.21: Analysis Data for Horizontal Handheld Motion Controller Cognitive Systems

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90					30			
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50	50							
Motion control attacks									
Fixed time	30			30					
Variable time	35	30							
Simple Reaction Time				25					
Learning complex combos	90								
Attack Chaining	90								

Table 4.22: Analysis Data for Horizontal Handheld Motion Controller Fine Motor Abilities (Hands). Abbrv. as before.

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.23: Analysis Data for Horizontal Handheld Motion Controller Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.24: Analysis Data for Horizontal Handheld Motion Controller Gross Motor Abilities

In its horizontal orientation we see that this controller doesn't support the following challenges: rapid analog stick rotation, indiscriminate tapping, alternating tapping, rapid line drawing, rapid shape drawing, rapid controller rotation, and motion control attacks. This is one more challenge than its vertical orientation. By the same rationale as stated in the vertical orientation, this makes sense for rapid analog stick rotation and the rapid tapping challenges (indiscriminate, alternating). This controller loses its motion control abilities due to the orientation making it awkward for players to move the controller in a meaningful way. We therefore lose the ability to draw or do certain motion control attacks. While the orientation limits the more apparent movements, the controller still supports actions like tilting and shaking, thus allowing it to venture into gross motor territory. We hypothesize that handheld motion controllers in this orientation will be extremely similar to standard controllers. There are two reasons we think this: orientation match, which means the handheld motion controller will be subject to similar constraints as the Standard Controller, and the multitude of gameplay examples which allow you to swap out the motion controller for a standard controller. We therefore created another visualization to compare the horizontal motion controller context to the standard controller:

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.25: Differences between Standard Controller and Horizontal Handheld Motion Controller - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input				50					
Multiple button input				50					
Alternating button input									
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking									
Rapid controller rotation									
Single input									
Multiple input									
Motion control attacks									
Fixed time				30					
Variable time									
Simple Reaction Time	25			25					
Learning complex combos.		80							
Attack Chaining		80							

Table 4.26: Differences between Standard Controller and Horizontal Handheld Motion Controller - Fine Motor Abilities (Hands). Abbrv. as before.

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.27: Differences between Standard Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.28: Differences between Standard Controller and Vertical Handheld Motion Controller - Gross Motor Abilities

We interpret this table the same way we interpreted Tables **??**, 4.18, 4.19, 4.20. From this we can see that these controller contexts are almost identical. For the button mashing challenges (single and multiple), Handheld Motion Controllers lose the wrist shaking ability that characterized the Standard Controller. That ability was originally tied to the Standard Controller because of how players were noticed to strategically hold the controller in one hand and shake their arm to get optimal button pushes. For handheld motion controllers, we did not find competitive button mashing segments of games. Examples for button mashing in Handheld Motion Controllers came from single player games, where a low button mashing score did not trigger a lose condition. Therefore, it did not need to be as aggressive as when players are competing against each other. It's possible that competitive button mashing is not as abundant in handheld motion controller contexts, making players generally less invested in playing optimally and so leading to them dropping the wrist shaking. Other instances of differences like Fixed Time Movement show that the challenge requires the same abilities as the Standard Controller and additional wrist movements. We take this table to mean that horizontally oriented handheld motion controllers are extremely similar in challenge-motor experience to Standard Controllers. Future work should further investigate this, as it would be interesting to see whether all horizontally oriented controllers use the same abilities as the Standard Controller.

4.3 Full Body Motion Controllers

Isolating the full body motion controller data, we get the following table:

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks	80	90	50
Fixed time	90	80	
Variable time			
Simple Reaction Time	90	75	
Learning complex combos.			
Attack Chaining			

Table 4.29: Analysis of Full Body Motion Controller - Cognitive Systems

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking									
Rapid controller rotation									
Single input									
Multiple input									
Motion control attacks									
Fixed time									
Variable time									
Simple Reaction Time									
Learning complex combos.									
Attack Chaining									

Table 4.30: Analysis of Full Body Motion Controller - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.31: Analysis of Full Body Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25							
Learning complex combos.								
Attack Chaining								

Table 4.32: Analysis of Full Body Motion Controller - Fine Motor Abilities (Head and Foot)

We see that this controller type cannot interpret any fine motor abilities. This is due to our basis for understanding full body motion controllers: the Kinect. The Kinect was not able to detect fine motor abilities due to its technical limitations, and as it was the only major consumer-focused full body motion controller. We therefore assumed that no full body motion controller could support these movements and continued to base our understanding exclusively on the Kinect. The Kinect was released in 2010, and discontinued in 2017 [269]; it is possible that future full body motion controllers will be able to distinguish between minute movements enough to accommodate fine motor abilities.

This controller only supports 3 of our challenges: motion control attacks, fixed time movement, and simple reaction time. As this controller type lacks obvious hard controls (i.e. buttons, analog sticks, touchscreens) it automatically distances itself from challenges that focus on these controls.

Advanced Combat challenges do not exist on this controller; however, "Fighting games" (where these challenges comprise the entirety of the gameplay) do, specifically Dragon Ball Z for Kinect [199] . Though this game is marketed as a "Fighting game", the combat is significantly different from the other examples of the genre as it is reactionary rather than combo and strategy focused. While Dragon Ball Z for Kinect has "combos" they amount to the game telling you to rapidly punch the other character, with no variation in controls or player choice in the input. Attack chaining similarly doesn't exist; the combat in the game is segments of punching at the screen interspersed with cinematics of the characters fighting when enough damage has been incurred. In terms of our definitions of these challenges (Section 3), and the player abilities associated with them, the combat in Dragon Ball Z for Kinect requires little to no cognitive abilities, and so is more of a motion control attack challenge than an Advanced Combat challenge.

For the challenges that this controller does cover, we see that it emphasizes gross motor movements. Comparing the cell values for Fixed Time Movement and Simple Reaction Time between Standard Controllers, Handheld Motion Controllers, and Full Body Motion Controllers, we see that they are cognitively identical. Though the motor component is different (fine vs. gross), we see that in all instances the motor component is in the same range (noticeably used-level 2, 26-50) implying that they are similar at a high level. This begs the question whether the switch to gross motor abilities makes the challenge "feel" sufficiently different from the fine motor version. If it does, that perhaps merits further investigating whether the gross motor versions are actually different challenges that we were unable to identify at this point. As well, further investigation would be needed to confirm the Full Body Motion Controller numbers; since gross motor abilities are more involved and fatiguing than fine motor abilities, it is possible that the gross motor numbers should actually be higher to reflect a greater importance in completing the challenge. For example, in the endless runner style Kinect Adventures! minigame "River Rush" players step left or right to change which path they're on, and must jump to leap over obstacles [78]. These movements are more involved, and therefore more important to completing the challenge than an endless runner game like Runner 2 which only requires the player to press a single button [72]. Confirming whether gross motor ability numbers need to be higher would become important as we consider different age groups; using the River Rush example, very young (toddler) and elderly individuals would struggle with jumping in the game as opposed to pushing a button which both groups would be able to do with little problem. The same reasoning applies to Simple Reaction Time challenges.

It is interesting to note that this controller doesn't support Variable Time Movement challenges, when it does support Fixed Time Movement challenges. From going over the analyses and examples, if a controller supports one of these challenges it tends to support the other. We believe this is because of the challenge similarities Variable Time Movement challenges are Fixed Time Movement challenges with an additional cognitive component because you control the avatar movement. It is possible that this controller doesn't support this challenge because it requires multi-dimensional input (i.e. simultaneously pressing a direction on the analog stick and a button), which the Kinect (the only commercial full body motion controller) cannot do (see Section 2.3).

The limited number of available challenges could be due to three things: an unconscious bias in challenge construction, our own limited knowledge of games, or the short lifespan of the Kinect. Firstly, our challenges exist with certain implicit biases (e.g. Rapid Controller Rotation assumes a physical controller that can be rotated) which automatically precludes novel controllers like the Kinect (controller is your body, not another object). If we were to expand our challenge definitions, it is possible that more would appear on this and other controllers. Secondly, we are limited by the games we are familiar with; if we can't find an example of a challenge, we are apt to think it doesn't exist in this context. Finally, it could be the result the Kinect's relatively short lifespan as a controller. As previously discussed (see Section 2) we decided to not explore non-standard controllers (e.g. WiiFit Board, DDR mats, etc). It is possible that because of the negative response to the Kinect and its short lifespan [267] we unconsciously lumped it with these non-standard controllers. The reason that we never explicitly scoped the Kinect out of our work was its ability to showcase gross motor abilities, which we still see through the 3 potential challenges it can accommodate. It is important that future work further research and expand our understanding of full body motion controllers from other vantage points.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.33: Differences between Full Body Motion Controller and Standard Controller

4.3.1 Differences with Standard Controller

The following table compares Full Body Motion Controllers and Standard Controllers. We interpret it as follows: navy blue means that exists on Standard Controllers and not on Full Body Motion Controllers, light blue means that exists on Full Body Motion Controllers and not Standard Controllers, grey means that those abilities are not recognized on either controller, green cells mean that the two controllers are identical at that point, green cells with a dotted pattern indicate that challenge does not exist on either controller, yellow means that there is a difference between the two, yellow with dots means that challenge does not exist on one of the controllers (the opposite controller to the colour in the rest of the row). Generally, this means that only rows with no dots are shared between the controllers.

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90			50		26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50	50							
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90	80							
Attack Chaining	90	80							

Table 4.34: Differences between Full Body Motion Controller and Standard Controller - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.35: Differences between Full Body Motion Controller and Standard Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25							
Learning complex combos.								
Attack Chaining								

Table 4.36: Differences between Full Body Motion Controller and Standard Controller - Gross Motor Abilities

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Fixed time			
Simple Reaction Time			

Table 4.37: Differences between Full Body Motion Controller and Standard Controller

We see that they only share 2 challenges: Fixed Time Movement and Simple Reaction Time. For easier reading, we condense the differences table to just those challenges.

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Fixed time	30								
Simple Reaction Time	25								

Table 4.38: Differences between Full Body Motion Controller and Standard Controller - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Fixed time				
Simple Reaction Time				

Table 4.39: Differences between Full Body Motion Controller and Standard Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Fixed time	30							
Simple Reaction Time	25							

Table 4.40: Differences between Full Body Motion Controller and Standard Controller - Gross Motor Abilities

We see from this that the challenges are identical across the cognitive abilities. The only differences seem to be that the Standard Controller versions focus on fine motor abilities, while the Full Body Motion Controller focuses on gross motor abilities. We can see that the importance of each motor ability is identical between the controllers (for each challenge the number in the motor ability cells are the same). This seems to support our idea that Full Body Motion Controller challenges are essentially the same with a focus switch to gross motor abilities.

4.3.2 Differences with Handheld Motion Controllers

The following table compares vertically oriented handheld motion controllers to full body motion controllers. It is meant to be interpreted in the same way as Table 4.37, 4.38, 4.39, 4.40.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks			
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.41: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing					81				
Rapid shape drawing						50			
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40								
Multiple input	50	50							
Motion control attacks				40					
Fixed time	30			30					
Variable time	35	30							
Simple Reaction Time				25					
Learning complex combos.									
Attack Chaining									

Table 4.42: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.43: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks						40		
Fixed time	30							
Variable time								
Simple Reaction Time	25		25					
Learning complex combos.								
Attack Chaining								

Table 4.44: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Gross Motor Abilities

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Motion control attacks			
Fixed time			
Simple Reaction Time			

Table 4.45: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Cognitive

To better read this table, we reduce it to just the rows of shared challenges:

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Motion control attacks				40					
Fixed time	30			30					
Simple Reaction Time				25					

Table 4.46: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Motion control attacks				
Fixed time				
Simple Reaction Time				

Table 4.47: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Motion control attacks						40		
Fixed time	30							
Simple Reaction Time	25		25					

Table 4.48: Differences between Full Body Motion Controller and Vertical Handheld Motion Controller - Gross Motor Abilities

We see from this table that these controllers share 3 challenges: motion control attacks, fixed time movement, and simple reaction time. For these challenges we see that the cognitive abilities overlap for both controllers. The major differences we see is that handheld motion controllers emphasize wrist movements versus the arm movements for full body motion controllers. It is interesting to note that for Motion Control Attacks the two controllers overlap for all of the arm motions, meaning both controllers use them to the same amount. This is potentially because of the limited number of challenges; it is possible that this subset just focuses on gross motor abilities more than the rest. It would be interesting to investigate whether they overlap as much if Full Body Motion Controllers expand into other challenges. Generally, it would be interesting to compare how similar the experiences these 3 challenges are on these different controllers considering they have the largest overlap in abilities.

The following table compares the horizontally oriented handheld motion controller to the full body motion controller. As we previously theorized that this orientation of handheld motion controller closely matches the standard controller, we would expect the comparison of these two to be similar to the comparison with the standard controller.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.49: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90					30			
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50	50							
Motion control attacks									
Fixed time	30			30					
Variable time	35	30							
Simple Reaction Time				25					
Learning complex combos.	90								
Attack Chaining	90								

Table 4.50: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Fine Motor Abilities (Hands)

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.51: Differences between Full Body Motion Controller and Horizontal Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25		25					
Learning complex combos.								
Attack Chaining								

Table 4.52: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Gross Motor Abilities

Challenges	**Cognitive Systems**		
	Perception	**Attention**	**Memory**
Fixed time			
Simple Reaction Time			

Table 4.53: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Cognitive

To make this easier to read we reduce it to the shared challenges:

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Fixed time	30			30					
Simple Reaction Time				25					

Table 4.54: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Fixed time				
Simple Reaction Time				

Table 4.55: Differences between Full Body Motion Controller and Horizontal Handheld Motion Controller - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Fixed time	30							
Simple Reaction Time	25		25					

Table 4.56: Differences between Horizontal Handheld Motion Controllers and Full Body Motion Controllers - Gross Motor Abilities

We notice that these controllers only share 2 challenges: fixed time movement, and simple reaction time. This is identical to the standard controller differences. We notice that the cognitive abilities are identical for these challenges on the different controllers. It is important to notice that for Simple Reaction Time the horizontal handheld motion control and full body motion controller overlap in arm swinging.

4.4 Smartphones/Tablets

Isolating the Smartphone/Tablet data we get the following:

Challenges	**Cognitive Systems**		
	Perception	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks			
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.			
Attack Chaining	70	60	20

Table 4.57: Smartphone/Tablets Analysis Data - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40					40			40
Multiple input	50								
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.									
Attack Chaining	90	80							

Table 4.58: Smartphone/Tablets Analysis Data - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.59: Smartphone/Tablets Analysis Data - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.60: Smartphone/Tablets Analysis Data - Gross Motor Abilities

Before moving forward, we reiterate that Smartphones/Tablets can be held in two orientations: portrait or landscape mode. Though we separated those modes out for handheld motion controllers, we left them combined here as there were no significant differences in the availability of challenges for each orientation.

We see that tapping challenges are currently exclusive to this controller. This is because this controller features a touchscreen, with many models use as the main method of interaction with the device. This may skew how designers create games for this controller, as the default interaction mechanic would be tapping the screen. This is somewhat indicated by the sheer number of tapping games on mobile devices; in the Google Play Store alone, when searching for the term "tapping" 250 games pop up — this is not counting the tapping games that do not have the word in their name or tags, and is therefore a lowball estimate. On top of being the default interaction, we believe that tapping challenges are so abundant because of their simplicity — in our original explanation (see Section 1.3) we mentioned tapping challenges to be the button mashing of touchscreen controls, which makes it one of the most accessible controls for a wide demographic. We speculate that handheld consoles will also see a significant amount of tapping challenges because of their touchscreen capabilities.

We see that this controller type does not support any button mashing or analog rotation challenges. This is because these controls do not exist on this controller. The initial reaction to this may be to discredit this, since this controller can support soft controls such as on-screen buttons, therefore allowing for "button mashing". We have two rebuttals. Firstly, our original explanation of "button mashing" challenges discussed how specifying where on a screen a player has to tap incorporates accuracy into the challenge due to factors like the size and placement of the soft control and the player has to be more considerate of whether the input went through since there is no haptic feedback, whereas the haptic feedback of the physical button which informs you where pressing is possible and whether pressing has occurred (see Section 1.3). This would create a difference on the analysis table as they would use different abilities and was the original justification for creating the tapping challenges. Secondly, as we previously discussed (see Section 3.1.1), the lack of physical controls creates a measurable difference in motor ability and experience. Since what we are concerned with are these measurable differences and how they impact challenges, this distinction between hard and soft controls becomes important and makes button mashing by our challenge definition (see Section 1.1) impossible on this controller.

This controller does not support motion control attacks. This is surprising to us as this controller type comes equipped with various sensors that can detect motion (e.g. gyroscope, accelerometer). This would imply that we could create instances of this challenge. We speculate that we don't find any of these challenges because the screen and controller are the same device for this controller type. To measure movements for motion control attacks, the player would lose sight of the screen which may hinder gameplay as they would lose visual feedback from the game while playing. We see in the case of handheld consoles like the Switch that motion control attacks in games like the Legend of Zelda: Breath of the Wild [152] require the player to switch to using handheld motion controllers and relegate the main console (Switch) to exclusively being a screen. It may be possible to create cross platform experiences with a TV screen displaying the game information, and the player using their smartphone/tablet similarly to how a handheld motion controller is used.

This controller supports attack chaining challenges, but not learning complex combination moves. We believe that more investigation is required to determine whether this was accurate. To study both Advanced Combat challenges in this context, we explored "fighting games" that exist on this controller. We found that there are only a handful of classical "fighting games" on this controller. This is surprising since we see that handheld consoles like the Nintendo 3DS and Playstation Vita have multiple classical fighting games such as Super Smash Bros. for 3DS, Dissidia: Final Fantasy, and the Dragon Ball Z: Tenkaichi Budokai series. We used Injustice 2 for Android [265] as a representative example for this genre. This allowed us to compare the gameplay on this smartphone version to the console version with the standard controller. In this game all the combat moves were either tapping an on-screen button (1 button for punching, 1 button for kicking), or swiping on the screen to jump or move. Therefore, as the player could string together a series of kicks, jumps, and punches this would constitute as supporting an attack chaining challenge. However, this control scheme limits the types of combat moves that we can make by removing the light/medium/heavy attack dynamic and other strategic moves, such as throws and blocking, which are elements of classical fighting games. It makes sense that these are removed to simplify the controls for a smaller screen size, especially where using the controls would obscure part of the screen during gameplay, but this creates an uninteresting set of possible attack chains and makes the gameplay feel shallow. Since the game does not offer a wide variety of combat moves and does not support multi-dimensional input (i.e. moving while attacking) the game cannot support combination moves that often require multi-dimesional or multiple simultaneous inputs to be executed. Therefore, since the game can't support combos is can't support the Learning Complex Combination Moves challenge. From spending time with this game and attempting to understand what challenges it has and why, it leads us to speculate that attack chaining and learning combination moves are codependent challenges in so far as they rely on each other to create well rounded and engaging gameplay. More work should be done examining the relationship between these two challenges. If we could determine that some challenges are symbiotic, it would create a multitude of questions about challenge groupings based around codependent pairs and would define a space of experiments to study new pairings.

We see that foot, legs, and torso movements are greyed out in this table as they are not measured by this controller. Head movements are left in because mobile devices come equipped with front facing cameras and microphones which can parse these movements. We see examples of these in instances like Snapchat's Snappables games [196] that use head and face tracking as game mechanics, and Chicken Scream [165] that uses talking/screaming as the main mechanic. Arm movements aren't directly measured; however, they can be used for larger motions like shaking the controller, rotating the controller, etc. These are possible to be measured because of the motion sensors (gyroscope, accelerometer) built into the controller. Therefore, we believed it was important to leave them available on the table.

There still seems to be a heavy emphasis on fine motor abilities, specifically wrist and finger movements. Wrists are more prominently featured in the drawing and tapping challenges. Generally, this may be because of the option to use your finger or a stylus to interact with the touchscreen. As we previously discussed (see Section 2.4), when using a finger to interact with a touchscreen the movement occurs at the first knuckle in a swiping motion; in comparison, using a stylus (or a finger held stiffly to emulate a stylus) movements come from the wrist because of how the stylus is held (in a pincer grip like a pencil). This may also be because of the challenge capabilities of the controller. Since this is one of the only controller types that allow for tapping challenges and scribbling challenges with haptic feedback, it is possible that these challenges (and the fine motor abilities used to complete them) are overrepresented on this controller, and therefore on this table.

4.4.1 Differences with Standard Controller

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining			

Table 4.61: Differences between Standard Controller and Smartphones/Tablets - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90			50		26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input		40				40			40
Multiple input		50							
Motion control attacks									
Fixed time		30							
Variable time									
Simple Reaction Time			25						
Learning complex combos.	90	80							
Attack Chaining									

Table 4.62: Differences between Standard Controller and Smartphones/Tablets - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.63: Differences between Standard Controller and Smartphones/Tablets - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.64: Differences between Standard Controller and Smartphones/Tablets - Gross Motor Abilities

We notice that there is an equal challenge trade-off between Smartphones/Tablets and Standard Controllers, with each having 5 challenges that the other does not. Smartphones/Tablets have tapping challenges and controller rotation, whereas Standard Controllers have button mashing, rapid analog stick rotation, and learning complex combination moves.

These controllers share 4 challenges with minor differences. These are: Single Input Attacks, Multiple Input Attacks, Fixed Time Movement, and Simple Reaction Time. We see in these challenges that the differences are mostly the smartphones/tablets adding more motor abilities to the mix (predominantly wrist abilities). This is possibly because the designers of mobile challenge instances understand the different capabilities of the controller and are focused on exploring them, or at least showcasing them.

We see that rapid controller shaking, variable time movement, and attack chaining are identical (indicated by the green cells all the way through). A potential reason for this could be that Smartphones/Tablets can be held in a horizontal orientation, allowing for players to hold it the same way they would a Standard Controller. Therefore, it is possible that this orientation congruity creates the need for identical motions to be used. Further research should examine the on-going relationship between controller orientation, challenge availability, and the motions that can be used to complete the challenge.

4.4.2 Differences with Handheld Motion Controller

We speculate that focusing on the differences between these two controllers will produce interesting information, as they are the only controllers where the orientation can be changed. We believe that in comparing these controllers for the horizontal orientation, we ought to see many similarities to the difference table of Smartphones/Tablets and Standard Controllers, since those shared the same orientation as well.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks	80	90	50
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining	70	60	20

Table 4.65: Differences between Vertical Handheld Motion Controller and Smartphones/Tablets - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60					80
Rapid shape drawing						50		90	
Rapid controller shaking									
Rapid controller rotation									
Single input		40				40			40
Multiple input		50							
Motion control attacks				40					
Fixed time		30		30					
Variable time									
	25					25			
Learning complex combos.									
Attack Chaining	90	80							

Table 4.66: Differences between Vertical Handheld Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.67: Differences between Vertical Handheld Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.68: Differences between Vertical Handheld Motion Controller and Smartphones/Tablets - Gross Motor Abilities

We see that the Smartphone/Tablet can accommodate 3 different challenges (indiscriminate tapping, alternating tapping, and attack chaining) than the vertically oriented handheld motion controller which supports only 1 different challenge (motion control attacks). This mainly stems from the lack of a touchscreen on the handheld motion controller, such that it cannot accommodate tapping challenges. Similarly, smartphones/tablets do not support motion control attacks. Though they have the capability to do so, we speculate they are not popular because of the screen being built-in to the controller, and the social context of playing these games. If the player is performing a motion control attack, they would not be able to see the screen because it would move in their hand. As well, the larger movements may not be viable in social settings like in public transit or at a restaurant.

We see that these controllers are identical for rapid controller shaking, rapid controller rotation, and variable time input challenges. This is likely because both controllers exist in the vertical orientation and are operated single handedly. This would imply that most interaction movements would use the same motor abilities. For example, when rotating a smartphone held vertically, it is intuitive to move our wrist and arms to spin the device. The same intuition comes into play with handheld motion controllers.

We see that these controllers share 6 challenges. In these challenges there is an even distribution of added/different abilities (7 added abilities for smartphones/tablets, and 7 for handheld motion controllers). We found that the handheld motion controller added more gross motor movements to the table. This would make sense as its playing context is usually in a living room or other area with space for large movements. In comparison smartphones/tablets added more wrist abilities such as drawing and tilting. We speculate that those movements are used more to accommodate both the capabilities of the device (drawing on a touchscreen is intuitive) and the space the player has to work with (wrist movements are good alternatives to arm movements in a limited space). Future work should explore whether players experience challenges as being the "same" when the only difference is between wrist and arm movements.

Novel experiences could come from aligning these two controllers in their use of gross motor abilities. It would be interesting to create a mobile game that focuses on gross motor abilities that would be played in a living room style setting. This would also facilitate future work to understand how controller orientation defines experience and affects design.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining			

Table 4.69: Differences between Horizontal Handheld Motion Controller and Smartphones/Tablets - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90					30			
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input		40				40			40
Multiple input		50							
Motion control attacks									
Fixed time		30		30					
Variable time									
Simple Reaction Time	25				25				
Learning complex combos.	90								
Attack Chaining		80							

Table 4.70: Differences between Horizontal Handheld Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Hands)

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking			‘	
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.71: Differences between Horizontal Handheld Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.72: Differences between Horizontal Handheld Motion Controller and Smartphones/Tablets - Gross Motor Abilities

The overall impression with this orientation is that the differences are identical to the standard controller. Both difference tables show only 7 shared challenges, with rapid controller shaking and variable time movement being completely identical. There are two noticeable differences: the appearance of a gross motor option in Simple Reaction Time, and the fact that Handheld Motion Controllers do not support finger swiping for attack chaining. The gross motor ability makes sense as it adapts to the capabilities of the controller; it is possible to have a simple reaction time event on a horizontal handheld motion controller where the reaction is meant to be thrusting the controller. It is surprising that we don't see similar motions available on smartphones/tablets since they have the same sensors. Again, we speculate that it may be because of the play context that we don't see this, and that future work should be done to see whether instances of gross motor simple reaction time exists on smartphones/tablets. The lack of finger swiping for attack chaining is because the handheld motion controller does not have an analog stick, so instead directional input comes from a d-pad which is a button. Though smartphones/tablets don't have analog sticks either, movement and directional input is usually handheld through the touchscreen interface which uses finger swiping as its method of interaction. Generally, we believe that this almost identical breakdown between smartphones/tablets, handheld motion controller held horizontally, and standard controllers emphasizes the effects of orientation and solidifies why more work ought to be done to explore, understand, and push the envelope in that area.

4.4.3 Differences with Full Body Motion Controller

Challenges	**Cognitive Systems**		
	Perception	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.			
Attack Chaining	70	60	20

Table 4.73: Differences between Full Body Motion Controller and Smartphones/Tablets - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40					40			40
Multiple input	50								
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.									
Attack Chaining	90	80							

Table 4.74: Differences between Full Body Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.75: Differences between Full Body Motion Controller and Smartphones/Tablets - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25							
Learning complex combos.								
Attack Chaining								

Table 4.76: Differences between Full Body Motion Controller and Smartphones/Tablets - Gross Motor Abilities

We see that the difference table is dominated by challenges that the smartphone/tablet supports. This makes sense since the current challenge applications of the full body motion controller are so limited. These controllers only share 2 challenges: fixed time movement, and simple reaction time. The only difference between these is the emphasis of fine motor interaction on the smartphone/tablet versus the gross motor emphasis with the full body motion controller. It is interesting to note that though the type of interaction is different, the value (and therefore importance) of each is the same. We speculate that this means on a meta level the challenges feel the same, and so even with different interactions players can recognize these as the same type of challenge. If this is true, it would imply that we may be able to create other instances of challenge where the motor ability used is generally irrelevant. An interesting point to make is that both challenges are from the Reaction Time family, which fall under more cognitively focused challenges. This further strengthens our case that some challenges are more cognitively focused while others are motor focused. If we were able to identify challenges where motor ability is generally irrelevant to a player's understanding and meta experience of the challenge we could begin to empirically classify challenges as either motor or cognitive focused by testing fine versus gross motor instances of a challenge and comparing them.

4.5 Handheld Consoles

Isolating the handheld consoles data, we get the following:

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks			
Fixed time	90	80	
Variable time	90	80	50
Simple Reaction Time	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.77: Handheld Console Analysis - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40					40			40
Multiple input	50	50							
Motion control attacks									
Fixed time	30				30				
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90	80							
Attack Chaining	90	80							

Table 4.78: Handheld Console Analysis - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.79: Handheld Console Analysis - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.80: Handheld Console Analysis - Gross Motor Abilities

Foot, legs, and torso movements are greyed out on this table, since this controller cannot read those inputs. Head motions are left in as the controller has microphone and front facing camera capabilities. Arm motions are also left in since this controller type is equipped with sensors (e.g. gyroscopes, accelerometers) to measure these movements.

We could not find any examples of motion control attack challenges for this controller. This is the controller context with the fewest challenge restrictions since we could find examples for every challenge except one. We speculate this is because handheld consoles combine elements from all the other controller forms (barring full body motion controllers). It has a multitude of buttons and analog sticks like a standard controller, a touchscreen, camera, and microphone like a smartphone/tablet, and gyroscopes and accelerometers like the handheld motion controller. Therefore, is has access to most of the challenges as it acts like other controllers.

We read from this table that there is still a focus on fine motor abilities (specifically finger and wrist movements), which we believe is due to sharing many features of other controller types which also focus on fine motor abilities. We speculate that there may have been some design bias in the creation of handheld consoles; designers of games and game hardware were familiar with standard controllers and the games that they facilitate, so it they may have implicitly tried to recreate the standard controller experience when designing handheld consoles. If this is the case, we should see similar challenge breakdowns when we compare the handheld console to the different controllers. Future work should examine whether this is true by attempting to better quantify the numbers in the analysis; if handheld consoles are based on other controllers, the numbers found for their shared challenges should be the same or extremely similar. Future work should then expand this to determine whether the same challenge implemented across handheld consoles and a matching controller have the same overall player experience because of these similarities (i.e. does single input button mashing on a handheld console feel the same as it would on a standard controller?)

Though it appears to be the most versatile controller, it is not doing well in the market with handheld device ownership among gamers dropping across all age groups and genders since 2014 [16] [2]. This poor market performance has meant that fewer games are being produced for this controller/platform, with the 2018 Game Developer's Conference State of the Game Industry report stating that only 3% of developers report their last completed project being on an established handheld console (Nintendo 3DS, Nintendo Wii U, PlayStation Vita) [5]. This lack of market success implies to us that it's not necessarily

[2] We reference the 2016 ESA Essential Facts report here because it is the most recent version to discuss the statistics of individual platform types, whereas the 2017 report does not mention handheld consoles at all.

Controller	Controller	Weight (g)	Citation
HMC	JoyCon	48-51	[8]
HMC	PlayStation Move	145	[51]
HC	PSP Go	158	[1]
S	Nintendo Pro Controller	203	[6]
S	Dualshock 4	209	[195]
HC	DS Lite	218	[63]
HC	PS Vita (Model 2000)	219	[168]
HC	3DS	235	[137]
HC	PS Vita WiFi Only (Model 1000)	260	[168, 4]
HC	DS	275	[139]
HC	PS Vita 3G (Model 1000)	279	[4]
S	Xbox One Controller	280	[195]
HC	PSP	280	[7]
HC	DSi XL	314	[63]
HC	3DS XL	336	[138]
HC	Nintendo Switch with JoyCons	399	[8]
HC	Wii U Gamepad	500	[9]

Table 4.81: Weights of Different Handheld Consoles

S = Standard, HC = Handheld Console, HMC = Handheld Motion Controller

the potential gameplay movements that make a platform or controller popular just because it has a lot of functionality doesn't mean people enjoy it.

We speculate that two factors may have influenced the popularity of this device: the weight of a handheld console, and the widespread success of smartphones/tablets. We believe weight may be a factor because it induces fatigue over longer play sessions. However, when we compare the weight of several handheld consoles to the prominent standard and handheld motion controllers, we see that they are mostly within the same weight range (see Table 4.81). The outliers are the "larger" models of existing handheld consoles (DSiXL and 3DS XL) and the multi-modal hybrids (Switch and WiiU). While the weight fatigue may factor into issues with the console-handheld hybrids, we have a hard time seeing why it would affect the previous smaller handheld consoles. We believe that further research ought to be done to see whether there is noticeable play session fatigue between standard controllers and handheld consoles, or whether there is a perceived difference for the player.

We believe that the rise in smartphone/tablet ownership and usage is the more prominent factor in the demise of the handheld console. We speculate that this is because the hardware is easily accessible and available, meaning that there is no additional barrier to game playing on these devices. As well, since smartphones/tablets become multi-purpose (phone and game console), players would not need to carry around a standalone gaming device. In 2016 smartphones/tablets were reported as the most popular gaming device with 89% of Canadian adults owning one, and 41% of gamers stating that they most often play games on their mobile device [16]. As well as hardware proliferation, there is an abundance of games for a variety of prices (including free) which makes it easy for people to engage in game playing in a non-committal way as opposed to playing games on another platform which requires investing some money into the actual games. From the ESA 2016 report, among people who do not self-identify as gamers, 79% reported playing games on their mobile devices in the past 4 weeks [16]. From a business standpoint, we can see how this large market would prioritize mobile game development for monetary gain. It is possible that as more mobile games became available, and more people had mobile devices capable of playing games, the market for handheld consoles moved over to investing more in smartphones/tablets thus causing the decline in handheld console sales and development.

It is possible that the handheld console market may change dramatically now that the Nintendo Switch is available. The Switch acts as both a handheld console when the JoyCons are attached but can also switch to be a screen with handheld motion controller inputs through the JoyCons or plug into a larger screen and use the JoyCons with the JoyCon Grip for a standard controller input. This hybridization of console and controller may spark more interest in developing titles for the Switch, which could potentially revitalize the handheld console market and generate novel play experiences. From the 2018 GDC State of the Game Industry Report, we see that while only 5% of developers had published a title for the Switch, 12% were currently working on titles for it, with 15% anticipating working on a title for it soon [5]. While it is possible that this development could revitalize the handheld console market, it is equally possible that many of these developers will focus on the handheld motion controller, or standard controller experiences that the Switch provides. We believe that we need to wait to see how development progresses and allow future work to explore the implications of that development.

4.5.1 Differences with Standard Controller

Challenges	**Cognitive Systems**		
	Perception	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.82: Differences between Standard Controller and Handheld Consoles - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input				50					
Multiple button input									
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input						40			40
Multiple input									
Motion control attacks									
Fixed time		30			30				
Variable time									
Simple Reaction Time			25						
Learning complex combos.									
Attack Chaining									

Table 4.83: Differences between Standard Controller and Handheld Consoles - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.84: Differences between Standard Controller and Handheld Consoles - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.85: Differences between Standard Controller and Handheld Consoles - Gross Motor Abilities

From this table we see that handheld consoles are capable of 5 more challenges than the standard controller (indiscriminate tapping, alternating tapping, rapid line drawing, rapid shape drawing, and controller rotation). While most of these make sense, as the standard controller does not have a touchscreen that can accommodate the tapping and scribbling challenges, it is surprising that handheld consoles have rapid controller rotation. We originally speculated that standard controllers could not support rapid controller rotation because of the awkwardness from holding the controller between two hands (see Section 1.9). However, handheld consoles are held in the same way, but we find an example in the "Hammer Throw" event in Mario and Sonic at the London 2012 Olympic Games for Nintendo 3DS [189]. We speculate that this example only exists because the Mario and Sonic series is a cross-platform release with the Wii and WiiU; therefore, where the Wii version has the player rotate a Wii Remote, the handheld version of this game attempted to emulate the experience by asking the player to "Spin" the controller. Following installments in this franchise do not include this event or this movement anymore.

If we focus on the rows they share, we see there is a significant overlap in abilities with 9 challenges being identical across cognitive and motor abilities. For 3 of the challenges they share (single input attacks, fixed time movement, and simple reaction time) we see that handheld consoles cover all standard controller inputs and adds additional wrist and finger abilities on top of them. This makes sense since handheld consoles have more interaction features (e.g. touchscreen, microphone) than the standard controller; therefore, it can add more motor interactions for a shared challenge. Where a standard controller asks the player to push a button, the handheld console can ask them to push a button, shake the controller, tap on the touchscreen, etc.

The only instance where we find standard controllers using more motor abilities is with Single Input Button Mashing, where it adds an element of wrist shaking. This is because of how button mashing is done on this controller; as we previously discussed (see Section 1.1,1.3,1.2), to optimize button mashing players tend to hold their controller in an unorthodox way that relies on wrist and forearm movements as opposed to finger movements. We don't see this happen on handheld consoles. This is potentially due to the size and shape of the console, as most are blocky and uncomfortable to hold outside of their intended orientation. It is also possible that because handheld consoles put the screen in your hand, holding it in an unintended orientation makes playing the game difficult as it effectively changes your ability to see and react to the onscreen events appropriately. For example, consider a single input button mashing mini-game, where the game switches which button you need to mash periodically. For a standard controller there is no issue in seeing what button needs to be mashed, since the screen is in front of the player and does not move

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks	80	90	50
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.86: Differences between Vertical Handheld Motion Controllers and Handheld Consoles - Cognitive

regardless of controller orientation. For handheld consoles, the screen tilts and rotates with the controller, and has no way to adjust to its new orientation. Combining the screen movement with any wrist and forearm motion and its likely the screen may be obscured from view.

4.5.2 Differences with Handheld Motion Controllers

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60					80
Rapid shape drawing						50		90	
Rapid controller shaking									
Rapid controller rotation									
Single input						40			40
Multiple input									
Motion control attacks				40					
Fixed time		30		30					
Variable time									
Simple Reaction Time	25								
Learning complex combos.	90	80							
Attack Chaining	90	80							

Table 4.87: Differences between Vertical Handheld Motion Controllers and Handheld Consoles - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			**Foot**
	Neck	**Face**		**Ankle/Foot**
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.88: Differences between Vertical Handheld Motion Controllers and Handheld Consoles - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					**Legs**		**Torso**
Challenges	**Push**	**Swing**	**Draw**	**Rotate**	**Position**	**Move**	**Position**	**Position**
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.89: Differences between Vertical Handheld Motion Controllers and Handheld Consoles - Gross Motor Abilities

From this table, we see that handheld consoles and vertically oriented handheld motion controllers share 9 challenges.

Handheld consoles support 8 more challenges than the vertical handheld motion controller (single input button mashing, multiple input button mashing, alternating input button mashing, rapid analog stick rotation, indiscriminate tapping, alternating tapping, learning complex combination moves, and attack chaining). Some added challenges have to do with an increase in functionality for the handheld console; the handheld motion controller does not have touchscreen to tap or an analog stick to rotate. We speculate that the remaining challenges (button mashing and advanced combat) exist on the handheld console due to context bias. Since handheld consoles offer similar features to standard controllers, and are held in the same way, there may be a bias to incorporating more traditional standard controller style challenges onto this controller context. This is entirely speculative, and we urge future researchers to better understand why we see these challenges on handheld consoles.

Handheld motion controllers support 1 challenge that is not found on handheld consoles: motion control attacks. Though handheld consoles have the capability to detect motion, and therefore ought to be capable of motion control attacks, we do not see any in existing games. It is possible that we just don't have the breadth of gaming knowledge to find one, as there are many games we have not encountered. However, we think the lack of appearance of this challenge is likely due to the screen being attached to the controller. As we previously stated, moving the controller would mean moving the screen, which in turn means not being able to see or react to what is happening in the game.

For the challenges that they share, we see 4 are completely identical, while the other 5 seem to trade off abilities between the two controller contexts; handheld consoles focus on more fine motor abilities, while handheld motion controllers opt for gross motor abilities. For example, with rapid shape drawing we see that handheld consoles emphasize wrist drawing motions, while handheld motion controllers use arm drawing and wrist pointing. This makes sense as the size of the drawing canvas for each controller affects the size of motion that is being used. For handheld consoles there is a finite canvas (the touchscreen) which is relatively small; this leads to wrist movements being prioritized as a single wrist movement can cover the whole canvas. In comparison, the canvas for handheld motion controllers is as wide as the controller can be tracked since the player is drawing in the air. This means that arm and shoulder movements need to be made to quickly cover the canvas. We speculate that this difference in canvas size, which leads to a difference in motor abilities should also lead to a measurable difference in experience. We believe future work should look into the measurable differences in fatigue, engagement, timing, and other factors between handheld consoles and handheld motion controllers for all the challenges they share.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.90: Differences between Horizontal Handheld Motion Controller and Handheld Consoles - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input				50					
Alternating button input									
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input						40			40
Multiple input									
Motion control attacks									
Fixed time		30		30	30				
Variable time									
Simple Reaction Time	25				25				
Learning complex combos.		80							
Attack Chaining		80							

Table 4.91: Differences between Horizontal Handheld Motion Controller and Handheld Consoles - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.92: Differences between Horizontal Handheld Motion Controller and Handheld Consoles - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.93: Differences between Horizontal Handheld Motion Controller and Handheld Consoles - Gross Motor Abilities

For horizontally oriented handheld motion controllers, we find 12 shared challenges, and 6 handheld console exclusives (rapid analog stick rotation, indiscriminate tapping, alternating tapping, rapid line drawing, rapid shape drawing, rapid controller rotation). Most of these are because the handheld console supports interaction that the handheld motion controller does not (touchscreen, analog stick). We previously speculated that horizontally handheld motion controllers don't support rapid controller rotation because of how they were held (see Section 4.2.2). However, handheld consoles are held in the same orientation (between two hands) and somehow still support this challenge. As we previously explained, we were only able to find one instance on this challenge on this controller (Hammer Throw [189]). Therefore, since it is possible to have and instance on handheld consoles, it is likely possible to create one on the horizontal handheld motion controller. However, we believe that this challenge may not be considered "fun" or engaging by players because of the awkwardness in the movement (coordinating wrist motions). Though it is possible to create, we may not find more examples because it doesn't make for interesting gameplay. More research should be done to compare the engagement and enjoyment of rapid controller rotation in a horizontal orientation versus a vertical orientation.

For the 12 challenges they share, we find 6 of them are identical across cognitive and motor abilities. The other 6 see mostly fine motor additions from the handheld console with a focus on wrist movements. Generally, we believe this is because the handheld console has more features available than the handheld motion controller in this orientation. We see future contributions from the handheld motion controller because the orientation limits the gross motor potential of this controller, thereby eliminating opportunities for motion control attacks, line and shape drawing, and any larger arm movements like swinging, pushing, etc.

4.5.3 Differences with Full Body Motion Controllers

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking	10	10	
Rapid controller rotation	15	10	
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.94: Differences between Full Body Motion Controllers and Handheld Consoles - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking				85					
Rapid controller rotation								85	
Single input	40					40			40
Multiple input	50	50							
Motion control attacks									
Fixed time	30				30				
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90	80							
Attack Chaining	90	80							

Table 4.95: Differences between Full Body Motion Controllers and Handheld Consoles - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.96: Differences between Full Body Motion Controllers and Handheld Consoles - Fine Motor Abilities (Head and Foot)

	Physical System (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	**Push**	**Swing**	**Draw**	**Rotate**	**Position**	**Move**	**Position**	**Position**
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25							
Learning complex combos.								
Attack Chaining								

Table 4.97: Differences between Full Body Motion Controllers and Handheld Consoles - Gross Motor Abilities

From this table we see that handheld consoles and full body motion controllers only share 2 challenges: fixed time movement, and simple reaction time. For both challenges we see that the cognitive abilities are identical while the motor ability differs between fine and gross abilities. This makes sense as the handheld console can only track fine motor abilities while the full body motion controller can only track gross motor abilities. Therefore, the only way they could share challenges would be by swapping the motor abilities used.

4.5.4 Differences with Smartphones/Tablets

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining			

Table 4.98: Differences between Smartphones/Tablets and Handheld Consoles - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90			50		30			
Alternating button input	90					40			
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking									
Rapid controller rotation									
Single input		40							
Multiple input		50							
Motion control attacks									
Fixed time					30				
Variable time									
Simple Reaction Time						25			
Learning complex combos.	90	80							
Attack Chaining									

Table 4.99: Differences between Smartphones/Tablets and Handheld Consoles - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.100: Differences between Smartphones/Tablets and Handheld Consoles - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.101: Differences between Smartphones/Tablets and Handheld Consoles - Gross Motor Abilities

From this table we see that these controllers share 13 challenges, with 5 exclusive to handheld consoles. The 5 exclusive challenges are: single input button mashing, multiple input button mashing, alternating button mashing, rapid analog stick rotation, and learning complex combination moves. The first four make sense as they rely on physical hardware that smartphones/tablets do not have (buttons and analog stick). We previously discussed why Learning Complex Combination moves don't exist on smartphones/tablets (see Section 4.4). Though the capability ought to be there for smartphones/tablets, the lack of good multi-dimensional input limits the depth of combat moves in a game and reduces the ability to perform combos. In comparison, handheld consoles get the benefit of operating like a standard controller with several buttons and an analog stick.

These controllers have the most overlap in abilities, and the most identical challenges with 2/3rds of the shared challenges being identical. When comparing the non-identical shared challenges we find minimal differences, often related to handheld consoles adding additional motor abilities. We believe that this is potentially due to the versatility of smartphones/tablets as they can be held in both a vertical and a horizontal orientation. Therefore, they can share many challenges with the handheld console and the standard controller when in horizontal mode, and then switch to mimicking the vertical handheld motion controller in its vertical mode. In comparing the feature capabilities of the smartphone/tablet and handheld console, we find it interesting that there are any differences in the shared challenges. With only 4 noticeable differences over 13 shared challenges, we believe further investigation is necessary. Future work should confirm whether these differences do exist and are measurable, or whether they are just a perceived difference based on our analysis method. It is possible that in our analysis of games we have picked outliers or unintentionally omitted examples that would created these differences. We would not want outlier data to end up being overrepresented in the analysis this way; if there's only one example and it informs an entire row, that would be problematic. Alternatively, omitted examples may show us that there are far more differences than we originally saw. We urge future work to explore many more games, and to have a variety of people play and comment on them to get a more accurate representation of how these challenges work.

4.6 Keyboard and Mouse

Isolating keyboard and mouse data, we get the following:

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks			
Fixed time	90	80	
Variable time	90	80	50
	90	75	
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.102: Keyboard and Mouse Analysis - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90								
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50								
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90								
Attack Chaining	90								

Table 4.103: Keyboard and Mouse Analysis - Fine Motor Abilities (Hands)

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck Move	Face Talk	Express.	Ankle/Foot Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.104: Keyboard and Mouse Analysis - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.105: Keyboard and Mouse Analysis - Gross Motor Abilities

We see that it cannot support the tapping (indiscriminate, alternating), scribbling (rapid line, rapid shape), rapid analog stick rotation, rapid controller rotation, or motion control attack challenges. This is because of the limited input methods for this controller. Between the keyboard and mouse, we find buttons, scroll wheels, and mouse movement inputs, all of which focus on hand and arm movements. Since there are no analog sticks, touchscreens, gyroscopes or accelerometers, it can't accommodate the challenges that rely on those types of inputs.

Similarly, because the inputs are limited we see that there is a focus of fine motor abilities with emphasis on hand and arm movements. The head, foot, legs, and torso movements are greyed out. Arm movements remain because they contribute to larger mouse movements like shaking.

The keyboard and mouse are extremely different in design from other gaming controllers, as they were not designed for the specific purpose of gaming; they adopted that purpose with the rise of computer games. Because they are standardized, we find that most players use keyboards/mice in the same way (e.g. WASD controls [246, 258, 259]) whereas there can be some variation in how players hold and interact with other controllers (e.g. Nooch hold [71]). We speculate that changes in how players hold controllers are strategic; intended either to decrease the cognitive or motor load of the challenge (thereby making it "easier"), or to reduce stress and fatigue (thereby making it possible to continue). Future research should further examine whether this is true and compare the effects of changing hold on and between different controllers. For example, the keyboard and mouse is a similar two handed setup to the Wii Remote and the Wii Nunchuck (a primary controller in one hand, and a secondary directional controller in the other). It would be interesting to compare how a dominant hand setup for keyboard and mouse would compete against a dominant hand setup for Wii Remote and Nunchuck in various challenges and game scenarios. Generally, we would like to be able to quantify the motor and cognitive loads of each controller for each challenge (by formalizing the numbers in our table) and compare them to see how different controllers stack up against each other.

4.6.1 Differences with Standard Controller

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation	10		
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.106: Differences between Standard Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input				50					
Multiple button input				50		30			
Alternating button input									
Rapid analog stick rotation		65				30			
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking									
Rapid controller rotation									
Single input									
Multiple input		50							
Motion control attacks									
Fixed time									
Variable time									
Simple Reaction Time		25							
Learning complex combos.		80							
Attack Chaining		80							

Table 4.107: Differences between Standard Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.108: Differences between Standard Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.109: Differences between Standard Controller and Keyboard and Mouse

From this table we see a large overlap in the available challenges, with only 1 difference (rapid analog stick rotation). They are identical cognitively and physically for 5 challenges.

Though they share most of their challenges, we see that Standard Controllers add more finger and wrist motions to how they are completed. This makes sense since the Standard Controller is equipped to measure these things better than the Keyboard/Mouse. Generally, we see the physical experience does not differ much; not having to hold the keyboard and mouse (as they rest on a table) deters a focus on wrist and gross movements, but both focus on fine motor movements with an emphasis on finger pressing (for buttons/keys). However, the extreme overlap in abilities and challenges leads us to believe that the overall "play" experiences are likely the "same". Future work should investigate whether the perceived "play" experience is the "same" between keyboard/mouse and standard controller across all the shared challenges. We believe that understanding when and why two controller types create the same high-level experience will allow us to create better cross platform, and cross controller, experiences.

4.6.2 Differences with Handheld Motion Controller

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks	80	90	50
Fixed time			
Variable time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.110: Differences between Vertical Handheld Motion Controller and Keyboard and Mouse - Cognitive

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90								
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing					81				
Rapid shape drawing						50			
Rapid controller shaking									
Rapid controller rotation								85	
Single input									
Multiple input		50							
Motion control attacks				40					
Fixed time				30					
Variable time									
Simple Reaction Time	25			25					
Learning complex combos.	90								
Attack Chaining	90								

Table 4.111: Differences between Vertical Handheld Motion Controller and Keyboard and Mouse - Fine Motor Abilities (Hands)

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.112: Differences between Vertical Handheld Motion Controller and Keyboard and Mouse - Fine Motor Abilities (Head and Foot)

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing		70						
Rapid shape drawing			90					
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks	40							
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.113: Differences between Vertical Handheld Motion Controller and Keyboard and Mouse - Gross Motor Abilities

This table shows for the vertical orientation of the handheld motion controller, there are only 6 shared challenges. Generally, we see that the differences are drawn across motor lines; the handheld motion controller incorporates many more motor abilities than the keyboard/mouse. For example, Simple Reaction Time is available on both controllers but we see that handheld motion controllers incorporate a wider range of both fine and gross motor abilities (5 abilities for handheld motion controller, 1 for keyboard/mouse). Closely related to motor lines, we also see a division among available inputs, with the keyboard focusing on button pushing challenges (button mashing, advanced combat) while the handheld motion controller focuses on motion based challenges (line and shape drawing, controller rotation). Of the 6 shared challenges, 3 are identical along cognitive and motor lines: rapid controller shaking, single input attacks, and variable reaction time challenges. Future work should confirm whether this is the case, as there is significant context difference to make us question this; for example, rapid controller shaking for the keyboard/mouse combo comes from vigorously shaking the mouse back and forth. This left-right motion for a mouse is measurably different than the erratic shaking capable of a Wii Remote; however, that degree of difference hasn't been captured in this current iteration of the table. We expected these controllers to be extremely different considering that one is designed around gross motor abilities, and the other is designed around fine. It is likely that we will see more overlap between the keyboard/mouse and horizontal handheld motion controller since they share a similar fine motor focus.

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking			
Rapid controller rotation			
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.114: Differences between Horizontal Handheld Motion Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input						30			
Alternating button input									
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking									
Rapid controller rotation									
Single input									
Multiple input		50							
Motion control attacks									
Fixed time				30					
Variable time									
Simple Reaction Time	25			25					
Learning complex combos.									
Attack Chaining									

Table 4.115: Differences between Horizontal Handheld Motion Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.116: Differences between Horizontal Handheld Motion Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time		25						
Learning complex combos.								
Attack Chaining								

Table 4.117: Differences between Horizontal Handheld Motion Controller and Keyboard and Mouse

For the horizontal orientation, we see a greater overlap in available challenges and used abilities. These controllers share all their challenges, with 7 being identical. At a glance, this looks like the standard controller table (Table 4.106, 4.107, 4.108, 4.109), likely due to similar limitations due to the orientation of the two controllers. We do notice that there are fewer differences in motor abilities for the shared challenges, with the handheld motion controller only adding a couple or additional wrist movements.

Though we previously speculated that Standard Controllers and Keyboard/Mouse may have similar meta-level play experiences due to the overlap in challenges and abilities, we would be surprised to find the same true here. Our surprise is likely due to the prevailing perception of motion controls as "casual" (for casual gamers) [81, 261, 260, 245, 257]. This reveals an understated obstacle in games research, community bias; our meta experience of a game is dependent upon its context (controller, platform, aesthetics, genre). When we associate platforms like the Wii with "casuals" or family friendly entertainment and then pit it against a platform that's generally accepted as being "hardcore" (e.g. PC) or mature, our experience skews depending on our own biases towards these labels. This means that even if these controllers could deliver measurably equivalent experiences (i.e. the identical challenges), the subjective experience feedback may show that players consider them wildly different.

4.6.3 Differences with Full Body Motion Controller

Challenges	Cognitive Systems		
	Perception	**Attention**	**Memory**
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping			
Alternating tapping			
Rapid line drawing			
Rapid shape drawing			
Rapid controller shaking	10	10	
Rapid controller rotation			
Single input	80	90	60
Multiple input	80	90	30
Motion control attacks	80	90	50
Fixed time			
Variable time	90	80	50
Simple Reaction Time			
Learning complex combos.	10		70
Attack Chaining	70	60	20

Table 4.118: Differences between Full Body Motion Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90								
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping									
Alternating tapping									
Rapid line drawing									
Rapid shape drawing									
Rapid controller shaking				85					
Rapid controller rotation									
Single input	40								
Multiple input	50								
Motion control attacks									
Fixed time	30								
Variable time	35	30							
Simple Reaction Time	25								
Learning complex combos.	90								
Attack Chaining	90								

Table 4.119: Differences between Full Body Motion Controller and Keyboard and Mouse

Challenges	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
	Move	Talk	Express.	Press
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.120: Differences between Full Body Motion Controller and Keyboard and Mouse

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking		70						
Rapid controller rotation								
Single input								
Multiple input								
Motion control attacks	40							
Fixed time	30							
Variable time								
Simple Reaction Time	25							
Learning complex combos.								
Attack Chaining								

Table 4.121: Differences between Full Body Motion Controller and Keyboard and Mouse

Expectedly, there is extremely limited sharing between these controllers (fixed time, and simple reaction time). At this point we reiterate a pattern that seems to exist across the comparisons; shared challenges differ along the fine-gross motor line. We can see clearly from the shared challenge that the motor ability is the only difference as the importance value is the same regardless of whether it is a fine or gross motor ability.

4.6.4 Differences with Smartphones/Tablets

Challenges	Cognitive Systems		
	Perception	Attention	Memory
Single button input	10	10	
Multiple button input	15	35	
Alternating button input	10	40	
Rapid analog stick rotation			
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction			
Learning complex combos.	10		70
Attack Chaining			

Table 4.122: Differences between Smartphones/Tablets and Keyboard and Mouse

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input	90					26			
Multiple button input	90								
Alternating button input	90					40			
Rapid analog stick rotation									
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input	40					40			40
Multiple input									
Motion control attacks									
Fixed time		30							
Variable time									
Simple Reaction Time		25							
Learning complex combos.	90								
Attack Chaining		80							

Table 4.123: Differences between Smartphones/Tablets and Keyboard and Mouse

	Physical Systems (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.124: Differences between Smartphones/Tablets and Keyboard and Mouse

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation								
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.125: Differences between Smartphones/Tablets and Keyboard and Mouse

These controllers share 7 challenges, 3 of which are identical along cognitive and motor lines. Most of the challenge differences come from input limitations (having a touchscreen, microphone, camera, motion sensors, etc).

We are not surprised at the limited number of shared challenges for these controllers. Common sense would say that the way a device is intended to be used guides the types of challenges we find on it. Generally, phones are intended for short play sessions such as when you're on public transit, while a PC with a keyboard/mouse combo is intended for a longer play sessions. Longer play times means that the player can engage in more cognitively stimulating gameplay, whereas shorter times tend to focus on motor abilities. This is seen in party games and mini-games where individual play instances are short, self-contained and usually focusing on a motor task with minimal cognitive engagement. This design restriction of shorter playtimes influences the types of games being made; if we were to compare popular games on a smartphone/tablet versus popular games on a keyboard/mouse they would be significantly different. In turn, the differences in intended use and subsequent differences in game design mean that the play experience of the same game across these two platforms could be wildly different. Future work should explore whether the meta "experience" of a game can be retained across platforms and controllers, and quantify the differences in the experiences.

4.6.5 Differences with Handheld Consoles

	Cognitive Systems		
Challenges	**Perception**	**Attention**	**Memory**
Single button input			
Multiple button input			
Alternating button input			
Rapid analog stick rotation	10		
Indiscriminate tapping	10	10	
Alternating tapping	30	50	
Rapid line drawing	10	15	
Rapid shape drawing	40	10	
Rapid controller shaking			
Rapid controller rotation	15	10	
Single input			
Multiple input			
Motion control attacks			
Fixed time			
Variable time			
Simple Reaction Time			
Learning complex combos.			
Attack Chaining			

Table 4.126: Differences between Handheld Consoles and Keyboard and Mouse

	Physical Systems (Motor Abilities)								
	Fine Motor Abilities								
	Hands								
	Fingers			Wrist					
Challenges	Press	Swipe	Pinch	Shake	Flick	Point	Swing	Draw	Tilt
Single button input									
Multiple button input				50		30			
Alternating button input									
Rapid analog stick rotation		65				30			
Indiscriminate tapping	90								
Alternating tapping	90					10			
Rapid line drawing				60	81				80
Rapid shape drawing								90	
Rapid controller shaking									
Rapid controller rotation								85	
Single input						40			40
Multiple input		50							
Motion control attacks									
Fixed time		30			30				
Variable time									
Simple Reaction Time		25							
Learning complex combos.		80							
Attack Chaining		80							

Table 4.127: Differences between Handheld Consoles and Keyboard and Mouse

	Physical System (Motor Abilities)			
	Fine Motor Abilities			
	Head			Foot
	Neck	Face		Ankle/Foot
Challenges	**Move**	**Talk**	**Express.**	**Press**
Single button input				
Multiple button input				
Alternating button input				
Rapid analog stick rotation				
Indiscriminate tapping				
Alternating tapping				
Rapid line drawing				
Rapid shape drawing				
Rapid controller shaking				
Rapid controller rotation				
Single input				
Multiple input				
Motion control attacks				
Fixed time				
Variable time				
Simple Reaction Time				
Learning complex combos.				
Attack Chaining				

Table 4.128: Differences between Handheld Consoles and Keyboard and Mouse

	Physical Systems (Motor Abilities)							
	Gross Motor Abilities							
	Arms					Legs		Torso
Challenges	Push	Swing	Draw	Rotate	Position	Move	Position	Position
Single button input								
Multiple button input								
Alternating button input								
Rapid analog stick rotation	90							
Indiscriminate tapping								
Alternating tapping								
Rapid line drawing								
Rapid shape drawing								
Rapid controller shaking								
Rapid controller rotation	70			70				
Single input								
Multiple input								
Motion control attacks								
Fixed time								
Variable time								
Simple Reaction Time								
Learning complex combos.								
Attack Chaining								

Table 4.129: Differences between Handheld Consoles and Keyboard and Mouse

We see that handheld consoles share a significant number of challenges (11) with keyboard/mouse. This is because the handheld console incorporates elements from the other styles of controller. While handheld consoles share features with smartphones/tablets, they have the advantage of being known for slightly longer play sessions and therefore more involved challenges. They are also perceived as slightly less "casual" than smartphones/tablets thus potentially making the overlapping keyboard/mouse experiences more similar than they would be on other controllers.

4.7 Overall

Generally, we find that from looking at these tables our challenge families stand out as being distinct. The Reaction Time challenges very noticeably focus on the cognitive abilities and are recognizably like each other. The Speed challenges are more diverse on the motor end, but all share the same focus of being low in the cognitive abilities and high in one or two motor abilities. Advanced Combat challenges noticeably lie between the two categories, as being motor-oriented challenges with a high cognitive component. These apparent differences imply that we were correct in our original assumption that Speed and Reaction Time challenges needed to be split up as opposed to Adams' original challenge that lumped them together. The fact that this pattern holds even after we separated the tables by controller would imply that the challenges we have constructed are not entirely controller dependent, since the groupings are still distinct between controllers.

It is interesting to note, that there are many challenges that have a low cognitive-high fine motor ability structure, but very few that have a low cognitive-high gross motor ability structure. While we generally expected a cognitive motor trade-off, we didn't expect there to be so few instances of gross motor abilities, especially when considering Handheld Motion Controllers and Full Body Motion controllers. There are two major reasons we could think to explain this: the fatigue of motion, and the unpopularity of motion controls. Both would create contexts where developing challenges for gross motor abilities is ill-advised as they would not provide either a monetary reward (people won't buy it), or a player enjoyment reward (people won't like it). More research needs to be done into whether there are more instances that we were not able to capture in this work, and the development of new challenges to address this area. It may also be worth looking at whether existing challenges can be adapted to use gross motor abilities without changing the overall challenge feel.

Since we were unable to cover all of Adams list it is possible that all issues we have discussed will be addressed as more challenges are deconstructed and analysed.

Part IV

Discussion and Future Work

In this part of the thesis we discuss some potential applications of our work and future work.

Chapter 1

Applications

There are several possible ways that we see our work being used:guiding the creation of novel gameplay experiences, analysing existing and developing games from the point of view of differently abled players, and analysing game series for similarities and differences. We can create novel gameplay by examining "holes" in the table (empty cells) and tailoring gameplay to fit them. We can analyse games for motor inclusivity by changing weighting (importance; number in the cells) of motor abilities and examining alternative motor interactions for different challenges. We can analyse game series by adjusting the values in the table to match each game and comparing how their tables look. The following sections will address these possible applications in some more detail.

1.1 Novel Gameplay

From the data we have compiled (Part III) we can attempt to generate novel gaming experiences by exploring under represented motor abilities or creating new challenge types. As previously stated, this is starting to take shape through games like Chicken Scream and the Snapchat games which use the facial fine motor abilities.

The games that we create are influenced by our understanding of existing game design practices, and so these novel mechanics are only being explored in Speed (Snapchat) and Reaction Time (Chicken Scream) contexts. It would be interesting to see whether facial fine motor abilities can be addressed in Timing and Rhythm challenges through things like moving your head to the music, and Pattern Recognition challenges through mimicking facial expressions. We could also look at using previously uncoupled abilities to see whether we can create new challenges. For example, would we be able to create an Advanced Combat challenge akin to Learning Combination moves where it used exclusively gross motor abilities through performing real martial arts moves.

As this work expands into more challenges we may see some of these “novel” challenges appear, and new ideas may also pop up.

Although it is possible to create novel gameplay, that does not mean it will be well received by players. As it has been previously discussed, there is divide amongst the gaming community along the lines of “casual” and “hardcore”, with many in the latter camp rejecting innovative gameplay like motion controls due to its perceived “gimmick” and “casual” nature [245]. While we do not subscribe to this segregation of “casual” and “hardcore”, we understand that companies may not want to explore novel challenges and gameplay in fear of financial retribution (i.e. poor sales and reviews) from these groups.

The following sections will outline different ways novel gameplay experiences can be created on the different controller contexts.

1.1.1 Standard Controller

For this controller, novel gameplay can be created by focusing on wrist movements. Since standard controllers come equipped with gyroscopes and accelerometers, theoretically we should be able to create challenges that ask the player to tilt, thrust, and generally move the controller. Because the standard controller only has one orientation (held between two hands), to enable these movements the players would be forced to engage in wrist and forearm movements. We could, for example, have a Mario Kart style racing game where directional movement is controlled by tilting the controller (left, right, or back to pop a wheelie on the bikes) - this would open up the avenues to potentially new challenges, but would also address the lack of wrist abilities showing up on the standard controller chart. Similarly arm movements would come into play depending on how individual players decide to thrust and tilt. We could not address the other motor abilities without changing the controller design and capabilities.

1.1.2 Handheld Motion Controller

Novel experiences can be created for this controller in the vertical orientation by expanding the use of wrist and arm motions. It is possible that exer-games such as WiiFit already explore this area, but as those games were not the focus of this thesis we have not explored them. There is likely a large market of younger gamers who would enjoy more challenges in this vein, as they have the energy to offset the fatigue of this control method. Generally, it is possible that as more challenges are added to this table we will see more wrist and arm movements. For example, rhythmic dance games are not covered by the listed challenges and may be a significant area where gross motor abilities become largely important. It is also possible that several Switch games may be addressing this, but as we are currently unaware of their specifics we cannot comment on them in this work.

For the horizontal orientation, we create novel gameplay by reintroducing gross motor actions (thrusting, shaking, tilting). We speculated that these don't currently exist because of how the controller was being held, and that this orientation was used in place of a standard controller. It is highly likely we missed games that use these abilities because of scoping our work to Speed, Reaction Time, and Advanced Combat Challenges. It is possible that as this work continues, and new challenges are added, there will be more ways to create novel gameplay for this controller.

1.1.3 Full Body Motion Controller

Creating novel experiences for this controller type would require expanding the number of challenges that it can be used for. These controllers do not have the capability to accurately detect fine motor abilities (thus the columns being greyed out), so we would not be able to expand the challenges into fine motor territory. We speculate that we could create gross motor capable versions of the fine motor oriented challenges; this would require further examining the existing challenges to distill their experiential essence and then figuring out how to preserve it while changing the motor abilities. For example, we could adapt Rapid Shape Drawing to Rapid Shape Making and require the players to create shapes with their body (i.e. instead of drawing a circle, making a circle with your arms). Similarly we could create instances of Rapid Shape Drawing similar to how it is implemented in the vertically oriented handheld motion controller, where the device tracks your hand drawing the shape instead of a controller. There are several issues with these ideas; firstly, both examples would fatigue the player much more than existing fine motor versions. Secondly, we would need to understand whether it is engaging for players, as challenges that are too tiring or boring would not be novel, just tedious. Lastly, we would need to consider the technical limitations of these controllers; is it possible to reliable track small body parts like hands across the body with these existing controllers? Though we can't address these issues here, this speculation does show that there are areas where innovation can occur. It is also possible that Full Body Motion Controllers thrive in other challenge examples that we have yet to analyse. For example, the Dance Central games were fairly well received for the Kinect [46, 47, 214, 266] and generally praised for their controls. However, we haven't explored Adams' Timing and Rhythm challenges, and so can't see how this controller performs in that area.

1.1.4 Smartphones/Tablets

To create novel experiences on this controller, we believe that more head and face abilities should be explored. Though we currently see them in smaller styled games like the Snappables and Chicken Scream, it would be interesting to create a more involved game revolving around those mechanics. A reason we may not yet see that is the higher level context of when mobile devices are used for gaming. Most smartphone/tablet games are designed for quick play sessions that are repeated throughout the day; this is likely to capitalize on how and when players would be accessing their devices throughout the day, such as while commuting to and from work. Incorporating more facial or vocal interactions may make people uneasy when using their devices in public spaces, and so this may be what has cause designers to shy away from these interaction mechanics. It is also possible that we may see these abilities more used in challenges we have yet to explored, such as Timing and Rhythm games. It may be the case that there exist rhythm games where the player is intended to move their head to the beat, or to use their phone as a karaoke microphone, and we have just yet to explore these instances.

1.1.5 Handheld Consoles

Novel experiences for this controller should start with exploring more facial and neck movements or speaking challenges. Pokemon Amie started this on a small scale, with the player being able to interact with their Pokemon by smiling or tilting their head. We even see short incorporations of speaking or microphone usage in games like Professor Layton and the Diabolical Box which ask players to blow on the microphone to clear sand off a puzzle. These ideas could be incorporated into more complete games instead of being one off gimmicks in larger games. It is possible that more games like this exist, and that given more challenges outside of the Speed, Reaction Time, and Advanced Combat families we would see them appear. It is also possible that we could see things like this become more popular with the rise of the Switch; however, it may take time to see whether development heads in that direction.

The other way to explore novel experiences would be to begin incorporating more gross motor abilities. We believe this could be done similarly to how we recommended the Standard Controller incorporate more gross motor abilities; by creating instances of controller tilting, thrusting, and other motions to affect something in game. A similar experience was created on the Gameboy Colour with Kirby Tilt'n'Tumble [83], where the game cartridge came equipped with an accelerometer calibrated for the cartridge to be held vertically. The player would then navigate through the world by tilting the controller to roll Kirby around on the screen. [1] Therefore, we know that these types of controls are possible and it would just be a matter of expanding them into other contexts.

1.1.6 Keyboard and Mouse

Novelty in this context would come from expanding the challenges it can accommodate. We generally expected many challenges to not be present in this context because of the input limitations; without motion sensors, you can't have motion challenges. Similarly, we can't expect to expand the existing challenges into other motor configurations (e.g. gross motor simple reaction time) since the capabilities are not there. We speculate that keyboard and mouse set up would support more cognitively focused challenges, such as strategy, tactics, and logistics challenges, since they often require a high input throughput (a lot of inputs very quickly) and/or extremely accurate inputs. We may also see it expand greatly through accuracy and precision challenges through genres like First Person Shooters, as the mouse is still proven to be the most accurate pointing device [270, 133, 132, 172]. At this point we need to discuss the difference between expert level players who have logged significant gaming hours, and casual players. Expert level players are more keen on optimizing their play. We see this in the eSports industry, where players are constantly attempting to optimize their play by improving throughput, accuracy, and game meta-knowledge. Casual players are less focused on optimal play, and may trade accuracy and throughput for familiarity, comfort, or novelty. Generally, we believe that novelty in challenges and usage for the keyboard/mouse combo will come through games geared towards expert level players and so can leverage the qualities of the keyboard/mouse.

1.2 Game Analysis

We can also use this data to analyse completed and in-development games from the viewpoint of differently abled individuals to promote more inclusive play. According to the ESA Yearly Essential Facts reports, the average gamer age has been increasing from 2012, and is currently 36 years old [12, 13, 14, 15, 16, 17]. It is likely that this average age will continue to increase with the aging gamer population. This means that game designers will need to accommodate for motor skill differences between different age demographics who will be playing the same competitive games. Similar adjustments could be made for people with motor disabilities, or at different levels of cognitive

[1] Kirby Tilt'n'Tumble was not included in existing analyses for 2 reasons: 1) it is the only instance of this that we were aware of, and we did not want one outlier to contribute an entire section of data. 2) Navigation is not a Speed, Reaction Time, or Advanced Combat challenge, and so there was no place to discuss tilt navigation mechanics.

development. Generally, we believe our analysis tables could be used as a tool for designers to check whether the challenges in their game are "playable" (within acceptable cognitive and motor range) by their target demographic.

1.3 Internal Game Series Analysis

Finally we could use this relationship analysis to quantify how similar games in the same series are. When informally describing how a game feels and plays, players tend to make analogies between it and other commercial games, such as "God of War has Dark Souls combat". While this intuitively means something between avid players, we would be able to compare what this means in terms of actual challenges in the games. By analysing the combat in both games we could see whether that just means the same challenges are being used, or whether it runs deeper such as the same challenges with the same motor relationships are used. Our work would also allow us to analyse the similarities between games within the same series. We can isolate instances where new challenges are incorporated in the series and compare the success of that instance to the previous entries of the series. By doing this for several franchises, we could begin to understand the inter-challenge relationships (which challenges form common sets that make up easily identifiable types of gameplay). From there we could begin to experiment with these sets; understand how they are affected when challenges are added or removed, how the player interprets them when the same challenges are used but emphasized in different amounts, etc. This would allow us to begin understanding games more formally, as well as allowing us to attempt to craft more novel gameplay experiences.

Chapter 2

Future Work

From the compiled analysis (Chapter 4) and applications discussion (Section 1) we realise we want to refine our methodology and refine the existing data. We also want to explore questions about the impact of controller type and fatigue on player experience of a game.

2.1 Refining Methodology

Though we followed a methodology that we believe to be systematic, we recognize there is room for improvement. There are two general ways to refine our methodology: adding more to the framework, or further exploring the areas we have covered.

2.1.1 Framework Additions

Adding to the framework would mean adding to either the player or challenge model.

For the player model we believe the first area to be addressed needs to be cognitive abilities as they are the focal point of many high-level challenges and are key in understanding the game-player relationship. We encourage future researchers to explore the following areas to expand this knowledge: cognitive architectures like ACT-R and SOAR, neural networks, cognitive psychology, child developmental psychology, and neuroscience. We recommend these areas for both a theoretical understanding of the neural processes and the structure of cognition, as well as an implementation of these cognitive theories. It is our hope that in exploring the existing cognitive models and their software implementations that we could distill more information about how the human and game systems interact. However, we warn against delving too far into neuroscience and neurobiology as it is possible to get too caught up in the biological structures and lose focus of the cognitive processes.

For the challenge model we believe that more challenges need to be added to this list. This would mean returning to Adams' List (see Table 1) and deconstructing the rest of its challenges. This would give us many different types of games to explore, and would open up the world of cognitively complex challenges such as Strategy, Tactics, and Logistics for us to examine.

2.1.2 Framework Expansions

Expanding on this framework would mean expanding the existing player and challenge models.

For the player model we believe that the motor abilities could be further clarified. Though our current method for distilling motor abilities is systematic, and makes sense in the context of this research, it is possible that certain motions are being overlooked due to the bias in the controllers and games we were examining. It is entirely possible that there are many more fine and gross motor movements that are possible and used in video games but are not fully explored here. For example, we don't cover specialty controllers like the Rockband Drum Set (Figure 2.1); this controller requires you to hit the drum pads with the sticks (a combination arm movement) and occasionally press the pedal with your foot (a combination leg, and foot movement). Stepping motions, necessary for things like the foot pedal on a drum or the accelerator on a racing game rig, can't be captured by this table. Similarly, we don't cover multi-hand or "split" controllers (e.g. Wii Remote and Nunchuck, two Joy-cons) which leads to an issue of neglecting certain coordinated motor abilities. In addition to adding more motor abilities, we believe future work should create more data through more analysis tables for niche controller types (Rockband Drum Set) and "split" controllers.

Another way that the motor abilities could be expanded is to cover more passive motions; things like grip or other hand positions are important in holding a controller and become more important over longer play sessions or with controllers that require gross motor abilities (like the Wii Remote) to make sure it doesn't drop out of the player's hands. This would help us to better understand how to design controllers that are functional but not reliant on being gripped by the player for demographics with low grip strength.

Another way to refine the motor abilities is to eliminate the multi-column cells; doing this would require running several experiments to break these rows with multi-column cells in their own challenges with individual motor configurations. Doing this may give us insight into the issue of challenge similarity (if the only thing that differs are the motor controls, is the challenge experienced in a similar way?).

To expand the challenge model we recommend incorporating more games into the analyses. Though we have an extensive amount of games referenced

Figure 2.1: Rock Band Drum Set for Xbox 360 [104]

in this thesis, it is currently limited to only games that we have played. There are many games that we have not encountered, or even heard of, and it is entirely possible that they contain new challenges that need to be added to this model. By incorporating more games into the analysis, we ought to get a better understanding of the perimeters of individual challenges and what challenges coexist successfully. It ought to give us more data as well to understand the motor aspect of challenges, as every instance of a challenge can be interacted with slightly differently. Incorporating more games should also ease any concerns that the data for challenges with only one or two examples is skewed to be too instance specific. To be specific about the types of games that ought to be further explored, we believe that more Nintendo Switch and full body motion controller games should be covered. These are both areas which seem to have lost some popularity in recent years; with new technology there will be new ways to interact with a game and new challenges to play both of which merit further examination.

We also believe that we ought to review our challenge deconstruction methodology. We chose to examine Adams' challenges as they were the most complete list of challenges we could find at this point. However, using his list as a starting point could potentially have limited our own understanding of the challenges that exist in games. Since we focused on understanding, deconstructing, and reconstructing challenges based on his definitions we are implicitly limited by both Adams language, and understanding of games and gameplay. Therefore it is possible that novel gameplay challenges are left out of our challenge model because they had no place in Adams original listing. This could be corrected by expanding the games analysed and cataloging their individual challenges to see whether any are not captured by our model.

2.2 Refining Data

Since the numbers that we provided are approximates based on our subjective understanding of how we interact with a particular challenge, we believe that future researchers should look to empirically pin down these numbers. To refine the data for this entire thesis would take a minimum of 378 experiments (1 experiment per table cell). We believe future work should pursue a subset of these challenges to verify the numbers in the table. Finding more correct numbers would allow for us to compare different controller types for similarities in a challenge. It would also allow us to better predict potential trends in gaming, or potential issues with demographics. It would also allow us to validate this research by comparing the experimentally gathered numbers and comparing them to our own; if the same trends appear, it stands to reason that our general interpretations were on the right track.

2.2.1 Example Experiments to Refine the Data

There are two types of experiments that we propose: 1) refining the data for individual controller types, 2) refining the data for comparing controllers. The data for individual controller types needs to be refined before the comparison data can be collected.

The following sections will discuss examples of these experiments:

Refining Controller Data

Refining the data serves two purposes: 1) checking whether our idea of cognitively-focused and motor-focused challenges is correct (there is a distinction), and pinning down more accurate numbers for the purposes of analysis. We believe that examining Single Input Button Mashing and Fixed Time Movement challenges to start will best serve this purpose. This is because both challenges can be found as individual gameplay instances (not as part of a composite challenge), and so there won't be competing goals for the player. By design, this would reduce a significant number of confounding variables such as the inclusion of higher order thinking processes to prioritize and play optimally given various goals. We believe that to start, these should all be considered in the context of the Standard Controller. We would focus first on the Standard Controller to use it as a baseline against the other controllers.

For Single Input Button Mashing we we would need to run 3 experiments to understand the finger pressing, wrist shaking, and wrist pointing abilities. Refining these numbers would require a 2 fold process: firstly, we would need to have participants play through the examples and monitor which muscles are used and what actions are taken. This would off the bat let us know whether the motor abilities selected were correct, and allow us to adjust which motor abilities are used before attempting to see how much they contribute to success. Secondly, we would need to isolate each motor ability and test at different levels of hindrance how well the participants succeed at the button mashing task. Similarly we would need to test the way that cognitive abilities influence this. This would mean running through these experiments with controls around the attentional and perceptual processes.

For Fixed Time Movement challenges we run into the issue of cognitive abilities being the central focus of the challenge. This would mean that we would need to better understand the various mechanisms for attention and perception in order to isolate them from finger pressing. We would need to take special care for confounding variables as it is impossible to completely isolate the cognitive and motor processes; we can attempt to overload one process at a time (i.e. asking participants to perform a cognitive task to take up their attentional resources while playing the game, adding additional components to take up perceptual resources, or making it difficult to press the button) which may then show us the relative importance of each ability against each other.

We list below the important experiments to run in starting to refine controller data, in order:

1. Pin down the motor numbers for Single Input Button Mashing on the Standard Controller
   (a) Identify all motor abilities used in Single Input Button Mashing
   (b) Isolate and verify the numbers for Finger Pressing
      i. Run through experimental levels with no hindrances
      ii. Run through experimental levels with attentional hindrances (no motor or perceptual)
      iii. Run through experimental levels with perceptual hindrances (no motor or attentional)
      iv. Run through experimental levels with motor (finger pressing) hindrances (no cognitive)
      v. Run through experimental levels with motor (finger pressing) and attentional hindrances (no perceptual)
      vi. Run through experimental levels with motor (finger pressing) and perceptual hindrances (no attentional)
   (c) Isolate and verify the numbers for Wrist Shaking
      i. Run through experimental levels with no hindrances
      ii. Run through experimental levels with attentional hindrances (no motor or perceptual)
      iii. Run through experimental levels with perceptual hindrances (no motor or attentional)
      iv. Run through experimental levels with motor (wrist shaking) hindrances (no cognitive)
      v. Run through experimental levels with motor (wrist shaking) and attentional hindrances (no perceptual)
      vi. Run through experimental levels with motor (wrist shaking) and perceptual hindrances (no attentional)
   (d) Isolate and verify the numbers for Wrist Pointing
      i. Run through experimental levels with no hindrances
      ii. Run through experimental levels with attentional hindrances (no motor or perceptual)
      iii. Run through experimental levels with perceptual hindrances (no motor or attentional)

iv. Run through experimental levels with motor (wrist pointing) hindrances (no cognitive)
v. Run through experimental levels with motor (wrist pointing) and attentional hindrances (no perceptual)
vi. Run through experimental levels with motor (wrist pointing) and perceptual hindrances (no attentional)

2. Pin down the motor numbers for Fixed Time Movement on the Standard Controller
   (a) Identify all motor abilities used in Fixed Time Movement
   (b) Isolate and verify the numbers for finger pressing
      i. Run through experimental levels with no hindrances
      ii. Run through experimental levels with attentional hindrances (no motor or perceptual)
      iii. Run through experimental levels with perceptual hindrances (no motor or attentional)
      iv. Run through experimental levels with motor hindrances and no cognitive hindrances
      v. Run through experimental levels with motor and attentional hindrances (no perceptual)
      vi. Run through experimental levels with motor and perceptual hindrances (no attentional)

These are just preliminary breakdowns of what we think needs to be tested to verify these numbers. It is almost certain that when these experiments are properly designed we will find many more test cases to examine and variables that need to be controlled for.

Refining Comparison Data

After refining a couple of the controller data sets, the next step would be comparing data sets against each other. In selecting Single Input Button Mashing and Fixed Time Movement challenges we have given ourselves 3 and 6 other controllers to choose from respectively [1]. We believe it would be most interesting to study the Handheld Console (as it has the most available challenges), Smartphone/Tablet (as it is an extremely common gaming device), and the Full Body Motion Controller (as it has the most room for growth). We would need to run similar experiments to those outlined in Section 2.2.1 for each of these listed controllers. From there comparing the data would follow the same process as our current analysis.

[1] SIBM exists across 4 controllers, FTM exists across 7

2.3 Controller Impact on Challenge Experience

Future work should explore the impact of different controllers on challenge motor experience. We hypothesized that any identical challenges between controller types may be, on some level, “experienced” in the same way. To verify this, we would need to explore and quantify the actual similarities and differences between controllers for particular challenges. For example, the horizontal handheld motion controller is almost identical in challenges to the standard controller which we speculated was due to the orientation in which these controllers are used. Confirming this would lead to greater questions about the effect of controller orientation on not just the player’s experience, but challenge design (would a challenge designed for a vertical controller feel the same when implemented on a horizontal controller). We would also want to cover other cases where there’s an overlap of identical challenges such as with the handheld console and standard controller. We believe that the first step in accomplishing this is filling in the current data with more examples from each controller type. A larger sample size of games could indicate more to us about whether these experiences are similar or whether our limited game knowledge just makes it appear that way. In adding more examples it would also allow us to address the issue of outlier examples that may currently skew the data. For example, we see that the Keyboard/Mouse combination is identical to the Vertical Handheld Motion Controller for Rapid Controller Shaking which intuitively seems wrong because of the way you hold each controller (the mouse on a table, the handheld like a baton in the air). Adding more examples would allow us to better refine the numbers and see whether the underlying motor abilities are actually identical.

Related to this idea of controller impact on motor experience, we believe future work should explore how controllers factor into the meta-experience of a challenge. The meta-experience of a challenge is influenced by factors outside of the game itself, like public perception of a game, controller, or genre. We found in our analysis that there were many examples of challenges where the cognitive and motor abilities were identical between different controller. We believe that further investigation should be done into whether the meta-experience is identical in the same way the physical experience seems to be. Understanding whether the physical experience corresponds to the meta-experience would allow us to better understand cross-platform play and predict what types of challenges would feel the same when experienced in different contexts.

2.3.1 Fatigue

A specific component of experience we speculated on was fatigue and its bidirectional impact on players and challenges. Fatigue is the result of repetitive movements over sustained play sessions, with the amount of fatigue being related to the types of movements that the game is making you perform. For example, pushing a button is a low fatigue activity, and so a player ought to be able to repeat this action many times over a long play session without being fatigued to the point of stopping play. In comparison, jumping around on a DDR mat to hit arrows in time with a song beat is a high fatigue activity, and so a player may only engage with these movements for shorter play sessions. Regarding our thesis, we are interested in the effects of fatigue on player interactions with a challenge, how it influences challenge design, and how it is affected by different controller types and configurations.

We are interested in the effects of fatigue on player interactions with a challenge. Cognitive and motor abilities become fatigued when overloaded or used for an extended period of time. This affects play style as the player can no longer use their abilities to their fullest, and so may change the abilities they use to compensate for that fatigue. If fatigue affects play style, it would imply that the relationship between player and gameplay (as represented by the numerical data in our analysis table) changes with play session length. This would mean that we would need to study all the challenges we have laid out in both a short play length and long play length context to see how this change plays out. We could then test whether we can create a predictive model of fatigue and how it affects play. This would leave to testing whether a predictive fatigue-play model holds true across different demographics, or if there are other confounding factors. Understanding how fatigue influences play would also allow us to develop more fatigue-friendly or fatigue-centric challenges, effectively creating challenges whose difficulty or mechanic revolves around the idea that the player becomes more fatigued as they play.

We are interested in how fatigue influences challenge design because it would allow us to better understand how to couple challenge types into compelling gameplay. We would obviously not want to overload a player by demanding they complete high fatigue challenges back to back. Similarly, we would not want to couple high fatigue challenges to be done simultaneously (e.g. jumping jacks - fatiguing arm and leg movements - while solving math equations - cognitively fatiguing). We believe that over fatiguing gameplay would not be compelling to many demographics, and would be bad design.

Understanding how fatigue influences challenge design would also allow us to understand how to design novel experiences for certain controller types. For example, when discussing adding arm movements to standard controllers to explore the potential gross motor movements available with the equipped gyroscope/accelerometer we now know to discuss the intended play session and environment so as to adjust for the levels of fatigue that would be felt. We speculate that when designing gameplay experiences the overall "energy consumption" for the player needs to be considered. A set of challenges that work well together on a Standard Controller may be too taxing when translated to a Full Body Motion Controller.

We are also interested in how fatigue differs across controller types and configurations. This opens a world of questions as we can begin to compare every controller type against each other. One area we believe needs to be researched is the difference between fatigue and play experiences for single handed handheld motion controllers (e.g. sword combat in Legend of Zelda: Skyward Sword [144], and the two-handed handheld motion controller with a peripheral attachment (e.g. shield bash in Legend of Zelda: Skyward Words [144]. It is possible that needing to move the extra controller (peripheral) adds to the overall motor load of the challenge and modifies the used abilities in our table, which in turn would add to the fatigue of this challenge. It is important to understand the effects of using split controllers in games as it could influence how new challenges are designed. This also leads us to the question whether players view challenges as being the "same" when they use different combinations of motor abilities. For example, would players perceive rotating a tumbstick to execute a fighting game move to be the same as performing the same action on a larger joystick? One uses exclusively fine motor abilities, while the other focuses on a combination of fine and gross motor abilities. It is possible that research will show this perception depends on player expertise: a casual player may see them as the same, while an expert player that understands the hardware limitations of each controller will view them as distinct. Another aspect to consider is controller weight over play sessions, and how that impacts fatigue. In the same way that peripherals may add to motor load, weight of a device carried over a long play time may significantly affect motor performance. This is something that should be looked into to see whether some controllers better support certain challenges that require more physical intensity. Similarly we believe that more work needs to be done in comparing abilities between handheld consoles, handheld motion controllers, and smartphones/tablets. We speculated in our analyses (Section III) that their controller design created overlapping experiences between these devices (shared challenges) but also limited the types of challenges they could afford. By more thoroughly analysing and quantifying the similarities and differences of these devices along the lines of challenges, experiences, and fatigue we would have a better understanding of how challenges are afforded by controller design. We would also then be able to recommend crossover gameplay experiences based on their similarities, allowing for more cross-platform novel gameplay. One of the questions that could be answered from this is the importance of fatigue and haptic feedback in scribbling challenges across these controllers.

2.3.2 Experiments Regarding Experience

We have raised many questions through the course of this work regarding experience. Below are the questions we feel are most important to be answered, as well as potential experiments that can be run to address them:

Questions	Experiments
1. Are challenges with identical cognitive/motor configurations on different controllers "experienced" the same way?	Compare Single Input Button Mashing and Variable Time Movement between Standard Controllers, Horizontal Handheld Motion Controllers, Handheld Consoles, and Keyboard and Mouse at both a motor/cognitive level and higher "experiential" level
2. Can challenges with different cognitive/motor configurations on different controllers be "experienced" the same way? (Does presentation matter more to experience than interaction mechanics?)	Compare Simple Reaction Time across all controllers
3. Do 2 handed or "split" controllers significantly affect the cognitive motor load?	Compare Vertical Handheld Motion Controller challenges against a Vertical Handheld Motion Controller with a Peripheral
4. Do cognitive and motor fatigue affect gameplay in the same way?	Compare challenge success data and the motor/cognitive data between motor fatigued and cognitively fatigued players

Table 2.1: Questions to Answer Regarding Experience

Part V

Conclusion

The purpose of this work is to study how motor and basic cognitive abilities are used to interact with gameplay challenges. To that end we surveyed different gameplay challenge models and determined that they were insufficiently detailed for our purposes. We created from this survey a unified definition of challenges (Definition 1.2.1), and a definition of when two challenges are the same or different (Definition 1.2.2). We created our own gameplay model based off Adams Challenges [10] using the same/different definition to expand the definitions of two of his challenge types (Speed and Reaction Time, and Learning Combination Moves). Our gameplay model identified 18 unique challenge types from these two (ChapterI). These challenges were divided into 3 challenge families [1] (Speed, Reaction Time, Advanced Combat) based on our understanding of the core aspects of each challenge. For example, the core quality of Single Input Button Mashing is physically pressing the button faster than your opponent. This core quality is related to physical speed, and so this challenge would fall in the Speed family. In comparison the core quality of Simple Reaction Time is how quickly the player processes on screen stimuli, and so would fall into the Reaction Time family. We also surveyed different approaches to player modeling, which led to identifying 21 distinct motor abilities (Part II). Our player model gave us a way to discuss how players of differing abilities are impacted by challenge design.

We then were able to compare the relationship between them through analysing gameplay instances. We did this systematically by picking a challenge, finding at minimum four examples of it in existing commercial games, playing through these instances multiple times and recording the motor and cognitive abilities used based on our player model, and then analysing their importance relative to each other. We repeated this process for all the challenges we outlined in our gameplay model. We summarize these results in several analysis tables one hardware independent one, and several hardware specific ones (Chapter 4).

Our analysis tables were able to visualize a lot of data about how games are used, and areas for improvement. The first thing that we noticed is that our challenge families were able to distinctly be seen in our analysis tables across input types. This indicates to us that our separation of challenges seemed to make sense. Through our analysis we came up with a baseline for how an “average player interacts with the challenges. We can use this understanding to compare against the abilities of other demographics to see whether certain challenges are “unplayable.

Our analysis helped us to find areas for novel gameplay by examining empty segments of the tables. We were able to outline various ways that each controller type could explore novel gameplay ideas with different motor focuses (Section 1). For example, the Standard Controller is capable of detecting

[1] sets of challenges that share a similar ability configuration

wrist/forearm and arm movements but it is underexplored in current gameplay; we propose a new control scheme for a Mario Kart style racing game that uses controller tilt to direct the car. Our analysis also gave us a way to analyse existing video games, both independently and against other games in their series.

We outline how to continue this work by refining the methodology and analysis data. We explore several ways to refine the methodology, including: expanding the player model, adding more challenges to the game model, and reconsidering whether Adams challenges were the best basis for our challenges. We also explored how to experimentally refine our data; our analysis generates upwards of 378 experiments to run to really quantify these relationships.

We also discuss different questions that our work has brought up regarding the player experience. We end by giving a prioritized list of experiments that could be run to answer the most interesting of these questions.

Bibliography

[1] (2009). Official psp go specs.

[2] (2012). Mobile games winning the war for your attention.

[3] (2012). Why a keyboard and mouse provides a better gaming experience.

[4] (2013). Ps vita tech specs - ps vita wiki guide.

[5] (2018). *Game Developer's Conference - State of the Game Industry Report 2018*.

[6] (2018). Nintendo wii u pro controller specs.

[7] (2018). Sony psp specs.

[8] (2018). Technical specs - nintendo switch official site - system hardware, console specs.

[9] (2018). Wii u - technical specs.

[10] Adams, E. (2010). *Fundamentals of Game Design*. New Riders, Second edition.

[11] Apple (2018). Compare iPad Models.

[12] Association, E. S. (2012). Essential facts about the computer and video game industry.

[13] Association, E. S. (2013). Essential facts about the computer and video game industry.

[14] Association, E. S. (2014). Essential facts about the computer and video game industry.

[15] Association, E. S. (2015). Essential facts about the computer and video game industry.

[16] Association, E. S. (2016). Essential facts about the computer and video game industry.

[17] Association, E. S. (2017). *Essential Facts About the Computer and Video Game Industry 2017*.

[18] Atlus (2006). *Trauma Center: Second Opinion*. Game [Wii]. Atlus USA, California, USA.

[19] Atlus (2007). *Trauma Center: New Blood*. Game [Wii]. Atlus USA, California, USA.

[20] Atlus (2010). *Trauma Center: Trauma Team*. Game [Wii]. Atlus USA, California, USA.

[21] Bandai Namco Games, S. and tri-Crescendo (2014a). *Super Smash Bros. for Nintendo 3DS*. Game [3DS]. Nintendo, Kyoto, Japan.

[22] Bandai Namco Games, S. and tri-Crescendo (2014b). *Super Smash Bros. for WiiU*. Game [WiiU]. Nintendo, Kyoto, Japan.

[23] Barr, P., Noble, J., and Biddle, R. (2007). Video game values: Human–computer interaction and games. *Interacting with Computers*, **19**(2), 180–195.

[24] Bartle, R. (1996). Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD research*, **1**(1), 19.

[25] Bateman, C. and Boon, R. (2005). *21st Century Game Design (Game Development Series)*. Charles River Media, Inc.

[26] Bee Square Games (2018). *Almost a Hero*. Game [Android]. Bee Square Games, Barcelona, Spain.

[27] Biery, T. (2016). Fan crushes dark souls 3 using only dance pads.

[28] Big Pixel Studios (2016). *Pocket Morty's*. Game [Android]. Adult Swim Games, Georgia, USA.

[29] Bjork, S. and Holopainen, J. (2004). *Patterns in game design (game development series)*. Charles River Media.

[30] Bosser, T. (1993). Models of user behaviour and measures of usability. In M. Janse and T. Harrington, editors, *Human-machine communication for educational systems design*, pages 89–98. Institute for Perception Research.

[31] Bungie (2001). *Halo: Combat Evolved*. Game [Xbox]. Microsoft Game Studios, Washington, USA.

[32] Burgess, S. and Ironmonger, D. (2014). Controller for video game console. US Patent 8,641,525.

[33] Buro, M. and Furtak, T. M. (2004). Rts games and real-time ai research. In *Proceedings of the Behavior Representation in Modeling and Simulation Conference (BRIMS)*, volume 6370.

[34] Capcom (2011a). *Devil May Cry 4*. Game [iOS]. Capcom, Osaka, Japan.

[35] Capcom (2011b). *Ultimate Marvel vs. Capcom 3*. Game [Xbox 360]. Capcom, Osaka, Japan.

[36] Capcom Production (2005). *Resident Evil 4*. Game [Gamecube]. Capcom, Osaka, Japan.

[37] Card, S. K., Newell, A., and Moran, T. P. (1983). *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.

[38] Carroll, J. B. (1993). *Human cognitive abilities: A survey of factor-analytic studies*. Cambridge University Press.

[39] Carroll, J. M., editor (2003). *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[40] CD Projekt Red (2007). *The Witcher*. Game [Microsoft Windows]. Atari, New York, USA.

[41] CD Projekt Red (2012). *The Witcher 2: Assassins of Kings*. Game [Xbox 360]. Atari, New York, USA.

[42] CD Projekt Red (2015). *The Witcher 3: Wild Hunt*. Game [Xbox One]. WB Games, California, USA.

[43] Charness, N. and Boot, W. R. (2009). Aging and information technology use. *Current Directions in Psychological Science*, **18**(5), 253–258.

[44] Chen, L. (2016). Games by the numbers q3 2016: Overview of the mobile games landscape unity blog.

[45] Choi, D., Könik, T., Nejati, N., Park, C., and Langley, P. (2007). A believable agent for first-person shooter games. In *AIIDE*, pages 71–73.

[46] Clements, R. (2010). Dance central kinect review.

[47] Clements, R. (2011). Dance central 2 review.

[48] Climax Studios (2009). *Silent Hill: Shattered Memories*. Game [Wii]. Konami, Tokyo, Japan.

[49] Coffee Stain Studios (2014). *Goat Simulator*. Game [Android]. Coffee Stain Studios, Skvode, Sweden.

[50] Colman, A. M. (2018). *A Dictionary of Psychology*. Oxford University Press.

[51] Contributor, G. (2010). Full tech specs: Playstation move.

[52] contributors, V. (2018). Fury (final fantasy x).

[53] Cooking Mama Ltd (2006). *Cooking Mama*. Game [Nintendo DS]. Majesco Entertainment, New Jersey, USA.

[54] Cooking Mama Ltd (2007). *Cooking Mama: Cook Off*. Game [Wii]. Majesco Entertainment, New Jersey, USA.

[55] Djaouti, D., Alvarez, J., Jessel, J.-P., Methel, G., and Molinier, P. (2008). A gameplay definition through videogame classification. *International Journal of Computer Games Technology*, **2008**, 4.

[56] Drake, A. (2011). Eiji Aonuma: Zelda Series Sticking with Motion Controls.

[57] Edmonds, R. (2017). Why mechanical keyboards are better than traditional membrane keypads.

[58] Epyx (1985). *Summer Games II*. Game [Commodore 64]. Commodore Gaming, Pennsylvania, USA.

[59] Eysenck, M. and Keane, M. (2010). *Cognitive Psychology: A Student's Handbook, 6th Edition*. Taylor & Francis.

[60] Federoff, M. A. (2002). *Heuristics and usability guidelines for the creation and evaluation of fun in video games*. Ph.D. thesis, Citeseer.

[61] Feil, J. and Scattergood, M. (2005). *Beginning game level design*. Thomson Course Technology.

[62] Fernández-Vara, C. (2009). Play's the thing: A framework to study videogames as performance. *Proceedings of DiGRA 2009*.

[63] Fletcher, J. (2016). Dsi xl faq: Sized up.

[64] Francis, B. (2016). Q&A: What Insomniac has learned from working on VR brawler Feral Rites.

[65] FromSoftware (2009). *Demon's Souls*. Game [PS3]. Atlus USA, California, USA.

[66] FromSoftware (2011). *Dark Souls*. Game [Xbox 360]. Namco Bandai Games, Tokyo, Japan.

[67] FromSoftware (2014). *Dark Souls II*. Game [Xbox 360]. Namco Bandai Games, Tokyo, Japan.

[68] FromSoftware (2015). *Bloodborne*. Game [PS4]. Sony Computer Entertainment, California, USA.

[69] FromSoftware (2016). *Dark Souls III*. Game [Xbox One]. Bandai Namco Entertainment, Tokyo, Japan.

[70] Gach, E. (2017a). Speedrunner beats every cuphead boss using ddr pads.

[71] Gach, E. (2017b). This guy prefers to play smash bros. melee with the controller upside down.

[72] Gaijin Games (2013). *Bit.Trip Presents...Runner2: Future Legend of Rhythm Alien*. Game [WiiU]. Aksys Games, California, USA.

[73] Galotti, K. M. (2015). *Cognitive development: Infancy through adolescence*. Sage Publications.

[74] Game Freak (2013a). *Pokemon X*. Game [3DS]. Nintendo, Kyoto, Japan.

[75] Game Freak (2013b). *Pokemon Y*. Game [3DS]. Nintendo, Kyoto, Japan.

[76] Games, N. (2017). *Nielsen 360 Gaming Report*.

[77] Godde, B. and Voelcker-Rehage, C. (2017). Cognitive resources necessary for motor control in older adults are reduced by walking and coordination training. *Frontiers in Human Neuroscience*, **11**, 156.

[78] Good Science Studios (2010). *Kinect Adventures!* Game [Xbox 360]. Microsoft Game Studios, Washington, USA.

[79] Grasshopper Manufacture (2008). *No More Heroes*. Game [Wii]. Ubisoft, Rennes, France.

[80] Griptonite Games (2011). *Kung Fu Panda 2*. Game [Xbox 360 + Kinect]. THQ, California, USA.

[81] Hagopian, M. (2010). Motion controls: Not just for casual games.

[82] HAL Laboratory (2002). *Kirby: Nightmare in Dreamland!* Game [Gameboy Advance]. Nintendo, Kyoto, Japan.

[83] HAL Laboratory and Nintendo R&D2 (2001). *Kirby Tilt'N'Tumble*. Game [Gameboy Colour]. Nintendo, Kyoto, Japan.

[84] HAL Laboratory, Inc. (1999). *Super Smash Bros.* Game [N64]. Nintendo, Kyoto, Japan.

[85] HAL Laboratory, Inc. and Intelligent Systems (2001). *Super Smash Bros. Melee*. Game [Gamecube]. Nintendo, Kyoto, Japan.

[86] HAL Laboratory, Inc. and Pax Softnica (1999). *Pokemon Snap!* Game [N64]. Nintendo, Kyoto, Japan.

[87] HAL Laboratory, Inc., Intelligent Systems, S. G. M. and Paon (2008). *Super Smash Bros. Brawl*. Game [Wii]. Nintendo, Kyoto, Japan.

[88] Hamilton, N., Weimar, W., and Luttgens, K. (2012). Kinesiology: Scientific basis of human motion.

[89] Hancock, P. (2012). A quick guide to popular fighting game people and jargon.

[90] Hays, R. T., Jacobs, J. W., Prince, C., and Salas, E. (1992). Flight simulator training effectiveness: A meta-analysis. *Military Psychology*, **4**(2), 63–74. Date revised - 19930401; 20130204; Number of references - 23; Last updated - 2017-10-02; SubjectsTermNotLitGenreText - Aircraft Pilots 266 266PS5 199 199PS5 6783 6783PS5 6342 6342PS5; Flight Simulation 3348 3348PS5 7894 7894PS5; Meta Analysis 5187 5187PS5 5211 5211PS5 8225 8225PS5 388 388PS5; Military Training 5275 5275PS5 8807 8807PS5; 5269 5269PS5 3638 3638PS5 6342 6342PS5.

[91] Houghton, A. (2012). Pc gaming 101: Mouse grip styles.

[92] Hudson Soft (1999). *Mario Party*. Game [N64]. Nintendo, Kyoto, Japan.

[93] Hudson Soft (2000). *Mario Party 2*. Game [N64]. Nintendo, Kyoto, Japan.

[94] Hudson Soft (2001). *Mario Party 3*. Game [N64]. Nintendo, Kyoto, Japan.

[95] Hudson Soft (2002). *Mario Party 4*. Game [Gamecube]. Nintendo, Kyoto, Japan.

[96] Hudson Soft (2003). *Mario Party 5*. Game [Gamecube]. Nintendo, Kyoto, Japan.

[97] Hudson Soft (2007). *Mario Party 8*. Game [Wii]. Nintendo, Kyoto, Japan.

[98] HuniePot (2015). *HuniePop*. Game [PC]. HuniePot, USA.

[99] Ikeda, A., Ito, K., Kuroda, R., Takeda, G., and Urata, M. (2013). Video game system with wireless modular handheld controller. US Patent 8,430,753.

[100] Imangi Studios (2012). *Temple Run*. Game [Android]. Imangi Studios, North Carolina, USA.

[101] Intelligent Systems and Nintendo SPD (2005). *WarioWare: Touched!* Game [DS]. Nintendo, Kyoto, Japan.

[102] Intelligent Systems and Nintendo SPD (2007). *WarioWare: Smooth Moves*. Game [Wii]. Nintendo, Kyoto, Japan.

[103] Intelligent Systems and Nintendo SPD Group No. 1 (2005). *WarioWare: Twisted*. Game [GBA]. Nintendo, Kyoto, Japan.

[104] Jeff (2008). *Rockband Drum Set - Xbox 360 Version*. Gamespot.

[105] Jørgensen, A. H. (2004). Marrying hci/usability and computer games: a preliminary look. In *Proceedings of the third Nordic conference on Human-computer interaction*, pages 393–396. ACM.

[106] Julian Frost and Samuel Baird (2013). *Dumb Ways to Die*. Game [Android]. PopReach, Ontario, Canada.

[107] Julien Thiennot (2018). *Cookie Clicker*. Game [Android]. redBit, Rome, Italy.

[108] Karhulahti, V.-M. (2013). Puzzle is not a game! basic structures of challenge. In *DiGRA Conference*.

[109] Kelly, G. (2017). Pixel vs pixel xl: What's the difference?

[110] Knudsen, E. (2014). An attempt to dissect quick time events.

[111] Konami (1999). *Dance Dance Revolution*. Game [Arcade]. Konami, Tokyo, Japan.

[112] Konami Computer Entertainment Japan (2001). *Metal Gear Solid 2: Sons of Liberty*. Game [PS2]. Konami, Tokyo, Japan.

[113] Laird, J. and VanLent, M. (2001). Human-level ai's killer application: Interactive computer games. *AI magazine*, **22**(2), 15.

[114] Laird, J. E. (2000). An exploration into computer games and computer generated forces. In *Eighth Conference on Computer Generated Forces and Behavior Representation*. Citeseer.

[115] Laird, J. E. (2001). It knows what you're going to do: Adding anticipation to a quakebot. In *Proceedings of the Fifth International Conference on Autonomous Agents*, AGENTS '01, pages 385–392, New York, NY, USA. ACM.

[116] Laird, J. E. (2002). Research in human-level ai using computer games. *Communications of the ACM*, **45**(1), 32–35.

[117] LeJacq, Y. (2013). How fast is fast? some pro gamers make 10 moves per second.

[118] Lesesne, N. (2013). Quick-time events are holding back the development of video game storytelling.

[119] Level-5 (2009). *Professor Layton and the Diabolical Box*. Game [DS]. Nintendo, Kyoto, Japan.

[120] Lim, C.-U., Baumgarten, R., and Colton, S. (2010). *Evolving Behaviour Trees for the Commercial Game DEFCON*, pages 100–110. Springer Berlin Heidelberg, Berlin, Heidelberg.

[121] Lindenberger, U., Marsiske, M., and Baltes, P. B. (2000). Memorizing while walking: increase in dual-task costs from young adulthood to old age. *Psychology and aging*, **15**(3), 417.

[122] Lucasfilm Games (1989). *Indiana Jones and the Last Crusade: The Graphic Adventure*. Game [DOS]. Lucasfilm Games, California, USA.

[123] MacKenzie, I. S. (2012). *Human-computer interaction: An empirical research perspective*. Newnes.

[124] Malcolm, B., Foxe, J., Butler, J., and De Sanctis, P. (2015). The aging brain shows less flexible reallocation of cognitive resources during dual-task walking: A mobile brain/body imaging (mobi) study. *NeuroImage*, **117**, 230.

[125] Malone, T. W. (1980). What makes things fun to learn? heuristics for designing instructional computer games. In *Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems*, pages 162–169. ACM.

[126] McDonald, E. (2017). Newzoo's 2017 report: Insights into the $108.9bn global games market.

[127] McMahon, N., Wyeth, P., and Johnson, D. (2015). From challenges to activities: Categories of play in videogames. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '15, pages 637–642, New York, NY, USA. ACM.

[128] Metro Trains Melbourne (2018). *Dumb Ways to Die 2*. Game [Android]. PopReach, Ontario, Canada.

[129] Miller, P. (2011). Mechanical keyboards: Should you switch?

[130] Murray, J. H. (2006). Toward a cultural theory of gaming: Digital games and the co-evolution of media, mind, and culture. *Popular Communication*, **4**(3), 185–202.

[131] Nakayama, R., Huang, E., and Gary, N. (2013). Gaming controller. US Patent App. 13/842,975.

[132] Natapov, D. and MacKenzie, I. S. (2010). The trackball controller: Improving the analog stick. In *Proceedings of the International Academic Conference on the Future of Game Design and Technology*, Futureplay '10, pages 175–182, New York, NY, USA. ACM.

[133] Natapov, D., Castellucci, S. J., and MacKenzie, I. S. (2009). Iso 9241-9 evaluation of video game controllers. In *Proceedings of Graphics Interface 2009*, GI '09, pages 223–230, Toronto, Ont., Canada, Canada. Canadian Information Processing Society.

[134] Naughty Dog (2013). *The Last of Us*. Game [PS4]. Sony Computer Entertainment, California, USA.

[135] Nd Cube (2012). *Mario Party 9*. Game [Wii]. Nintendo, Kyoto, Japan.

[136] Ninja Theory (2013). *DmC: Devil May Cry*. Game [PC]. Capcom, Osaka, Japan.

[137] Nintendo (2011). Nintendo 3ds operations manual.

[138] Nintendo (2012). Nintendo 3ds xl operations manual.

[139] Nintendo (2018). Product information: Nintendo ds lite.

[140] Nintendo Creative Department (1985). *Super Mario Bros.* Game [NES]. Nintendo, Kyoto, Japan.

[141] Nintendo EAD (1998). *The Legend of Zelda: Ocarina of Time*. Game [N64]. Nintendo, Kyoto, Japan.

[142] Nintendo EAD (2003). *Legend of Zelda: The Wind Waker*. Game [Gamecube]. Nintendo, Kyoto, Japan.

[143] Nintendo EAD (2006). *Wii Sports*. Game [Wii]. Nintendo, Kyoto, Japan.

[144] Nintendo EAD (2011). *The Legend of Zelda: Skyward Sword*. Game [Wii]. Nintendo, Kyoto, Japan.

[145] Nintendo EAD (2013). *The Legend of Zelda: A Link Between Worlds*. Game [3DS]. Nintendo, Kyoto, Japan.

[146] Nintendo EAD (2014). *Mario Kart 8*. Game [WiiU]. Nintendo, Kyoto, Japan.

[147] Nintendo EAD Group No. 2 (2008). *Animal Crossing: City Folk*. Game [Wii]. Nintendo, Kyoto, Japan.

[148] Nintendo EAD Group No. 2 (2009). *Wii Sports Resort*. Game [Wii]. Nintendo, Kyoto, Japan.

[149] Nintendo EAD Group No. 5 (2008). *Wii Fit*. Game [Wii]. Nintendo, Kyoto, Japan.

[150] Nintendo EAD Group No. 5 (2009). *Wii Fit Plus*. Game [Wii]. Nintendo, Kyoto, Japan.

[151] Nintendo EPD (2017a). *Super Mario Odyssey*. Game [Switch]. Nintendo, Kyoto, Japan.

[152] Nintendo EPD (2017b). *The Legend of Zelda: Breath of the Wild*. Game [Switch]. Nintendo, Kyoto, Japan.

[153] Nintendo EPD and Platinum Games (2016). *Star Fox Zero*. Game [WiiU]. Nintendo, Kyoto, Japan.

[154] Nintendo Research and Development 4 (1987). *The Legend of Zelda*. Game [NES]. Nintendo, Kyoto, Japan.

[155] Nintendo SPD Group No. 4 and Nd Cube (2010). *Wii Party*. Game [Wii]. Nintendo, Kyoto, Japan.

[156] Noller, K. and Ingrisano, D. (1984). Cross-sectional study of gross and fine motor development: Birth to 6 years of age. *Physical therapy*, **64**(3), 308–316.

[157] Obsidian Entertainment and South Park Digital Studios (2014). *South Park: The Stick of Truth*. Game [Xbox]. Ubisoft, Rennes, France.

[158] Office Create Corp. (2018). *Cooking Mama: Let's Cook!* Game [Android]. Office Create Corp, Kanagawa, Japan.

[159] Pagoda West Games, H. (2017). *Sonic Mania*. Game [PC]. Sega, Tokyo, Japan.

[160] Pagulayan, R. J., Keeker, K., Wixon, D., Romero, R. L., and Fuller, T. (2002). *User-centered design in games*. CRC Press Boca Raton, FL.

[161] Papegaaij, S., Taube, W., Baudry, S., Otten, E., and Hortobgyi, T. (2014). Aging causes a reorganization of cortical and spinal control of posture. *Frontiers in Aging Neuroscience*, **6**, 28.

[162] Payne, V. G. and Isaacs, L. D. (2012). *Human motor development: a lifespan approach*. McGraw-Hill, 8 edition.

[163] Pearl Abyss (2016). *Black Desert Online*. Game [PC]. Kakao Games, Jeju, South Korea.

[164] Perez, S. (2017). Mobile gaming sessions down 10 percent year-over-year, but revenue climbs.

[165] Perfect Tap Games (2017). *Chicken Scream*. Game [Android]. Perfect Tap Games, Dubai, UAE.

[166] Platinum Games (2010). *Bayonetta*. Game [Xbox 360]. Sega, Tokyo, Japan.

[167] Platinum Games (2014). *Bayonetta*. Game [WiiU]. Nintendo, Kyoto, Japan.

[168] Pocket-lint (2014). Ps vita slim vs ps vita: What's the difference?

[169] PopCap Games (2001). *Bejeweled*. Game [PC]. PopCap Games, Washington, USA.

[170] Quantic Dream (2010). *Heavy Rain*. Game [PS3]. Sony Computer Entertainment, California, USA.

[171] Quinlan, P. T. and Dyson, B. J. (2008). *Cognitive psychology*. Pearson Prentice Hall.

[172] Ramcharitar, A. and Teather, R. J. (2017). A fitts' law evaluation of video game controllers: Thumbstick, touchpad and gyrosensor. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, pages 2860–2866, New York, NY, USA. ACM.

[173] Retro Studios (2010). *Donkey Kong Country: Returns*. Game [Wii]. Nintendo, Kyoto, Japan.

[174] Retro Studios (2014). *Donkey Kong Country: Tropical Freeze*. Game [WiiU]. Nintendo, Kyoto, Japan.

[175] Reverge Labs (2013). *Skullgirls*. Game [PC]. Marvelous, Tokyo, Japan. and Autumn Games, New York, USA.

[176] Rogers, T. (2011). Full Reactive Eyes Entertainment: Incorporating QTEs into gameplay.

[177] Rosado, J., Silva, F., Santos, V., and Lu, Z. (2013). Reproduction of human arm movements using kinect-based motion capture data.

[178] Russell, S., Norvig, P., and Intelligence, A. (1995). A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, **25**, 27.

[179] SCE Santa Monica Studio (2005). *God of War*. Game [PS2]. Sony Computer Entertainment, California, USA.

[180] SCE Santa Monica Studio (2007). *God of War 2*. Game [PS2]. Sony Computer Entertainment, California, USA.

[181] Schell, J. (2014). *The Art of Game Design: A book of lenses*. CRC Press.

[182] Sega Sports R&D (2007). *Mario and Sonic at the Olympic Games*. Game [Wii]. Sega, Tokyo, Japan.

[183] Sega Sports R&D (2008). *Mario and Sonic at the Olympic Games*. Game [DS]. Sega, Tokyo, Japan.

[184] Sega Sports R&D and Arzest (2016a). *Mario and Sonic at the Rio 2016 Olympic Games*. Game [3DS]. Nintendo, Kyoto, Japan.

[185] Sega Sports R&D and Arzest (2016b). *Mario and Sonic at the Rio 2016 Olympic Games*. Game [3DS]. Nintendo, Kyoto, Japan.

[186] Sega Sports R&D and Racjin (2009a). *Mario and Sonic at the London 2012 Olympic Games*. Game [DS]. Sega, Tokyo, Japan.

[187] Sega Sports R&D and Racjin (2009b). *Mario and Sonic at the London 2012 Olympic Games*. Game [Wii]. Sega, Tokyo, Japan.

[188] Sega Sports R&D and Racjin (2011). *Mario and Sonic at the London 2012 Olympic Games*. Game [Wii]. Sega, Tokyo, Japan.

[189] Sega Sports R&D and Racjin (2012). *Mario and Sonic at the London 2012 Olympic Games*. Game [3DS]. Sega, Tokyo, Japan.

[190] Seidler, R. D., Bernard, J. A., Burutolu, T. B., Fling, B. W., Gordon, M. T., Gwin, J. T., Kwak, Y., and Lipps, D. B. (2010). Motor control and aging: links to age-related brain structural, functional, and biochemical effects. *Neuroscience & Biobehavioral Reviews*, **34**(5), 721–733.

[191] Shaffer, D. R. and Kipp, K. (2010). *Developmental psychology: Childhood and adolescence*. Cengage Learning, 8 edition.

[192] Shaul, B. (2016). Report: Average app session length is around 5 minutes.

[193] Simpson, C. (2014). Behavior trees for ai: How they work.

[194] Sleeping Beast Games (2012). *Spaceteam*. Game [Android]. Sleeping Beast Games, Montreal, Canada.

[195] Smith, S. L. (2013). Dualshock 4 vs. xbox one controller - ps4 vs. xbox.

[196] Snap Inc. (2018). *Snapchat Snappables*. Game [Android]. Snap Inc, California, USA.

[197] Sonic Team (1991). *Sonic the Hedgehog*. Game [Genesis]. Sega, Tokyo, Japan.

[198] Sonic Team (2008). *Sonic Unleashed*. Game [Wii]. Sega, Tokyo, Japan.

[199] Spike (2012). *Dragonball Z for Kinect*. Game [Xbox 360 with Kinect]. Bandai Namco Entertainment, Tokyo, Japan.

[200] Square (2002). *Kingdom Hearts*. Game [PS2]. Square Electronic Arts, California, USA.

[201] Square Enix (2006). *Final Fantasy XII*. Game [PS2]. Square Enix, Tokyo, Japan.

[202] Square Enix 1st Production Department (2012). *Kingdom Hearts 3D: Dream Drop Distance*. Game [3DS]. Square Enix, Tokyo, Japan.

[203] Square Enix Business Division 2 (2016). *Final Fantasy XV*. Game [PS4]. Square Enix, Tokyo, Japan.

[204] Square Enix Product Development Division 1 (2006). *Kingdom Hearts 2*. Game [PS2]. Square Enix, Tokyo, Japan.

[205] Square Product and Development Division 1 (2001). *Final Fantasy X*. Game [PS2]. Square Electronic Arts, California, USA.

[206] Steinberg, R. K. and Blume, D. J. (2013). Walk-up acceptability vs. real usability? designing user interfaces for the gaming culture. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, **57**(1), 1470–1474.

[207] Sternberg, R., Sternberg, K., and Mio, J. (2011). *Cognitive Psychology*. Wadsworth/Cengage Learning.

[208] Stevenson, A. (2010). *Oxford Dictionary of English*.

[209] Stobing, C. (2017). If you havent tried a mechanical keyboard yet, youre missing out.

[210] Team Bondi (2011). *L.A. Noire*. Game [Xbox 360]. Rockstar Games, New York, Games.

[211] Tencé, F., Buche, C., De Loor, P., and Marc, O. (2010). The challenge of believability in video games: Definitions, agents models and imitation learning. *arXiv preprint arXiv:1009.0451*.

[212] Terminal Reality (2012). *Kinect Star Wars*. Game [Xbox 360]. Microsoft Game Studios, Washington, USA.

[213] Terrell, R. (2014). The Design of Combos and Chains.

[214] Totilo, S. (2013). Should you buy dance central 2? yes.

[215] Ubisoft Montreal (2007a). *Assassin's Creed*. Game [Xbox 360]. Ubisoft, Rennes, France.

[216] Ubisoft Montreal (2007b). *Assassin's Creed*. Game [PS3]. Ubisoft, Rennes, France.

[217] Ubisoft Montreal (2008). *Shaun White Snowboarding: Road Trip*. Game [Wii]. Ubisoft, Rennes, France.

[218] Ubisoft Montreal (2009a). *Assassin's Creed II*. Game [Xbox 360]. Ubisoft, Rennes, France.

[219] Ubisoft Montreal (2009b). *Assassin's Creed II*. Game [PS3]. Ubisoft, Rennes, France.

[220] Ubisoft Montreal (2010a). *Assassin's Creed Brotherhood*. Game [Xbox 360]. Ubisoft, Rennes, France.

[221] Ubisoft Montreal (2010b). *Assassin's Creed: Brotherhood*. Game [PS3]. Ubisoft, Rennes, France.

[222] Ubisoft Montreal (2010c). *Shaun White Skateboarding*. Game [Wii]. Ubisoft, Rennes, France.

[223] Ubisoft Montreal (2011a). *Assassin's Creed: Revelations*. Game [Xbox 360]. Ubisoft, Rennes, France.

[224] Ubisoft Montreal (2011b). *Assassin's Creed: Revelations*. Game [PS3]. Ubisoft, Rennes, France.

[225] Ubisoft Montreal (2012a). *Assassin's Creed III*. Game [Xbox 360]. Ubisoft, Rennes, France.

[226] Ubisoft Montreal (2012b). *Assassin's Creed III*. Game [Xbox 360]. Ubisoft, Rennes, France.

[227] Ubisoft Montreal (2012c). *Far Cry 3*. Game [Xbox 360]. Ubisoft, Rennes, France.

[228] Ubisoft Montreal (2013a). *Assassin's Creed IV: Black Flag*. Game [Xbox 360]. Ubisoft, Rennes, France.

[229] Ubisoft Montreal (2013b). *Assassin's Creed IV: Black Flag*. Game [PS3]. Ubisoft, Rennes, France.

[230] Ubisoft Montreal (2014a). *Assassin's Creed Unity*. Game [Xbox One]. Ubisoft, Rennes, France.

[231] Ubisoft Montreal (2014b). *Assassin's Creed Unity*. Game [PS4]. Ubisoft, Rennes, France.

[232] Ubisoft Montreal (2017a). *Assassin's Creed Origins*. Game [Xbox One]. Ubisoft, Rennes, France.

[233] Ubisoft Montreal (2017b). *Assassin's Creed Origins*. Game [PS4]. Ubisoft, Rennes, France.

[234] Ubisoft Paris, and Ubisoft Milan (2009). *Just Dance*. Game [Wii]. Ubisoft, Rennes, France.

[235] Ubisoft Quebec (2015a). *Assassin's Creed Syndicate*. Game [PS4]. Ubisoft, Rennes, France.

[236] Ubisoft Quebec (2015b). *Assassin's Creed Syndicate*. Game [Xbox One]. Ubisoft, Rennes, France.

[237] Ubisoft Sofia (2014a). *Assassin's Creed Rogue*. Game [Xbox 360]. Ubisoft, Rennes, France.

[238] Ubisoft Sofia (2014b). *Assassin's Creed Rogue*. Game [PS3]. Ubisoft, Rennes, France.

[239] Van Lent, M., Laird, J., Buckman, J., Hartford, J., Houchard, S., Steinkraus, K., and Tedrake, R. (1999). Intelligent agents in computer games. In *AAAI/IAAI*, pages 929–930.

[240] VanPutte, C., Regan, J., and Russo, A. (2011). *Seeley's Anatomy and Physiology*. McGraw-Hill, 9 edition.

[241] Various Authors (2009a). Console button 'mashing' in gameplay...Hate it!

[242] Various Authors (2009b). If Button Mashing Works Is It Bad Game Design.

[243] Various Authors (2009c). Trouble with solidus cut scene.

[244] Various Authors (2009d). When did you start playing Pokemon?

[245] Various Authors (2010). Poll: Can we stop calling motion controls gimmicky?

[246] Various Authors (2011). Your fps keyboard setup.

[247] Various Authors (2013). [Help Please] The class with the least button mashing.

[248] Various Authors (2014a). *T-Rex Runner*. Game [Chrome Browser]. Google, California, USA.

[249] Various Authors (2014b). The 4 Best COD players on PC vs the 6 best on console.

[250] Various Authors (2014c). When did you start playing?

[251] Various Authors (2015a). Oh damn do I love Bayonetta...But f*** those QuickTime events. So. Much.

[252] Various Authors (2015b). The Cancer of Modern Gaming that is the Quick time Events!

[253] Various Authors (2016a). Controller vs. Keyboard and Mouse.

[254] Various Authors (2016b). Semi Rant: What's Wrong With Button Mashing.

[255] Various Authors (2016c). When did you start playing Pokemon?

[256] Various Authors (2016d). When did you start playing Pokemon?

[257] Various Authors (2017a). In all honesty - are the motion controls a gimmick?

[258] Various Authors (2017b). What is objectively the best keyboard layout for gaming?

[259] Various Authors (2018a). Most common keyboard/mouse inputs for pc games.

[260] Various Authors (2018b). Why does it feel like everyone hate motion control nowadays.

[261] Various Authors (2019). Why people hate skyward sword controls.

[262] Various contributors (2018a). Shield.

[263] Various contributors (2018b). Unblockable attacks.

[264] Vicarious Visions (2017). *Crash Bandicoot N-Sane Trilogy*. Game [PS4]. Activision, California, USA.

[265] Warner Bros. International Enterprises (2018). *Injustice 2*. Game [Android]. Warner Bros. International Enterprises, California, USA.

[266] Watters, C. (2010). Dance central review.

[267] Weinberger, M. (2015). The downfall of kinect: Why microsoft gave up on its most promising product.

[268] Weiten, W. (2007). *Psychology: Themes and variations: Themes and variations*. Cengage Learning.

[269] Yin-Poole, W. (2017). Kinect officially dead.

[270] Zaranek, A., Ramoul, B., Yu, H. F., Yao, Y., and Teather, R. J. (2014). Performance of modern gaming input devices in first-person shooter target acquisition. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, pages 1495–1500, New York, NY, USA. ACM.