UNIVERSITY OF CALGARY

A Visual Interaction Cue Framework from Video Game Environments

for Augmented Reality

by

Kody Dillman

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

GRADUATE PROGRAM IN COMPUTER SCIENCE

CALGARY, ALBERTA

SEPTEMBER, 2018

© Kody Dillman 2018

Abstract

Based on an analysis of 49 popular contemporary video games, I develop a descriptive framework of visual interaction cues in video games. These cues are used to inform players what can be interacted with, where to look, and where to go within the game world. These cues vary along three dimensions: the purpose of the cue, the visual design of the cue, and the circumstances under which the cue is shown. I demonstrate that this framework can also be used to describe interaction cues for augmented reality applications. Beyond this, I show how the framework can be used to generatively derive new design ideas for visual interaction cues in augmented reality experiences and use the framework as a guide to create three novel AR prototypes. Finally, I suggest design considerations for the creation of visual interaction cues for augmented reality and examine how the framework might be employed in future work.

Publications and Statement of Contribution

Some figures and material in this thesis have previously appeared in prior work:

Kody R. Dillman, Terrance Tin Hoi Mok, Anthony Tang, Lora Oehlberg, and Alex Mitchell. 2018. A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Paper 140, 12 pages. DOI: https://doi.org/10.1145/3173574.3173714

Portions of this co-authored publication are reproduced throughout my thesis with permission (Appendix D). As the lead author of this publication, I drove the effort and was primarily responsible for conceptualization of the work since it was an extension of research I had conducted in a course project (described in Chapter Three). I worked closely with one co-author to collect examples of visual interaction cues in games. From there, I organized and led meetings with my collaborators to analyze the example cues we had collected; my co-authors significantly contributed to the development of the framework through these consultations. This was necessary, as the framework was meant to be developed out of discussion, debate, and iteration. After the framework was developed, I outlined and guided the writing of the manuscript. Two collaborators and I contributed to the writing of the first draft of the document. Sections for which I did not write the first draft, I outlined the text to ensure that the person writing the section knew what I wanted to convey. All of the co-authors and I edited the final document, with me taking the final pass to ensure it accurately communicated my ideas.

For this thesis, I was solely responsible for producing the following entirely new content:

- Chapter One:
 - o 1.1 Research Goals
 - o 1.2 Research Context
 - 1.3 Thesis Contributions
 - o 1.4 Overview
- Chapter Two:
 - o 2.1 Visual Cues
 - o 2.3 Augmented Reality
- Chapter Three:
 - o All Content
- Chapter Five:
 - o 5.2 Describing AR Interaction Cues: Example 3
 - o 5.3 Generating AR Scenarios with the Framework: Example 3
- Chapter Six:
 - o All Content
- Chapter Seven:
 - o 7.2 Designing for Asynchronous Collaborative AR Experiences
 - o 7.3 Future Work
 - o 7.4 Summary
- Chapter Eight:
 - o All Content

Further, where portions of the publication were reproduced, I edited and extended them where appropriate.

For consistency and to present the results from my own perspective, I make use of the first-person singular pronoun (*I*, *we*, *etc*.) in this thesis, but I wish to stress the importance of the contributions of my collaborators.

Acknowledgments

Six years ago, I joined the Interactions Lab at the University of Calgary. It was a lifechanging decision. During those years, I developed so many incredible personal and professional relationships, and I have grown so much as a person thanks to these people. I want to begin by acknowledging all of the fantastic people in the iLab. Having been a member of the lab for such a long time, I can't even start naming people individually there are far too many to thank. So, if you are a member of the iLab, I owe you so much.

Many thanks to all of the collaborators on this work. Being able to spend months talking with friends about video games and calling it legitimate research is the coolest thing I've ever done. Terrance, Tony, Lora, Alex — thanks, that was fun!

My supervisor, Tony Tang, could deliver a master class in supporting graduate students. Thanks for your patience, support, and guidance.

Yolande and the "cheer squad," thank you for your love, support, and cheerleading. It kept me going.

I am so grateful for my family. For always loving and supporting me unconditionally.

Finally, I need to thank Santa Claus. For gifting me with a Sega Master System on the "Best Christmas Ever!" in 1989, sparking a love of all things video game related. None of this would have ever happened without that gift.

Table of Contents

Abstract	ii
Publications and Statement of Contribution	iii
Acknowledgments	vi
Table of Contents	vii
List of Tables	x
List of Figures	xi
CHAPTER ONE: INTRODUCTION	1
1.1 Research Goals	5
1.2 Research Context	7
1.3 Thesis Contributions	9
1.4 Overview	9
CHAPTER TWO: BACKGROUND AND RELATED WORK	11
2.1 Visual Cues	12
2.2 Interaction Cues in Video Games	14
2.3 Augmented Reality	16
2.4 Navigation in Augmented Reality	19
2.5 Summary	20
CHAPTER THREE: A NAÏVE APPROACH	22
3.1 Designs	23
3.1.1 Technology	24
3.2 Line Prototype	25
3.3 Off-Screen Guidance (OSG) Prototype	26
3.4 Reflection	29
3.5 Summary	30
CHAPTER FOUR: THE FRAMEWORK	32
4.1 Method	
111 Porcportivo	22

4.2.1 Dimension 1: Task/Purpose	4.2 The Framework	
4.2.2 Dimension 2: Markedness	4.2.1 Dimension 1: Task/Purpose	
4.2.3 Dimension 3: Trigger 52 4.3 Summary 54 CHAPTER FIVE: USING THE FRAMEWORK WITH AR 56 5.1 Mapping Framework Dimensions to AR 56 5.2 Describing AR Interaction Cues 61 5.3 Generating AR Scenarios with the Framework 63 5.4 Summary 66 CHAPTER SIX: THE PROTOTYPES 68 6.1 Design Approach 68 6.1.1 Autobiographical Design as a Method 72 6.2 Bookshelf Prototype (<i>Discover</i>) 73 6.2.1 Task 73 6.2.2 Implementation 74 6.3 Shoulder Surfer Detection Prototype (<i>Look</i>) 75 6.3.3 Cues 77 6.4.0 Task 78 6.4.1 Task 78 6.4.2 Implementation 79 6.4.3 Cues 80 6.4.4 Proximity 81 6.5 Reflection 82 6.6 Summary 84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK 85 7.1 General Design Considerations 86 7.1 Limitations Presented by the Physicality of AR 86	4.2.2 Dimension 2: Markedness	
4.3 Summary 54 CHAPTER FIVE: USING THE FRAMEWORK WITH AR 56 5.1 Mapping Framework Dimensions to AR 56 5.2 Describing AR Interaction Cues 61 5.3 Generating AR Scenarios with the Framework 63 5.4 Summary 66 CHAPTER SIX: THE PROTOTYPES 68 6.1 Design Approach 68 6.1.1 Autobiographical Design as a Method 72 6.2 Bookshelf Prototype (<i>Discover</i>) 73 6.2.1 Task 73 6.2.2 Implementation 74 6.3 Shoulder Surfer Detection Prototype (<i>Look</i>) 75 6.3.1 Task 75 6.3.2 Implementation 76 6.3.3 Cues 77 6.4 Office Mail Delivery Prototype (<i>Go</i>) 78 6.4.1 Task 78 6.4.2 Implementation 79 6.4.3 Cues 80 6.4.4 Proximity 81 6.5 Reflection 82 6.6 Summary 84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK 85 7.1 General Design Considerations 86 7.1.1 Limitations Presented by the Physicality of AR 86 <td>4.2.3 Dimension 3: Trigger</td> <td></td>	4.2.3 Dimension 3: Trigger	
CHAPTER FIVE: USING THE FRAMEWORK WITH AR 56 5.1 Mapping Framework Dimensions to AR 56 5.2 Describing AR Interaction Cues 61 5.3 Generating AR Scenarios with the Framework 63 5.4 Summary 66 CHAPTER SIX: THE PROTOTYPES 68 6.1 Design Approach 68 6.1.1 Autobiographical Design as a Method 72 6.2 Bookshelf Prototype (<i>Discover</i>) 73 6.2.1 Task 73 6.2.2 Implementation 74 6.3 Studer Surfer Detection Prototype (<i>Look</i>) 75 6.3.1 Task 75 6.3.2 Implementation 76 6.3.3 Cues 77 6.4 Office Mail Delivery Prototype (<i>Go</i>) 78 6.4.1 Task 78 6.4.2 Implementation 79 6.4.3 Cues 80 6.4.4 Proximity 81 6.5 Reflection 82 6.6 Summary 84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK 85 7.1 General Design Considerations 86 7.1.1 Limitations Presented by the Physicality of AR 86 7.1.2 Importance of Discover Cues<	4.3 Summary	54
CHAPTER FIVE: USING THE FRAMEWORK WITH AR .56 5.1 Mapping Framework Dimensions to AR .56 5.2 Describing AR Interaction Cues .61 5.3 Generating AR Scenarios with the Framework .63 5.4 Summary .66 CHAPTER SIX: THE PROTOTYPES .68 6.1 Design Approach .68 6.1.1 Autobiographical Design as a Method .72 6.2 Bookshelf Prototype (<i>Discover</i>) .73 6.2.1 Task .73 6.2.2 Implementation .74 6.3 Shoulder Surfer Detection Prototype (<i>Look</i>) .75 6.3.1 Task .75 6.3.2 Limplementation .76 6.3.3 Cues .77 6.4 Office Mail Delivery Prototype (<i>Go</i>) .78 6.4.1 Task .78 6.4.2 Implementation .78 6.4.3 Cues .80 6.4.4 Proximity .81 6.5 Reflection .82 6.6 Summary .84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK .86 7.1.1 Limitations Presented by the Physicality of AR .86 7.1.2 Importance of Discover Cues .86		
5.1 Mapping Framework Dimensions to AR	CHAPTER FIVE: USING THE FRAMEWORK WITH AR	
5.2 Describing AR Interaction Cues 61 5.3 Generating AR Scenarios with the Framework 63 5.4 Summary 66 CHAPTER SIX: THE PROTOTYPES 68 6.1 Design Approach 68 6.1.1 Autobiographical Design as a Method 72 6.2 Bookshelf Prototype (<i>Discover</i>) 73 6.2.1 Task 73 6.2.2 Implementation 74 6.2.3 Cues 74 6.3 Shoulder Surfer Detection Prototype (<i>Look</i>) 75 6.3.1 Task 75 6.3.2 Implementation 76 6.3.3 Cues 77 6.4 Office Mail Delivery Prototype (<i>Go</i>) 78 6.4.1 Task 78 6.4.2 Implementation 79 6.4.3 Cues 80 6.4.4 Proximity 81 6.5 Reflection 82 6.6 Summary 84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK 85 7.1 General Design Considerations 86 7.1.1 Limitations Presented by the Physicality of AR 86 7.1.2 Importance of Discover Cues 86	5.1 Mapping Framework Dimensions to AR	
5.3 Generating AR Scenarios with the Framework	5.2 Describing AR Interaction Cues	61
5.4 Summary	5.3 Generating AR Scenarios with the Framework	63
CHAPTER SIX: THE PROTOTYPES686.1 Design Approach686.1.1 Autobiographical Design as a Method726.2 Bookshelf Prototype (<i>Discover</i>)736.2.1 Task736.2.2 Implementation746.2.3 Cues746.3 Shoulder Surfer Detection Prototype (<i>Look</i>)756.3.1 Task756.3.2 Implementation766.3.3 Cues776.4 Office Mail Delivery Prototype (<i>Go</i>)786.4.1 Task786.4.2 Implementation796.4.3 Cues806.4.4 Proximity816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues86	5.4 Summary	66
CHAPTER SIX: THE PROTOTYPES686.1 Design Approach686.1.1 Autobiographical Design as a Method726.2 Bookshelf Prototype (<i>Discover</i>)736.2.1 Task736.2.2 Implementation746.2.3 Cues746.3 Shoulder Surfer Detection Prototype (<i>Look</i>)756.3.1 Task756.3.2 Implementation766.3.3 Cues776.4 Office Mail Delivery Prototype (<i>Go</i>)786.4.1 Task786.4.2 Implementation796.4.3 Cues806.4.4 Proximity816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues86		(0
6.1 Design Approach686.1.1 Autobiographical Design as a Method726.2 Bookshelf Prototype (<i>Discover</i>)736.2.1 Task736.2.2 Implementation746.2.3 Cues746.3 Shoulder Surfer Detection Prototype (<i>Look</i>)756.3.1 Task756.3.2 Implementation766.3.3 Cues776.4 Office Mail Delivery Prototype (<i>Go</i>)786.4.1 Task786.4.2 Implementation796.4.3 Cues806.4.4 Proximity816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues86	(1 Design Approach	
6.1.1 Autobiographical Design as a Method 72 6.2 Bookshelf Prototype (<i>Discover</i>)	6.1 Design Approach	
6.2 Bookshell Prototype (Discover)	6.1.1 Autobiographical Design as a Method	
6.2.1 Task736.2.2 Implementation746.3.3 Cues.746.3 Shoulder Surfer Detection Prototype (Look)756.3.1 Task756.3.2 Implementation766.3.3 Cues.776.4 Office Mail Delivery Prototype (Go)786.4.1 Task786.4.2 Implementation796.4.3 Cues.806.4.4 Proximity816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues86	6.2 Bookshelf Prototype (<i>Discover</i>)	
6.2.2 Implementation.746.2.3 Cues.746.3 Shoulder Surfer Detection Prototype (Look).756.3.1 Task.756.3.2 Implementation.766.3.3 Cues.776.4 Office Mail Delivery Prototype (Go).786.4.1 Task.786.4.2 Implementation.796.4.3 Cues.806.4.4 Proximity.816.5 Reflection.826.6 Summary.84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK.857.1 General Design Considerations.867.1.1 Limitations Presented by the Physicality of AR.867.1.2 Importance of Discover Cues.86	6.2.1 Task	
6.2.3 Cues	6.2.2 Implementation	
6.3 Shoulder Surfer Detection Prototype (Look)	6.2.3 Cues	
6.3.1 Task756.3.2 Implementation766.3.3 Cues776.4 Office Mail Delivery Prototype (Go)786.4.1 Task786.4.2 Implementation796.4.3 Cues806.4.4 Proximity816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues86	6.3 Shoulder Surfer Detection Prototype (<i>Look</i>)	
6.3.2 Implementation .76 6.3.3 Cues .77 6.4 Office Mail Delivery Prototype (Go) .78 6.4.1 Task .78 6.4.2 Implementation .79 6.4.3 Cues .80 6.4.4 Proximity .81 6.5 Reflection .82 6.6 Summary .84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK .85 7.1 General Design Considerations .86 7.1.1 Limitations Presented by the Physicality of AR .86 7.1.2 Importance of Discover Cues .86	6.3.1 Task	
6.3.3 Cues.776.4 Office Mail Delivery Prototype (Go).786.4.1 Task.786.4.2 Implementation796.4.3 Cues.806.4.4 Proximity.816.5 Reflection826.6 Summary84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK8484CHAPTER SEVEN: DISCUSSION AND FUTURE WORK857.1 General Design Considerations867.1.1 Limitations Presented by the Physicality of AR867.1.2 Importance of Discover Cues	6.3.2 Implementation	
6.4 Office Mail Delivery Prototype (Go)	6.3.3 Cues	
6.4.1 Task.786.4.2 Implementation.796.4.3 Cues.806.4.4 Proximity.816.5 Reflection.826.6 Summary.84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK.84CHAPTER SEVEN: DISCUSSION AND FUTURE WORK.857.1 General Design Considerations.867.1.1 Limitations Presented by the Physicality of AR.867.1.2 Importance of Discover Cues.86	6.4 Office Mail Delivery Prototype (Go)	
6.4.2 Implementation 79 6.4.3 Cues 80 6.4.4 Proximity 81 6.5 Reflection 82 6.6 Summary 84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK 85 7.1 General Design Considerations 86 7.1.1 Limitations Presented by the Physicality of AR 86 7.1.2 Importance of Discover Cues 86	6.4.1 Lask	
6.4.3 Cues	6.4.2 Implementation	
6.4.4 Proximity .81 6.5 Reflection .82 6.6 Summary .84 CHAPTER SEVEN: DISCUSSION AND FUTURE WORK .85 7.1 General Design Considerations .86 7.1.1 Limitations Presented by the Physicality of AR .86 7.1.2 Importance of Discover Cues .86	6.4.3 Cues	
6.5 Reflection	6.4.4 Proximity	
6.6 Summary	6.5 Reflection	
CHAPTER SEVEN: DISCUSSION AND FUTURE WORK	6.6 Summary	84
 7.1 General Design Considerations	CHAPTER SEVEN: DISCUSSION AND FUTURE WORK	
7.1.1 Limitations Presented by the Physicality of AR	7.1 General Design Considerations	
7.1.2 Importance of Discover Cues	7.1.1 Limitations Presented by the Physicality of AR	
	7.1.2 Importance of Discover Cues	

7.1.4 Trade-off: Triggers and Causality	88
7.1.5 Visibility and "Fit" of Cues	88
7.1.6 Impact of Field of View	89
7.2 Designing for Asynchronous Collaborative AR Experiences	90
7.3 Future Work	92
7.3.1 Application to Other Cues and Domains	92
7.3.2 Evaluation	94
7.4 Summary	95

CHAPTER EIGHT: CONCLUSION	96
8.1 Thesis Questions	96
8.2 Thesis Contributions	97
8.3 Closing Remarks	

References	
Ludography	
Appendix A: Game Corpus	
Appendix B: Thematic Analysis Raw Data	
Appendix C: Prototype Sketches	
Appendix D: Co-Author Permission	

List of Tables

36
69
(

List of Figures

Figure 1.1. These Go interaction cues provide navigation guidance along a path. Steep (left) [L16] displays a dotted line in the course; Lowe's In-Store Navigation, a mobile AR app (right) [28], uses a bold yellow line
Figure 1.2. Research Context
Figure 2.1. Evolution of the Windows Start button [53]14
Figure 2.2. The Reality-Virtuality Continuum [38]. Any combination of real and virtual is considered Mixed Reality
Figure 2.3. A quadcopter projects navigation directions onto the sidewalk [33] 17
Figure 2.4. A laser printer maintenance application [18] using a head-mounted see through display shows the user how to replace the printer's paper tray. Shown from external view (left) and from the user's perspective (right)
Figure 2.5. The IKEA Place App for iOS [27] allows customers to preview furniture in their homes. This app uses Apple's ARKit for markerless tracking
Figure 3.1. Early sketches. (clockwise from top left) Line prototype before visiting waypoint. Off-Screen Guidance (OSG) prototype before visiting waypoint. OSG after visiting waypoint. Line after visiting waypoint
Figure 3.2. Lumin [35]. The user is guided through a museum, with a dotted line guiding the user to specific locations in the building
Figure 3.3. Line visualization. Green line points to current waypoint. White line is drawn from waypoint to ground line. Grey line points to future waypoint
Figure 3.4. HUD from Destiny [L6], with the off-screen guidance icon bounded in red. (Annotation added.)
Figure 3.5. OSG visualization (emphasized for clarity). Target 1 is off screen to the left, and target 2 is marked with grey square (top). Target 1 is on screen and marked with blue box, target 2 is off screen (middle). After target 1 has been visited, target 2 is on screen, target 3 is off screen (bottom)

Figure 4.1. Screenshots from some of the games from my sample set: (a) [L11], (b) [L13], (c) [L14], (d) [L3], (e) [L12], (f) [L7], (g) [L15], (h) [L17], (i) [L16], (j) [L4], (k) [L5], (l) [L2]
Figure 4.2. An <i>Emphasized Discover</i> cue. A bag of gold coins is outlined in bright yellow to indicate it can be looted from [L3]
Figure 4.3. An <i>Overlaid Discover</i> cue. This minimap shows a birds-eye-view of nearby objects and points of interest [L4]
Figure 4.4. An <i>Emphasized Look</i> cue. The enemy is highlighted in orange, indicating that he can be killed with a special player attack. This cue is triggered based on the enemy's hit points [L12]
Figure 4.5. An <i>Integrated Look</i> cue. A yellow widget painted below the avatar points at a nearby enemy that is suspicious of the player's actions [L17]
Figure 4.6. Bioshock uses environmental lighting as a <i>Subtle Go</i> cue [L1]
Figure 4.7. A <i>Subtle Discover</i> cue. A part of the wall is coloured with slightly off- saturation to indicate to players that the wall can be manipulated [L11]
Figure 4.8. Dragon Age: Inquisition uses green smoke as a <i>Subtle Go</i> cue [L3] 45
Figure 4.9. An <i>Emphasized Go</i> cue. As the player gets close to the door, it becomes emphasized with a highlight around its edges [L7]
Figure 4.10. An <i>Integrated Discover</i> cue. A label appears to tell the player what actions can be taken on the object [L15]
Figure 4.11. An <i>Integrated Go</i> cue. The added white line and red arches show the player where to go in the race course [L16]
Figure 4.12. An <i>Overlaid Look</i> cue. A red indicator on the aiming reticule shows the player where the avatar is being attacked from [L5]
Figure 4.13. An <i>Overlaid Go</i> cue. A compass at the top of the player's HUD shows "North" in the game, along with specific points of interest [L2]
Figure 4.14. Jetpack Joyride uses a blinking <i>Overlaid Look</i> cue (top) to show where the rocket is about to appear on screen (bottom) [L10]

Figure 4.15. A <i>Subtle Go</i> cue. Light reflects off the sword, pointing to an in-game destination; the player raises their sword to see this light by pressing a button [L14]
Figure 5.1. Variations on an imaginary AR interface that provides a Go cue to the door on the left
Figure 5.2. Reitmayr and Schmalstieg's AR tour guide system [44] 61
Figure 5.3. The Lumin Project [35] gives museum-goers an AR experience for navigating exhibits and learning about artefacts. The AR view of artefacts provides Integrated Discover cues for more information
Figure 5.4. Aero Glass [25] uses a combination of Integrated and Emphasized cues to provide pilots with navigation information while flying, using a head-mounted display
Figure 6.1. Sketch of <i>Emphasized</i> cues highlighting books in a bookshelf
Figure 6.2. Sketch of <i>Subtle</i> cue casting a virtual shadow on a sensitive document 71
Figure 6.3. Sketch of <i>Integrated</i> cues marking desks in an office with flares of light 71
 Figure 6.4. Bookshelf (Discover) cues. (clockwise from top left) <i>Subtle</i> – a sticker is rendered on the spine of the book, <i>Emphasized</i> – an animated particle effect wraps the spine of the book, <i>Integrated</i> – a virtual 3D arrow points to the book, <i>Overlaid</i> – a minimap shows approximate locations of the books
Figure 6.5. Shoulder surfer detection (Look) cues. (clockwise from top left) <i>Subtle</i> – a virtual shadow is cast on the document, <i>Emphasized</i> – the document is given a red border when someone is peeking, <i>Integrated</i> – a virtual light beam is drawn from shoulder surfer to point on the document they are looking at, <i>Overlaid</i> – an icon is drawn at the edge of screen indicating where the shoulder surfer is
Figure 6.6. Office mail delivery (Go) cues. (clockwise from top left) <i>Subtle</i> – a handwritten label is placed on the desk, <i>Emphasized</i> – the drop off location is surrounded by a 3D cuboid, <i>Integrated</i> – a flare marks the next drop-off location, <i>Overlaid</i> – an on-screen compass orients the user toward the next desk

Figure 6.7. An animated line is drawn between waypoints in the Proximity variation	ı
of the Office Mail Delivery prototype. This is an <i>Integrated</i> cue	82

List of Symbols, Abbreviations, and Nomenclature

Symbol	Definition
AR	Augmented Reality
VR	Virtual Reality
FPS	First-Person Shooter
TQ	Thesis Question
ТО	Thesis Objective
TC	Thesis Contribution
FOV	Field of View
HUD	Heads-Up Display
[L(X)]	Ludography Reference

Chapter One: Introduction

Augmented Reality (AR) systems present digital information atop tracked visuals of the physical world [38]. Recent advances in device miniaturization, ubiquitous connectivity, and computing power have helped to commoditize consumer-grade augmented reality technologies, enabling a range of applications that were previously only possible in specially-designed research environments [12]. Many AR scenarios, including tour/museum guides, remote assistance, and emergency response involve providing the user with visual guidance about what to pay attention to in the visual space, or where to go in the physical space (e.g. [19, 29, 44]). The problem is that designers do not yet have a common visual language for constructing these visual guidance cues; consequently, current approaches tend to be idiosyncratic one-off designs. My interest is in developing recommendations for designers looking to provide interaction and navigational assistance in AR systems.

I draw inspiration from a related domain that has, to some extent, already developed this visual language: video games. Video game designers make use of visual interaction cues to guide players around virtual spaces, and these cues can be styled in many different ways. For instance, some games use large 3D arrows to point to off-screen destinations or targets; similarly, others use subtle variations in colour or lighting to guide a player's



Figure 1.1. These Go interaction cues provide navigation guidance along a path. Steep (left) [L16] displays a dotted line in the course; Lowe's In-Store Navigation, a mobile AR app (right) [28], uses a bold yellow line.

attention in the scene (i.e. to suggest the player look at one spot or another). In each of these cases, the purpose of the cue is different: in the first case, it could be to tell a player where to go to progress in the game, while in the second case, it might be to help the player find a hidden treasure. Designers do not yet have a formal vocabulary for describing and understanding these interaction cues broadly. To address this problem, I conducted an exploratory study of 49 video games¹ to understand how visual interaction cues are used to communicate information about the game world to players. My analysis suggests that games provide these cues to support three distinct tasks or purposes, encouraging the player to: *Discover* interactive artefacts, objects, or areas in the scene; *Look* at artefacts, objects, or areas in the scene that require timely action or reaction; and *Go* to important spatial locations in the virtual game world. These interaction cues vary in two other dimensions: the markedness of the cue (i.e. the extent to which the cues are a part of the game world: *Subtle, Emphasized, Integrated, Overlaid*), as well as how these cues are triggered (e.g. Player, Context, Other/Agent, or *Persistent*). Figure 1.1 illustrates the use of Go cues in the snowboarding game Steep [L16] (left) and Lowe's augmented reality In-Store Navigation app [28]. In both cases, these cues guide the player/user where to go in the environment.

I use this understanding as the basis of a framework that allows me to describe and compare the different kinds of interaction cues in AR systems. Furthermore, the framework is generative—that is, it can be used to inspire new designs for AR to guide a user (though there is no reason this framework cannot be used in other domains that

¹ See Appendix A for complete corpus of games.

utilize visual cues to guide users' attention). This thesis makes use of autobiographical design [39] as a method for the creation and evaluation of the framework. While initially seen as a less rigorous method in HCI research, researchers have recently started to understand just how valuable this type of approach can be. I discuss this method in detail in Chapter Four.

To learn more about how AR technology could support the task of guiding people through spaces, I started to develop AR prototypes to compare different visualizations, measuring task completion time and distance travelled in pilot studies. While developing these visualizations, I realized that I had no idea where to start and found myself defaulting to conventions I had seen employed in video game design. This resulted in a revelation for me: video games have expertly guided players through spaces for decades, and they must be able to provide some insight into how to best design for AR guidance.

The creation of this framework addresses the call by Billinghurst et al. [12] to develop new interaction vocabularies for AR, rather than merely re-using conventions from other domains. It is important at this stage of AR work to learn generally how to create visual cues for augmented reality: the tasks they must support, how they are styled, and how they are trigged in the application. This will provide a foundation for research that will formally evaluate cues and provide insight into which cues are most effective and appropriate for given tasks.

I.I Research Goals

The primary research problem this thesis aims to address is:

How should we design visual cues for augmented reality guidance?

Augmented reality applications are growing in popularity, and designers have powerful new tools to create meaningful AR experiences for consumers. Little work, however, has explored how to design the visual cues for these applications specifically. Rather than start with a clean slate when trying to develop new conventions for augmented reality guidance, game design techniques could provide a firm foundation for beginning to develop methods for use in the design of AR applications.

Thesis Questions

The main research problem can be further divided into the following, more specific, thesis questions:

Thesis Question 1 (TQ1): What are the ways video games make use of visual cues to guide players through virtual worlds? This question is necessarily self-contained and makes no considerations for augmented reality specifically. Video game designers leverage the fact that players have developed an understanding of specific visual cues in games and use that shared understanding of cues to guide players effectively through virtual worlds. These conventions have been developed over decades of trial and error. While games make use of multiple types of cues (e.g. visual, auditory, haptic), I focus exclusively on visual cues in this thesis.

Thesis Question 2 (TQ2): How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?

The insights gained while exploring TQ1 lead into this question. Not all cues and methods used in video games will translate well to other domains, and AR designers face specific constraints that game designers need not consider. I answer this question by redesigning existing AR interaction cues and by creating entirely new prototypes, reflecting on the design process, and providing recommendations for other designers.

Thesis Objectives

I aim to answer the thesis questions by meeting the following objectives:

Thesis Objective 1 (TO1): Perform a thorough study of visual interaction cues used in games (*TQ1*).

Thesis Objective 2 (TO2): Develop a framework to describe the types of visual interaction cues used in games (*TQ1*).

Thesis Objective 3 (TO3): Articulate how the framework cues can be ported to the design of AR applications, given the current technological considerations and limitations of working within a real-world setting (*TQ2*).

Thesis Objective 4 (TO4): Create augmented reality prototypes as a proof of concept, using the visual interaction cue framework as a guide (*TQ2*).

I.2 Research Context

This thesis is firmly situated within the Human-Computer Interaction (HCI) research space. This field deals primarily with how people work with technology, given their abilities (and limitations) regarding things like perception, cognition, and other human factors. Within that space, this work further focuses in on Visualization. There are many ways video games guide players and their attention, but I strictly focus on the visual channel. The two main parts of this thesis (TQ1 and TQ2) can be split further into two research areas within HCI and Visualization: Video Game Design and Augmented Reality (Figure 1.2). Video Game Design is a sub-discipline of Game Design, and a



Figure 1.2. Research Context.

significant amount of Video Game Design is positioned in the HCI space; all of those human factors we are concerned with in HCI are important to consider when designing video games. Augmented Reality exists on what is known as the Reality-Virtuality continuum [38], the defining factoring being that AR experiences principally utilize the user's view of the real world, overlaying some information in the form of virtual elements.

Thesis Question One deals explicitly with Video Game Design, and how visual cues are utilized within games. Thesis Question Two deals with how visual cues can be designed for AR applications, using game design as inspiration.

I.3 Thesis Contributions

- *Thesis Contribution 1 (TC1)*: Creation of a framework that describes how visual cues are used in game design to guide players through virtual spaces. This contribution is outlined in Chapter Four.
- *Thesis Contribution 2 (TC2)*: Demonstration of how the visual interaction cue framework can be used to describe and design new AR tools for spatial guidance in the real world. This contribution is discussed in Chapters Five and Six.
- *Thesis Contribution 3 (TC3):* The creation of augmented reality prototypes imagining new interactions for AR guidance in the real world. This is addressed in Chapter Six.

I.4 Overview

The thesis is structured as follows:

Chapter Two introduces the concepts of augmented reality and semiotic theory, and provides an overview of related work concerning navigation using augmented reality systems and interaction cues in video games.

Chapter Three recounts the design of initial prototypes for AR guidance, which is a naïve approach taken before the development of the interaction cue framework.

Chapter Four describes the visual interaction cue framework, providing examples of cues from some of the games studied in the creation of the framework. This chapter also describes the methods used throughout this process.

Chapter Five outlines how the framework can be applied to augmented reality, by both describing AR cues in current applications and imagining improvements to those applications by making use of the framework.

Chapter Six describes the development of novel prototypes for simple guidance tasks, providing a reflection on designing with the framework as a generative tool.

Chapter Seven is a discussion dealing with design considerations for the development of AR tools for guidance and provides thoughts about future work in this space.

Chapter Eight is a conclusion to this thesis, returning to the research goals and thesis contributions.

Chapter Two: Background and Related Work

In this chapter, I discuss the necessary background to develop a foundation for understanding the basic concepts upon which my work builds. I also give an overview of work related to augmented reality for guidance and interaction cues in video games.

I begin by providing background information on semiotics (the study of signs), giving working definitions for essential concepts, with an emphasis on visual cues and how they support us in our understanding of the world around us.

After discussing signs and visual cues generally, I give an overview of research areas that explore how video game designers style and utilize interaction cues in video games. I discuss how frameworks from the games research literature help to address some gaps in the AR space (specifically, the issue of visual design).

I then discuss augmented reality in general with the goal of merely providing background into AR research and technology. There are a number of in-depth surveys that describe the history and current state of the art in AR (e.g. [9, 12]), but I aim to discuss background and examples to describe the technology as well as its applications. Finally, I outline related work concerned with designing augmented reality applications for navigation, which positions my work in the research space and provides a foundation upon which to build.

This chapter sets up the subsequent chapters and supports them in addressing the main research problem and the thesis questions presented in Chapter One. It also provides a basis for discussions of augmented reality and visual interaction cues in the later chapters.

2.1 Visual Cues

Whether they are apparent or not, signs are abundant in our world and are vital to human communication. These can be blatantly evident, such as a road sign or a sign in a shop window, but can also be as subtle as someone gesturing while they are speaking. Semiotics, the study of signs, considers a sign to be anything that can "stand for" another thing [15]. According to Saussure, one of the founders of the field of semiotics, a sign consists of a *signifier* and a *signified*; the representation of the sign is the *signifier*, and the concept to which it refers is the *signified* [47]. While signs may take many forms, I focus strictly on visual signs in this work, which I define as some kind of visual marking, decoration, or symbol (in my case, with the purpose of giving a person information about their environment). So, when I refer to "visual cues," I am referring to the *signifier*. These visual cues can take many forms in our physical world. Some examples:

- A line drawn along a hospital floor guiding a patient to different departments. *Signifier*: Painted line on the ground. *Signified*: The concept that following this line will get you where you need to go.
- Signal lights on a car warning other motorists of a driver's intent to change lanes.
 Signifier: Flashing light on the side of the vehicle. Signified: The plan of the driver to move in the direction indicated by the flashing light.
- This **bolded text right here** is even a visual cue. *Signifier*: Text that is darker than the rest. *Signified*: The concept that this idea should be given more attention than the others in the text.

Signs evolve over time [47]. The meanings of individual signs come from social conventions, so as conventions change, so do common signs. One example related to human-computer interaction is the styling of the Windows Start button [53]. This example demonstrates the development of the convention of having a user "start" by clicking the button in the bottom left corner of the screen. Initially, when users were becoming familiar with the concept of a taskbar and start menu, it was necessary to very clearly label the button with text, telling the user "this is where you begin." Over time,



Figure 2.1. Evolution of the Windows Start button [53].

this button placement and meaning became convention, so the visual design of the button could become more abstract (Figure 2.1).

2.2 Interaction Cues in Video Games

Just as signs can help us navigate our physical world, visual cues in video games help players make sense of virtual worlds. Large virtual worlds are notably difficult to navigate, a problem addressed by Darken and Sibert [17] who applied traditional wayfinding techniques from architecture (e.g. [6]) to wayfinding in virtual spaces. Game design research has attempted to understand how cues are utilized in video games and has helped us to comprehend how cues affect the game experience. Bardzell [11] focuses on the design and use of *interaction cues* across a wide range of video games, which she defines as signs that provide information to players so that they are aware of their present interactive possibilities. When game designers add visual elements into games (e.g. objects, UI elements, or other types of overlays), they need to ensure the elements are usable [40]: visibility of affordances, clear conceptual models, natural mappings, and feedback for actions with these elements. As such, the principal challenge is to design cues that clearly signal their availability for action to the player (i.e. for interaction), and that the result of such action is clear. Thus, Bardzell was concerned with two properties of cues: their markedness (i.e. do they "stick out" visually), and diegesis (i.e. are they visible to the avatar in the game world). Other researchers have explored how diegetic elements influence game experience for players. Studies have evaluated how diegetic elements affect immersion [26, 45], as well as player performance [42]. Generally, the work points to increased feelings of immersion as non-diegetic HUD elements are removed (e.g. [26]).

Jørgensen [32] challenges the utility of "diegetic" as a descriptive property. In her work exploring music and sound in games, she argues that because the audience of a game is not passive, but rather participates in (i.e. acts on) the game world, distinguishing diegetic and non-diegetic forms of some kinds of sounds is challenging. Game sounds cue the user's understanding of the environment (e.g. as the player moves the avatar through a forest, the music suddenly changes to "enemy" music, signalling that the combat is about to begin). Thus, even while the music is styled to the universe and is non-diegetic because the avatar does not hear it, it ultimately affects the narrative that the avatar experiences, blurring the line between diegetic and non-diegetic elements. *Dead Space 3* [L19] is a game example that blurs this line, where traditional HUD elements like health are part of the avatar's suit. Similarly, *Horizon Zero Dawn* [L9] uses the in-game

mechanic of "augmented reality" goggles to see enemy movement paths. In both these cases, the cues are technically diegetic, but they are blurry lines. Thus, the diegetic distinction is not always useful: the consequence of the cue is the same from a player's perspective, regardless of its diegetic status. Instead, Jørgensen argues that the representation of the cue is more important in determining whether the user notices a cue (i.e. its markedness), and what to do with the cue (mental model).

2.3 Augmented Reality

Milgram et al. [38] imagines AR as an area on a continuum of mixed-reality experiences (Figure 2.2), with total reality at one end and total virtuality at the other². Any combining of virtual and real constitutes mixed reality, with augmented reality being more real than



Figure 2.2. The Reality-Virtuality Continuum [38]. Any combination of real and virtual is considered Mixed Reality.

² Milgram et al.'s continuum is just one model for AR. Others are more complex, but this simple model is suitable for the purposes of this thesis.

virtual; that is, if the user's view includes more elements of the real world than the virtual world, they are experiencing augmented reality. Definitions for augmented reality are necessarily loose, limiting not themselves to specific technologies, but AR must (according to Azuma): combine real and virtual elements, be interactive in real time, and be registered in 3D [9]. While AR



Figure 2.3. A quadcopter projects navigation directions onto the sidewalk [33].

systems can be built upon a number of different technologies for tracking (e.g. magnetic tracking, GPS, inertial tracking) [12], in this work I focus primarily on those that use fiducials (e.g. Vuforia [52]) for marker tracking, as well as mobile technology for markerless tracking (for instance, computer vision, accelerometers and gyroscopes, and depth cameras). These can be phone- or tablet-based (e.g. Apple iOS devices[3], Google Tango [21, 43]), or head-mounted (e.g. Hololens [37]).

Just as many technologies can be used for tracking, the same is true for displaying augmentations to the user. Some examples include: optical see-through displays, videobased displays, and projection-based displays [12]. For see-through AR technologies, the augmentations are rendered on a transparent overlay of the user's view of the world (e.g. Figure 2.4 [18]). For video-based AR, video screens are used to render a combined video image of both the world and the augmentations; mobile phones and tablets are video-based AR technologies (e.g. Figure 2.5 [27]). Finally, for projection-based experiences, the image is projected directly on the world, without a display of any kind between the user and world (e.g. Figure 2.3 [33]).

While AR has long been an area of interest in specific commercial and research domains [9, 12], for instance, manufacturing and repair [18] (Figure 2.4³), recent advances in mobile technology now allow for rich AR experiences for consumers. Virtually anyone with a



Figure 2.4. A laser printer maintenance application [18] using a head-mounted see through display shows the user how to replace the printer's paper tray. Shown from external view (left) and from the user's perspective (right).

³ This image is reproduced from [9].

smartphone can now experience markerless augmented reality thanks to toolkits such as Apple's ARKit [5] and Google's ARCore [4]. For instance, IKEA customers can now preview how furniture will fit in their homes by using the company's AR application,



Figure 2.5. The IKEA Place App for iOS [27] allows customers to preview furniture in their homes. This app uses Apple's ARKit for markerless tracking.

IKEA Place, for iOS devices [27]. New consumer devices such as Hololens [37] and Magic Leap [36] will soon provide consumers with the opportunity to experience head-mounted optical see-through AR.

2.4 Navigation in Augmented Reality

As AR overlays virtual elements on a user's view of the real world, navigation is a natural application for the technology. Grasset et al. [23] provide a rich survey describing navigation techniques in AR across several decades of augmented reality work. The principal distinction the authors make is whether AR is a primary source of spatial information (e.g. labelling objects in the user's environment with meaningful annotations), or whether it is a secondary source (e.g. viewing a virtual map of an external space, tracked with an arbitrary AR marker). My interest is in primary experiences, where the use of the AR display is to provide guidance information. Grasset et al. [23] distinguish between two types of navigation information: exploratory navigation, where the goal is to provide information about an environment, and goal-oriented navigation, where wayfinding instructions are visualized in the environment. One challenge is to make these visualizations easy to understand—i.e. how they are grounded/related to the surrounding world. Some work has explored visualizing a ground plane [34], while others have studied dealing with visual cues that need to be occluded in various ways (e.g. [7, 8, 10]). Other researchers have tried visual blending [46].

While this is a useful starting point for understanding previous approaches to designing intelligible cues in AR, I want to consider the specific visual and interaction language used to "paint" these interaction cues. Thus, I am interested not only in the visual intelligibility of the cues, but also the visual language of these cues for someone who is either designing or consuming the interaction cue.

2.5 Summary

This chapter outlined necessary background information on both augmented reality and visual cues. It also delivered an overview of work related to how interaction cues are

utilized in games to guide players as well as how augmented reality can be used for navigation in the real world.

The framework I present in Chapter Four (answering *TQ1*) ultimately builds on the vocabulary introduced by Grasset et al. [23], Bardzell [11], and Jørgensen [31, 32]. The fundamental departure from prior work is a more nuanced articulation of points along dimensions of purpose, markedness, and trigger. This articulation aligns nicely with designers' intentions in AR, and so I argue for its use as a generative framework (an argument I develop and present in Chapters Five, Six, and Seven).
Chapter Three: A Naïve Approach

My work in learning about how to address the main research problem (*How should we design visual cues for augmented reality guidance?*) began with taking an unstructured, naïve kind of approach. This chapter discusses my design and creation of two AR prototypes for guidance, which were used in a pilot study to compare the efficiency of the designs to guide users through indoor spaces. The pilot study itself is not addressed, but the main concern here is my experience with the naïve approach for design. This is in contrast to Chapter Six, which employs a more formal approach to the problem, informed by the answering of *TQ1* (*What are the ways video games make use of visual cues to guide players through virtual worlds?*).

This chapter does not specifically address any of the thesis questions or contributions, but instead lays the foundation for the development of those questions. It was only by working through this process that I discovered what the main issues were, allowing me to develop a strategy for challenging the problem more effectively.



Figure 3.1. Early sketches. (clockwise from top left) Line prototype before visiting waypoint. Off-Screen Guidance (OSG) prototype before visiting waypoint. OSG after visiting waypoint. Line after visiting waypoint.

3.1 Designs

The purpose of the prototypes is to guide a user through an office space, helping the user to move to different waypoints in a specific order. One visualization draws a line between waypoints for the user to follow, and the other uses only targets but will reorient the user if the waypoint is off-screen. Both of the prototypes provide some kind of indication of where the following waypoint will appear after the current target is visited. Both prototypes were initially designed to show distance to the next waypoint as a label on the target (Figure 3.1), but this feature was not implemented in the final prototypes as a way to keep the interface as simple as possible.

3.1.1 Technology

The mobile device used in this study was the Google Tango tablet [21, 43]. This is a 7-inch tablet that includes special hardware to localize itself in 3D space in indoor environments. The prototypes that run on the Tango were created in Unity [49] and use the Tango SDK as well as an off-screen indicator asset [41] from the Unity Asset Store to create the Off-Screen Guidance application. I developed a separate authoring application to register the positions and orders of targets in the space, again using Unity and Tango.



Figure 3.2. Lumin [35]. The user is guided through a museum, with a dotted line guiding the user to specific locations in the building.

3.2 Line Prototype

The Line prototype was inspired by current navigation systems such as Google Maps [22] that use a line to guide the user to a point of interest. One example of such a system using Google Tango is Lumin [35], which was created to guide museum visitors through exhibits using an augmented reality line with AR icons marking the points of interest (Figure 3.2).

The prototype presents the user with a line drawn at approximately ground-level between the previous waypoint and the current waypoint (Figure 3.3), and the line is coloured using a gradient between red (start) and green (end). The target has a white "stake" line drawn from the target itself, down to the end of the red-green line. At the



Figure 3.3. Line visualization. Green line points to current waypoint. White line is drawn from waypoint to ground line. Grey line points to future waypoint.

end of the red-green line, a new line begins that draws a grey path to the subsequent waypoint, letting the user anticipate which way they will need to move after they visit the current waypoint. This prototype was created to be visually bold, ensuring that the cue stands out among the other elements in the environment.

3.3 Off-Screen Guidance (OSG) Prototype

First-person shooter (FPS) games address the issue of guiding a person in 3D space, so lessons learned from the design of these interfaces can serve as a starting point for other navigation applications. The Off-Screen Guidance (OSG) prototype was inspired by the heads-up displays (HUDs) used in many FPS games, such as Destiny [L6] (Figure 3.4).



Figure 3.4. HUD from Destiny [L6], with the off-screen guidance icon bounded in red. (Annotation added.)

Specifically, when the point of interest is off the screen, there is an indicator at the edge of the display to cue the user to the position of the next waypoint.

In the final OSG prototype, the user sees an empty square over the current target if the target is in the tablet's view (Figure 3.5). If the target is not within view, an icon (filled square) is shown at the edge of the screen as an indication of how the user should turn to get the target onto the screen. The current target shows as a green square icon at the edge of the screen, and the subsequent target shows as a filled grey square at the edge of the screen, or an empty grey square if the subsequent target would be in view of the tablet screen.



Figure 3.5. OSG visualization (emphasized for clarity). Target 1 is off screen to the left, and target 2 is marked with grey square (top). Target 1 is on screen and marked with blue box, target 2 is off screen (middle). After target 1 has been visited, target 2 is on screen, target 3 is off screen (bottom).

3.4 Reflection

There are obvious benefits and drawbacks to taking a naïve approach to the design of new technologies. Some designers may enjoy the freedom of diving into a new technology, strictly relying on intuition to guide their work, and may be successful in doing so. As a more novice designer exploring a new space, I find that a formal approach better suits my style, allowing me to follow some kind of guiding questions or principles in my work.

The goal of the initial pilot study was to compare designs to discover which types of visual cues would be the most efficient in guiding users through spaces. The primary concern with the naïve approach is that it left me with a feeling that I was not convinced that these visualizations were even worth testing against each other. I needed to answer a question for myself: What types of solutions are even possible? With that answer, I could more confidently continue this research and explore a wider range of possibilities. As the discussion of semiotics in Chapter Two echoes, my instinct in developing the new guidance cues was to create things that stand out in a bold way. I found myself saying things like "I could use a big bold 3D arrow", or "draw a bright line". These are the types of gaudy solutions that seem to be common in the early stages of design in a new space, but are usually not the most elegant solutions, as time often reveals (as in the example of the Windows Start button).

While the naïve approach failed to produce much variation or novelty in designs, it did lead me to the realization that video games could inform the AR designs. It may, however, be that these solutions are only well-suited to current gamers who have come to understand the conventional cues that games use as a kind of language in which they are fluent. We may also find that these conventions are understood by non-gamers, but only after a period of training. While these issues are left for future work, I believe that video game designs can provide a solid foundation for learning about the range of possible solutions in developing designs for guidance. It is possible to take advantage of all the knowledge game designers have mined in guiding players through game worlds.

3.5 Summary

In this chapter, I presented two prototypes for AR guidance, developed as part of a pilot study to compare the efficiency of different styles of designs. This approach was naïve; I started developing without much more guidance than the notably vague question of how to best develop AR for guidance. This naïve method produced two prototypes: Line and Off-Screen Guidance (OSG). The OSG prototype was heavily inspired by FPS games, leading me to focus on the question of how video games guide players in virtual worlds, and prompting me to formulate the first thesis question: *TQ1 (What are the ways video games make use of visual cues to guide players through virtual worlds?)*. The next chapter goes

deep into the process of answering that question, providing a solid foundation for the rest of this thesis.

Chapter Four: The Framework

The previous chapter discussed a naïve first attempt at addressing the problem of how to design augmented reality applications to guide users through physical spaces. There were no formal guiding principles; I strictly relied on design intuition to start. Having noticed an inclination to draw from game design conventions using the naïve approach, I decided to take a step back to learn about how video games supported the guidance of players in virtual spaces, which would inform my designs for AR guidance techniques using visual interaction cues. To that end, I developed the following framework to describe visual interaction cues used in video games.

This development of this framework addresses *TQ1* (*What are the ways video games make use of visual cues to guide players through virtual worlds?*). This is made possible by completing objectives *TO1* (*perform a thorough study of visual interaction cues used in games*) and *TO2* (*develop a framework to describe the types of visual interaction cues used in games*).

This chapter is split into two main sections. The first, *Method*, discusses how I created the framework, while the second, *The Framework*, presents the final version of the framework. This framework is then used in depth in the subsequent two chapters to demonstrate how

it can be used to both describe current interaction cues used in AR applications and how it can be used as a generative tool for the creation of new AR designs.

4.1 Method

4.1.1 Perspective

While my focus on interaction cues comes from my interest in creating suitable visual cues for augmented reality as a designer, I tackle this question as an experienced gamer who plays games on both dedicated gaming platforms (Xbox, PlayStation, NES, etc.) and general-purpose computers. Apart from actually playing games, I actively consume news and media related to gaming and game design. The team I worked with in developing the framework were also avid gamers, with one member even having previously worked with a video game company. This experience, combined with the completion of a university-level course in video game history, provided a wealth of "insider knowledge" of the domain from which I am drawing my insights.

4.1.2 Game Selection

I selected 49 contemporary video games⁴. My goal was to collect interesting examples with high variance in how cues were designed and used. I used a purposive selection technique, where I selected games that use interaction cues to guide players. While I

⁴ See Appendix A for complete corpus of games.

began by identifying games I was familiar with, I was conscious of my personal preferences for game genres and sought to mitigate the effect of the potential bias. To this end, I expanded the set of games outside of my own experiences through recommendations from colleagues (with whom I discussed my research goals). Among these recommendations, I was additionally selective: if a game's interaction cues were already represented in the sample, I did not include the game. The sample I report on represents a mix of first-person shooter games, third-person adventure games, 3D and 2D platformers, driving games, and puzzle games. My sample is not intended to be exhaustive; however, it is representative of the wide range of experiences that contemporary game players enjoy.

4.1.3 Method and Analysis

I reflected on the gameplay experience for each game, considering how in-game UI and structural elements in the game supported a player's experience in navigating the game world. For games that I had experience with, I replayed some games; for games that I did not have personal experience with, I watched online "walkthrough" gameplay videos. For this latter set of games, I watched the game until I felt I had a clear sense of a player's in-game experience.

I was sensitive to games where the player/avatar navigated a game world larger than the space that could fill the screen (i.e. where the screen acts as a viewport into the world).

Within this context, I focused my attention on aspects of the game experience that could help the player, not specifically from the perspective of completing game objectives, but rather in terms of guiding a player's attention in the game world. I paid attention to both overt aspects of the UI, as well as understated elements. I reasoned that regardless of whether a cue worked well, they were explicitly created elements (from the perspective of the game designer) and that designers could learn from both successes and failures.

For each game, I identified visual elements that fulfilled my criteria for potentially helping a player navigate the game world. I collected screenshots of each of these, describing how a player would use them, what they looked like, and the context in which they appeared. I used a thematic analysis process [13] with my team, where we iteratively grouped, labelled, discussed, and re-labelled categories and axes that described and explained the various cues⁵. My colleagues and I iteratively refined these categories, labels, and axes as we added more games into the sample until the framework was found to be relatively stable.

It is important to note that, while video games use multiple techniques to guide players and their attention (such as auditory cues or haptic feedback), the focus in this work is

⁵ See Appendix B for an example of a spreadsheet used during the thematic analysis process.

strictly on visual cues. Other techniques are left for future work and are discussed in

Chapter Seven.

D1: Purpose	Discover	Informs the player of objects or points of interest in the environment.
		Figure 4.1.a: A part of the wall is coloured with slightly off-saturation to indicate to players that the wall can be manipulated [L11].
	Look	Informs the player where to put their visual attention in a timely manner.
		Figure 4.1.k: An overlaid red indicator on the aiming reticule shows the player where the avatar is being attacked from [L5].
	Go	Provides navigational assistance through environment.
		Figure 4.1.i: The added white line and red arches show the player where to go in the race course [L16].
D 3: Trigger	Subtle	The cue blends into the environment seamlessly.
		Figure 4.1.b: To indicate that the player is being shot at, the enemy's gun is painted with a lit flare [L13].
	Emphasized	An object or surface in the environment is highlighted.
		Figure 4.1.d: A bag of gold coins is outlined in bright yellow to indicate it can be looted from [L3].
	Integrated	A "virtual" object is added into the environment, tracked by the viewport.
		Figure 4.1.h: A yellow widget painted below the avatar points at a nearby enemy that is suspicious of the player's actions [L17].
	Overlaid	Virtual objects are added atop the viewport, and do not track the view.
		Figure 4.1.I: A compass at the top of the player's HUD shows "North" in the game, along with specific points of interest [L2].
	Player	The cue is activated by an explicit player action.
		Figure 4.1.c: The yellow beam of light emitted by the sword points to an in-game destination; the player raises their sword to see
		this light by pressing a button [L14].
	Context	The cue is activated by some implicit player action
		Figure 4.1.f: As the player gets close to the door, it becomes emphasized with a highlight around its edges [L7].
	Other/Agen	t The cue is activated by some other agent (system or other player)
		Figure 4.1.e: The enemy is highlighted in orange, indicating that he can be killed with a special player attack. This cue is triggered
		based on the enemy's hit points [L12].
	Persistent	The cue is always visible.
		Figure 4.1.j: This minimap shows a birds-eye-view of nearby objects and points of interest, and is visible on the player's HUD at
		all times [L4].

Table 4.1. Summary of the visual interaction cue framework. These dimensions are illustrated by in-game screenshots in Figure 4.1.

4.2 The Framework

My framework describes the interaction cues I found in the sample of video games⁶ along three dimensions: *task, markedness,* and *trigger* source. Described along these dimensions, interaction cues can be understood in terms of the purpose of the cue, the visual design of the cue, and the circumstances under which the cue is shown. Table 4.1 summarizes the dimensions of the framework, relating these to gameplay screenshots in Figure 4.1.

4.2.1 Dimension 1: Task/Purpose

I observed in the sample that interaction cues are purposely designed and used to help a player in one of three different ways: to *Discover* interactable objects, to *Look* at something in the environment, or to *Go* to a location in the environment.

Discover. *Discover* cues show the player what can be interacted with: what objects are interactable, what areas or spaces in the game world can be moved into, and so forth. Game worlds can be made up of thousands of objects (e.g. items, props, locations), yet, only a handful of these are designed to be interacted with. The Gibsonian [20] affordances of the environment may suggest more things that can be interacted with than the game

⁶ See Appendix A for complete corpus of games.



Figure 4.1. Screenshots from some of the games from my sample set: (a) [L11], (b) [L13], (c) [L14], (d) [L3], (e) [L12], (f) [L7], (g) [L15], (h) [L17], (i) [L16], (j) [L4], (k) [L5], (l) [L2].

designer had intended. For example, while the game may have a teapot in the environment, it does not necessarily mean that the teapot can be picked up, much less filled with water or used to pour liquid. Thus, the purpose of these visual interaction cues is to inform the player about what can be interacted with within the context of the virtual environment presented in the game.



Figure 4.2. An *Emphasized Discover* cue. A bag of gold coins is outlined in bright yellow to indicate it can be looted from [L3].

I generally consider *Discover* cues to help change a player's understanding of the environment—that is, what can be used, and what can be interacted with in the environment. For example, Figure 4.2 illustrates how *Dragon Age: Inquisition* [L3] uses an outlined highlighting cue to emphasize certain artifacts in the environment (here, that the gold pouch can be looted for gold). Figure 4.3 shows how *World of Warcraft* [L4] uses a "minimap" overlay (representing an iconic birds-eye-view of the entire game world) to show the player where mineable minerals and important characters can be found in the map relative to the player's location.



Figure 4.3. An *Overlaid Discover* cue. This minimap shows a birds-eye-view of nearby objects and points of interest [L4].

Look. *Look* cues are used by designers to focus a player's visual attention in a timely way. Many games feature time-based mechanics that involve events initiated by other agents, such as "enemies" (e.g. the enemy is shooting at the player), or objects (e.g. the pendulum is swinging toward the player). *Look* visual cues are sometimes designed as explicit hints provided by the game designer about an impending event (e.g. the pendulum will hit you). Other times, they seem to be designed to mimic the peripheral awareness one might have of the environment (e.g. Figure 4.5) to overcome inherent limitations such as the constrained viewport into the game world or the use of stereo sound rather than 3D sound (i.e. the enemy growled from behind the player's avatar).



Figure 4.4. An *Emphasized Look* cue. The enemy is highlighted in orange, indicating that he can be killed with a special player attack. This cue is triggered based on the enemy's hit points [L12].

I consider these cues to be designed to change what the player is doing in the environment. *Look* cues generally provide the player with a heightened awareness of something happening in the environment, or something that is about to happen in the environment. The player should then use this information to do something—be it to change the viewport, to engage in evasive maneuvers, etc. Figure 4.4 illustrates a *Look* cue in *Doom* [L12], where the enemy avatar is glowing orange; the bright glow indicates that the enemy is in a weakened state and can be killed if the player interacts with it at close range, providing the player with awareness information about the status of enemies. Figure 4.5 shows a *Look* cue where the yellow ring around the player's avatar points



Figure 4.5. An *Integrated Look* cue. A yellow widget painted below the avatar points at a nearby enemy that is suspicious of the player's actions [L17].

toward a nearby enemy position (relative to the player's location). Also, the red bars indicate that the enemy is currently suspicious of the player [L17].

Go. Finally, games frequently take place in large virtual environments that the player navigates through during the course of the narrative or gameplay to achieve goals in the game. *Go* cues are navigational cues that provide the player with guidance on how to navigate the environment to arrive at a destination. In most of the games in my sample, these destinations are fixed; other times, the destination is another object moving through the environment (e.g. representing another agent in the system). Regardless, cues in this category are intended to help a player move from one location to another.

Go cues are used to change a player's location in the game world. While it may still be a player's choice to respond to these *Go* cues, the intention is for the player to follow or move in a corresponding direction. These cues range in terms of how much information is provided as a navigational cue: some provide a direction relative to a current orientation, while others provide distance information, and still, others give a "walking path" to follow (e.g. *Steep* [L16] in Figure 1.1, left).

4.2.2 Dimension 2: Markedness

The second major dimension in my sample corresponds to some ideas first presented in [31, 32], where the dimension captures the extent to which the cue blends into the game environment (or how it stands out from that environment). This is distinct from notions of diegesis, which relates to the "story" of the game [11]. Here, I am strictly concerned



Figure 4.6. Bioshock uses environmental lighting as a Subtle Go cue [L1].

with the visual presentation or design of the cue: *Subtle, Emphasizing* an object, *Integrated* with the environment, or *Overlaid* atop the environment.

Subtle. Subtle cues are blended into the environment in such a way that they are difficult to distinguish from the environment itself. Such cues seem to be a part of the level or environment design, making use of lighting and contrast to draw a player's attention to features of the environment. While this can be done with garish neon signs (as part of the environment), this can also be done more subtly to guide a player's attention to visual features in the environment. As illustrated in Figure 4.6, the level design in *Bioshock* [L1] makes use of drastic contrast in lighting, where the purpose of the cue is to provide a



Figure 4.7. A *Subtle Discover* cue. A part of the wall is coloured with slightly offsaturation to indicate to players that the wall can be manipulated [L11].

player with a clear destination (*Go* cue). While the cue uses visual contrast, it does not stand out given the in-game narrative.

Figure 4.7 shows a *Subtle* cue in *Doom* [L11], where the wall's texture is slightly less saturated compared to nearby wall segments. This cues the player to activate the wall, as it leads to a hidden area (*Discover* cue). Figure 4.8 shows another example from *Dragon Age: Inquisition* [L3],



Figure 4.8. Dragon Age: Inquisition uses green smoke as a *Subtle Go* cue [L3].

where the player's next destination is a smoking tower, with smoke that is visible from a distance (*Go* cue). Such cues are fully unified with both the architecture and the gameplay mechanics, and so they are *Subtle* cues based on the context—it is not strange for a tower in *Dragon Age: Inquisition* to be smoking and for that smoke to be visible from a distance. Similarly, *Doom* [L12] uses flickering lights to attract a player's attention toward certain corridors, supported by the in-game narrative that fire has destroyed the base. Thus the neon lights are in a half-working state (*Go* cue).



Figure 4.9. An *Emphasized Go* cue. As the player gets close to the door, it becomes emphasized with a highlight around its edges [L7].

Emphasized. *Emphasized* cues highlight an existing object or surface in the game environment. This is done through various visual effects, for instance, via outlining the object, highlighting the object, or alternatively de-emphasizing every other object around the emphasized object. These effects do not add other virtual elements or objects into the game, rather the presentation of existing objects is amplified in some way. *Emphasized* cues are used to draw visual attention through distinctness or contrast.

As illustrated in Figure 4.2, *Dragon Age: Inquisition* [L3] emphasizes a money pouch with an outlining cue. This promotes discovery of the fact that the money can be "looted" (*Discover* cue). Figure 4.9 shows a highlighted outline effect from *The Witcher 3: Wild Hunt* [L7], emphasizing a door/doorway that the player is to pass through to progress in the game (*Go* cue).

Integrated. Integrated cues take the form of an added virtual object in the scene that is visible to the player but is not actually part of the game world. These virtual objects can track an object in the game world, and so their positions update correctly within the viewport as the player changes his/her view. Such *Integrated* cues range in form from text labels (e.g. "Enter here") to virtual arrows pointing at objects or other agents in the environment. Further, while these *Integrated* cues track the environment from the viewport, I observed that some deliberately ignore some aspects of space entirely. For instance, some ignore distance (where an icon representing a destination remains the same size regardless of how far away it is), others ignore orientation (text may be oriented so it is always legible to the player), while others may ignore both.

Figure 4.10 shows an *Integrated Discover* cue from *Thimbleweed Park* [L15], where a label appears to tell the player what actions can be taken on the object. Figure 4.11 shows a set of pillars in *Steep* [L16]. The pillars are virtual objects placed atop the game world



Figure 4.10. An *Integrated Discover* cue. A label appears to tell the player what actions can be taken on the object [L15].

that track the game world to show the player where to go (*Go* cue).

Note that, some first-person shooters make use of the same *Integrated* cue to represent a teammate in different contexts. However, the *Purpose* of this cue depends on the context of the gameplay. For instance, if the teammate is low on health, the cue could be considered as a *Go* cue ("Go help your teammate"), whereas in other non-combat situations, the exact same cue in the game could represent a *Discover* cue ("Your teammate is over here"). Thus, the usage of the cue is mostly context dependent, particularly as it relates to gameplay.



Figure 4.11. An *Integrated Go* cue. The added white line and red arches show the player where to go in the race course [L16].

Overlaid. Overlaid cues explicitly distinguish two different aspects of the player's viewport: first, the viewport into the game world, which shows the environment, and

second, a layer atop the viewport where UI elements sit atop the environment, and function largely independently of the changing view of the game world. *Overlaid* visual interaction cues that I found were represented either as UI widgets (e.g. a compass, birdseye-view minimap, aiming reticule), or widgets that made use of the edges of the screen to refer to objects or destinations beyond the side of the viewport into the world.



Figure 4.12. An Overlaid Look cue. A red indicator on the aiming reticule shows the player where the avatar is being attacked from [L5].

Figure 4.12 shows a screenshot from *Overwatch* [L5], where a red flare around the reticle is an *Overlaid Look* cue that tells the player that s/he is being attacked from the front (top edge represents front; right edge represents from the right side, and so forth). Figure 4.13 shows an instance of an *Overlaid Go* cue from *The Elder Scrolls V: Skyrim* [L2], where the compass, placed atop the HUD, shows the player which direction certain artifacts/destinations are relative to the player's current orientation.



Figure 4.13. An Overlaid Go cue. A compass at the top of the player's HUD shows "North" in the game, along with specific points of interest [L2].

Note that while video games typically only provide a limited field of view into the game world (e.g. a horizontal first-person viewing angle of 90°-120°), some cues may refer to objects outside of the field of view. A typical convention is to treat the display as an overlay where the centre of the screen represents the player's location, and the top edge represents what is in front, bottom edge what is behind, and so forth. For example, when a player takes damage in a first-person shooter, the edges of the screen may flash to indicate where the damage is coming from (i.e. if it is out of the field of view). Similarly, a related convention is to use arrows or icons at the edge of the screen to point to where an object is (e.g. Figure 4.14). The problem with this convention is that in principle, it could lead to confusion between objects that are literally "above" player in a 3D game



Figure 4.14. Jetpack Joyride uses a blinking *Overlaid Look* cue (top) to show where the rocket is about to appear on screen (bottom) [L10].

world with objects that are in front but indicated with an arrow at the upper edge of the screen.

I observed that some games make use of a visual transition in the type of cue that was being used based on whether the object was within the field of view. For instance, Figure 4.14 (top) shows an *Overlaid Look* cue for an object that is out of view; however, when the object enters the field of view Figure 4.14 (bottom), the cue changes to a *Subtle Look* cue [L10]. This transition is useful for players, as it helps to distinguish when something is within the perspective orientation vs. out of view. It makes sense for visual interaction cues to be visible when the target object or point of interest is within view; however, how games deal with obstructions (i.e. there are objects in the view that should obscure the view of the target) seems to be more of an explicit design choice. *Emphasized* cues are typically only visible when the target is visible. In rare cases, these *Emphasized* cues are visible through obstructions (e.g. teammate locations in *Left 4 Dead 2* [L18]). The intention here may be to provide a *Look* cue even when there is no clear line-of-sight. For the games in my sample, *Integrated* and *Overlaid* cues are visible regardless of whether there is an obstruction (perhaps a distinguishing feature of *Emphasized* cues). In the sample, *Integrated* and *Overlaid* cues never explicitly signal whether the target object or point of interest should be visible given the avatar's location and orientation. This, however, is something that would likely be of use in AR applications.

4.2.3 Dimension 3: Trigger

The third major dimension of my framework considers how the visual interaction cue is made visible. I identify four levels of trigger based on how much agency the player has over the trigger, from an explicit act by the player to triggers caused by other agents and finally to persistent cues.



Figure 4.15. A *Subtle Go* cue. Light reflects off the sword, pointing to an in-game destination; the player raises their sword to see this light by pressing a button [L14].

Player. *Player-triggered* visual interaction cues are activated by an explicit action by the player. The example in Figure 4.15 shows a player's avatar from *Shadow of the Colossus* [L14] holding a sword that shows the player where to go next. The player activates this by switching to hold the sword, and pressing a special key sequence (*Subtle Go* cue). Similarly, Figure 4.10 illustrates how a text cue shows when the player hovers his mouse above the oven holding the hot dogs in *Thimbleweed Park* [L15] (*Integrated Discover* cue). With *Player-triggered* cues, the player has full agency over when and if the cue is displayed.

Context. Context-triggered visual interaction cues are activated by the player through implicit actions. In the sample, a cue's "context" is typically comprised of a player's

location in the game world (i.e. entering a room or entering an activation area for a cue), or the player's view in the game world. For instance, in *Thief* [L8], "stealable" objects are highlighted when the player is near such objects, and when they are facing the object (*Emphasized Discover* cue).

Other/Agent. These are visual cues triggered by some other agent in the game: another player in a multi-player game, or another automated agent within the game environment. For example, the red damage indicator illustrated in Figure 4.12 (from *Overwatch* [L5]) shows the direction from which player is being shot from (*Overlaid Look* cue). These indicators are triggered by other players or non-player character (NPC) enemies. Similarly, in some games, a change in game state triggers the visual guidance cue.

Persistent. Finally, some cues are always visible. Examples of such cues include those that are built into the level's design (e.g. lighting and contrast in the level as in [L1]), or widgets that always appear on the HUD or UI atop the world (e.g. Figure 4.3 from *World of Warcraft* [L4]) (*Overlaid Discover* cue).

4.3 Summary

To create this framework, I studied 49 contemporary video games and used a thematic analysis process to group the visual interaction cues by theme. Over a number of iterations, I had developed the interaction cue framework. This descriptive framework rethinks the classification of interaction cues by ignoring diegetic distinction; instead, the primary dimensions it focuses on are the purpose of the cue, the visual design of the cue, and the circumstances under which the cue is shown.

The *Purpose/Task* dimension describes the types of tasks a cue can support (*Discover, Look, Go*). The *Markedness* dimension describes how the cues may be styled (*Subtle, Emphasized, Integrated, Overlaid*). And the Trigger dimension describes how the cue is "turned on" (*Player, Context, Other/Agent, Persistent*).

As a consequence of developing the framework, I had addressed *TQ1* (*What are the ways video games make use of visual cues to guide players through virtual worlds?*). This was the prerequisite for tackling *TQ2* (*How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?*), which the next two chapters will address. This is done by using the framework first as a way to describe visual interaction cues in AR applications, and then by using it generatively to create novel visual cues for AR guidance.

Chapter Five: Using the Framework with AR

Although I developed this framework by studying and describing interaction cues in video games, I view the primary application of the framework to be for generating ideas for interaction cues in augmented reality. In this chapter, I first describe how the framework dimensions apply to AR. Then, I show that the framework provides a practical vocabulary for describing and analyzing interaction cues in current augmented reality applications. Finally, I use the framework to suggest design alternatives for these applications.

This application of the framework to augmented reality begins to solve *TQ2* (*How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?*). This is made possible by satisfying objective *TO3* (*articulate how the framework cues can be ported to the design of AR applications, given the current technological considerations and limitations of working within a real-world setting*). The subsequent chapter will go a step further, actually creating new AR applications for guidance by using the framework as a design tool to create new visual interaction cues.

5.1 Mapping Framework Dimensions to AR

Two of the dimensions of the framework, the reasons for using interaction cues (*task/purpose*) and the interaction model (*trigger*), can be straightforwardly mapped to AR

applications. However, the visual design dimension (*markedness*) needs additional nuance in AR.

Task/Purpose. The video game task of *Discovery* of interaction points maps directly to real-world situations where it is unclear what artifacts in the world can be interacted with in an AR context. Not all artefacts in the real world have necessarily been registered, have content associated with them, or are available for interaction. Similarly, *Look* tasks can be mapped to context-sensitive/spatially-sensitive tasks. For instance, in a touring context, specific points of interest (e.g. a statue) may only be visible from the current location. Finally, *Go* tasks are like those in video game contexts—for example, where the system provides navigational guidance to the user to get to some location.

Trigger. From the game framework, I map the *Player-triggered* cue to a *User-triggered* cue, where the cue is made visible by an explicit user action, such as turning on layers from the UI, or by triggering a "navigation mode" on a GPS device. *Context-triggered* cues refer to a change in the state of the relationship between a user and his/her environment. In most AR contexts today, this refers to a user's movement through space, where the user's spatial location activates a cue (e.g. information about the space s/he has just entered). Intelligent interfaces could track the other elements of that context—for example, a constellation app might track the geographic position of the user and the time of evening,
using this information to trigger cues only for what should be visible in the night sky. *Other/Agent-triggered* cues could map to cues activated by other actors in the environment, be they humans or automated agents. Finally, *Persistent* cues always remain on and require no explicit action by the user.

Markedness. This framework defines four levels that describe how overt a cue is in its presentation—that is, the extent to which it stands apart from the game world. These levels can be immediately mapped to the AR context. *Subtle* cues are those that look like a part of the environment. These are necessarily spatially tracked, but beyond this, fit seamlessly into the overall visual environment such that they are effectively indistinguishable from it. *Emphasized* cues are those that highlight an object or a surface in the environment. The *Emphasized* cue could be one that makes the object recognizably distinct from the surrounding environment; however, the key is that it highlights an *existing* object or surface rather than adding a new virtual object to the environment (e.g. Figure 5.1.b). *Integrated* cues add some new virtual object into the environment in a tracked manner—that is, as the AR view changes, the virtual object stays properly

"affixed" spatially. Finally, *Overlaid* cues are recognizably affixed to the screen rather than an object in the physical world (e.g. Figure 5.1.d).



Figure 5.1. Variations on an imaginary AR interface that provides a Go cue to the door on the left.

The central departure when applying the framework to AR is this markedness dimension. From a technical perspective, every visual augmentation in AR is drawn over the view of the world (as *Overlaid* or *Integrated* cues are in games). The principal distinction between an *Integrated* cue and an *Emphasized* cue is focus—*Integrated* cues are entirely new objects visualized in the scene, whereas *Emphasized* cues highlight existing objects or surfaces in the scene. The distinction between an *Integrated* cue and a *Subtle* cue in AR is fit—*Subtle* cues need to look and feel as though they are a part of the environment. This depends on objective factors such as photorealism (does it visually

look like it fits), but also contextual fit to the environment (does it make sense in the context). This contextual fit is subjective, depending on a user's pre-existing knowledge about the context.

Figure 5.1 illustrates this subjective dilemma, where the series illustrates a set of imagined variations on an AR interface that is directing the user to the left: (a) is the actual view; (b) emphasizes the desired door; (c) uses an *Integrated* arrow cue; (d) shows a bird's eyeview overlay; (e) makes it appear as though two of the doors are closed; (f) darkens two undesirable entryways, leaving the desired path lit; (g) makes it appear as though there is only one door; (h) places "cleaning" signs on two of the doors; (i) places a set of photorealistic boulders in the path of two of the paths, while (j) does the same, but with cartoon boulders. If we consider each of (e)-(i) to be photorealistic, these are candidates to be Subtle cues. With respect to the fit to the context though: (i) would be considered Integrated, as boulders are rarely found in office environments (and thus does not fit the context); if the user had pre-existing knowledge of the environment (there are three doors), (g) may not be considered *Subtle*; similarly, (h) could fit in the context if such signs are typically found in the environment. Thus, the markedness dimension in the AR context is determined by both presentation and fit to the environment (which, in turn, is a subjective issue beyond the control of the designer).

5.2 Describing AR Interaction Cues

Next, I show how my framework can be used to describe AR interaction cues, using three examples of existing AR applications: Reitmayr and Schmalstieg's Vienna Tour Guide [44], the Lumin project [35], and Aero Glass [25].



Figure 5.2. Reitmayr and Schmalstieg's AR tour guide system [44].

Example 1 – Vienna Tour Guide System [44]. The system supports three tasks: navigation, information browsing, and annotation. In the navigation mode, one person can choose to follow another person as the target. In this mode, the system draws a purple frame around the target person (as illustrated in Figure 5.2, left, a *User-triggered Go* cue). Waypoints nearest the tracked user are used for guidance (*Integrated Go* cues visualized as red cylinders). These waypoints are *Context-triggered* cues, based on a mix of position and orientation of the user. A yellow walking path (tubes) are an attempt to show a direct path on the ground to the next waypoint (*Integrated*). Finally, as illustrated in Figure 5.2



Figure 5.3. The Lumin Project [35] gives museum-goers an AR experience for navigating exhibits and learning about artefacts. The AR view of artefacts provides Integrated Discover cues for more information.

(right), the system shows a yellow polygon around an annotated building (an *Integrated Discover* cue), signalling to the user that they can tap to get more information.

Example 2 – Lumin Project [35]. The Lumin Project is a handheld AR tool deployed at the Detroit Institute of Arts. It has two modes: a navigation mode and an information mode. In the navigation mode, the AR view shows the path to the target location with *Integrated Go* blue dot cues that hover above the ground (these are *Persistent*, Figure 5.3 - left). In the *User-triggered* information mode (Figure 5.3 - right), *Integrated Discover* cues are placed on the mummy sarcophagus to indicate points of interest that users can tap on to show additional information about the mummy.

Example 3 – Aero Glass [25]). The Aero Glass system provides pilots with real-time augmented reality support while flying. This primarily consists of a flight path drawn in the pilot's field of view, with the pilot being provided with targets through which they

must fly (*Integrated Go* cues as blue arrows with a magenta border, Figure 5.4 - left). Objects in the world have been registered and mapped, which allows for the use of *Emphasized Discover* cues. In Figure 5.4 (centre), the runways are tightly outlined with thick white borders and tracked by the Aero Glass system. Other potential hazards such as communication towers are also emphasized. Finally, the system presents the user with *Integrated* virtual signs while taxiing on the runway (Figure 5.4 - right).

These descriptions show that the framework is effective for unambiguously describing the task (*purpose*), the visual design (*markedness*) and interaction design (*trigger*) of interaction cues for AR.



Figure 5.4. Aero Glass [25] uses a combination of Integrated and Emphasized cues to provide pilots with navigation information while flying, using a head-mounted display.

5.3 Generating AR Scenarios with the Framework

Using the framework, I can also generate new design ideas for each of the examples above. I have found describing user stories to be a powerful mechanism to begin producing these ideas. I illustrate this approach by specifying a set of new scenarios given the examples above.

Example 1 – Vienna Tour Guide System [44].

Tina the Quick Tour Guide. Tina is a fast tour guide, and sometimes she moves to new landmarks before her tour group is ready. Tina's app immediately notifies her with an on-screen *Emphasized Look* cue that outlines the bodies of tourists who have gone beyond her immediate view (they have wandered too far). If they are not within her viewport, an *Overlaid Look* cue appears: arrows at the edge of her view show her how to turn her view so her clients are in front of her (and this transitions to the outline view). The first is *Other/Agent-triggered* (based on the tourists' location); the latter is *Context-triggered* based on Tina's orientation. Later, Tina realizes that she may still not have all the people in her group. From her app, she activates a function that traces a virtual "leash" to each of the members of her tour group, an *Integrated Go* cue showing her where her clients are.

<u>No-SIM Ned</u>. Ned was part of the tour group but is now lost without an active SIM card. He can use his AR app to identify likely locations for the tour group and uses the *Integrated Go* cues from the original app to navigate to the right points of interest. There is an *Overlaid* map which acts as a *Go* cue so he can see his current location, as well as the path the tour was to take.

Example 2 – Lumin Museum App [35].

<u>Ross the Curator</u>. Ross knows attendance is low at live shows because patrons lose track of time and do not know when the show is on. Ten minutes before the show, based on where patrons are in the museum, a *Context-triggered Subtle Look* cue appears on visitors' apps, informing them of the show that is about to start. The cue is in the form of a blinking spotlight simulating someone turning on and off the lights in the room.

Fei the Science Fan. Fei gets the *Look* cue for the show and wants to attend. She activates the *Integrated Go* cue which is styled as a set of footsteps on the ground, leading her to the show. As she is en route, the system takes note of a celebrity archaeologist who is in the venue (signing autographs) and provides a *Context-triggered Look* cue to her when she passes by.

Example 3 – Aero Glass [25].

<u>Fiona the Pilot in Training</u>. Fiona is a pilot in training and must learn to act in specific emergencies. While she is flying, her instructor gets her into an emergency situation to practice safely recovering from it, and the Aero Glass system flashes an alert as a bold message within Fiona's field of view (*Overlaid Look Context-triggered* cue). The system then highlights the knob she must adjust with a bright pulsing highlight (*Emphasized*), along with an animated 3D arrow indicating the direction she needs to turn it to rectify the

situation (*Integrated*). During this emergency, all non-vital cues are hidden to focus her attention on the crisis, decreasing potential information overload (*Context-triggered*).

Jai the Passenger. Jai is a passenger on a flight and has an interest in seeing certain monuments. From a distance, these structures are marked with *Integrated* text labels above them. As the plane approaches closer to the monument and Jai can see its features, the cue becomes highlighted in a bright colour to make it stand out in Jai's peripheral vision (*Emphasized Discover Context-triggered*). As he shifts his gaze to the monument, the highlight colour fades (*Context-triggered*) so that the cue does not interfere with the natural beauty of the structure. Finally, the AR system virtually fogs the glass on the window through which Jai is viewing the monument, obscuring all of the other structures in the view and making the one monument stand out (*Subtle Look* cue).

5.4 Summary

These scenarios illustrate how the framework functions as a generative tool, allowing designers to explore new possibilities for the use of interaction cues in AR. For instance, to generate the above examples, I start with the user story, identifying the intended purpose of the cue. My next consideration is markedness, where I explore how visible the cue needs to be (e.g. Figure 5.1). Based on the expected usage scenario, I consider different trigger opportunities and identify the one that best matches how I want the user to see the cue. The vocabulary provided by the framework gives me a precise language to

describe and think about these cues. For instance, if *Tina the Quick Tour Guide* has lost her clients, *Subtle* cues are inappropriate—in some cases (e.g. a group of school children), she needs to find her clients immediately; similarly, an *Integrated* cue may be inappropriate to tell her where the lost tourists are, as she may not have the tourist within her field of view. Finally, *Other/Agent-triggered* cues based on clients' locations are best suited for her scenario; a *User-triggered* cue may be triggered too late, and a *Persistent* cue would distract her from her primary task.

This chapter, in part, addressed TQ2 (How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?). It did so by examining some of the issues of porting the framework from the virtual worlds of games to the real-world environments of augmented reality. I have also shown that the framework stands up when being used to discuss cues in AR applications and that it can be used to generate entirely new cues that may improve the designs of augmented reality applications for guidance. The following chapter goes more in-depth into answering TQ2 by discussing how I used the framework to create entirely new AR experiences.

Chapter Six: The Prototypes

Having worked through a naïve approach to the creation of designs for augmented reality guidance (Chapter Three), I put the visual interaction cue framework to use as a guide to create prototypes using a more informed approach. In this chapter, I explain the informed approach and describe the prototypes I developed for three separate AR guidance tasks. These prototypes demonstrate how a varied range of solutions can be produced by using the framework as a guide. I then reflect on the process and compare it to Chapter Three's naïve approach.

Through the creation of these prototypes, this chapter finishes answering *TQ2* (*How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?*). It does so by satisfying *TO4* (*create augmented reality prototypes as a proof of concept, using the visual interaction cue framework as a guide*).

6.1 Design Approach

In Chapter Three, I reported on the naïve approach to the design of AR cues for guidance and reflected on how I struggled with that tactic. The main issue I encountered was a lack of understanding of the possible solutions to the problem. After developing the framework, I now had a new approach I could take. Rather than just coming up with ad hoc examples, I formalized the process by using the framework to guide my designs. I started from the *Purpose* dimension, considering the types of guidance I might support, and coming up with example tasks for *Discover*, *Look*, and *Go* cases. For each of those tasks, I created designs for the various styles of *Markedness: Subtle, Emphasized, Integrated*,

	DISCOVER (BOOKSHELF)	LOOK (SHOULDER SURFING)	GO (OFFICE MAIL DELIVERY)
SUBTLE	DMPUT & Internet of States		CALVIN STARK
EMPHASIZED			
INTEGRATED			
OVERLAID		0	

Table 6.1. Prototype screenshots mapped to framework dimensions. (See Figure 6.4, Figure6.5, and Figure 6.6 for descriptions.)

and *Overlaid*. (Table 6.1 shows final screenshots of prototypes; Figure 6.1, Figure 6.2, and Figure 6.3 show examples of sketches used in the design process⁷.) Using the framework guaranteed variations in the designs, which was not something the naïve approach could offer. To keep the scope reasonable, the Trigger dimension is not varied in the designs; all of the designs are *Persistent*, but 6.4.4 Proximity imagines how *Context* can be used to control the cues, based on the user's distance from the target.



Figure 6.1. Sketch of *Emphasized* cues highlighting books in a bookshelf.

⁷ See Appendix C for more sketches.



Cue

Figure 6.2. Sketch of Subtle cue casting a virtual shadow on a sensitive document



GO TASK - DELIVERING MAIL IN AN OFFICE.

Figure 6.3. Sketch of Integrated cues marking desks in an office with flares of light.

6.1.1 Autobiographical Design as a Method

While the HCI community has traditionally stressed the importance of user studies, researchers have recently challenged the necessity of usability evaluations [24]. One increasingly accepted approach, which does not centre around user studies, is autobiographical design [39]. Autobiographical design is, according to Neustaedter and Sengers, "design research drawing on extensive, genuine usage by those creating or building the system", where "genuine" refers to the system fulfilling some kind of need that is unique to the designers. Among other hallmarks of this method, autobiographical design: supports rapid development, requires real systems, and easily reveals significant issues. For my work, the essential characteristic of autobiographical design is that it promotes early innovation, where generalizability is less critical than exploring new ideas and creative solutions.

My approach to this work is autobiographical: I had a genuine need, and I created my own framework and prototypes to address that need. As with other types of case study, the results are not generalizable. This is acceptable as this is early, exploratory work, meant to raise more questions than it answers. The main departure in my usage of the method is that I lack an actual system, and instead the focus is on the creation and use of a framework. If we consider the framework a type of system that is designed, used, and reflected upon, then there is no reason the principles of autobiographical design would not translate in my case.

6.2 Bookshelf Prototype (Discover)

6.2.1 Task

In this *Discover* task (Figure 6.4), the user is a patron of a library or bookstore, shopping for books within specific categories such as author or genre. These bookshelves are not



Figure 6.4. Bookshelf (Discover) cues. (clockwise from top left) *Subtle* – a sticker is rendered on the spine of the book, *Emphasized* – an animated particle effect wraps the spine of the book, *Integrated* – a virtual 3D arrow points to the book, *Overlaid* – a minimap shows approximate locations of the books.

well-organized, so the user cannot easily see where all the books they might like are shelved. This is a *Discover* task since the user is not interested in a specific book but just wants certain books to stand out from the others.

6.2.2 Implementation

This prototype runs on the Google Tango tablet [21, 43] and was created in Unity [49] with the Tango SDK. I also created an authoring tool that allowed me to define bounding boxes around certain books in the bookshelf as targets. The books themselves were not registered in the system, and the Tango tablet only understands its position in the library, so I manually defined the bounds of the books so that the visual cues could be rendered in the correct positions. This solution is clearly only acceptable in a controlled lab environment but is suitable for this early proof of concept work.

6.2.3 Cues

Subtle. This *Subtle* cue adds a virtual sticker to the spine of all the books of interest. Books in real libraries may have such stickers as markers, making this a good fit for the environment and context. Thus, this sticker decoration is a *Subtle* cue. The cue is even styled with worn edges, making it visually similar to a sticker that has been on a book for some time and is showing age.

Emphasized. The spines of the books of interest are wrapped with an animated particle effect, making them sparkle and stand out from the others in the library. This is an *Emphasized* cue, as it uses the form of the actual book and adds a visual decoration.

Integrated. A virtual arrow is attached above the books we want the user to see. As this is an entirely new virtual object added to the space, it is an *Integrated* cue.

Overlaid. The prototype draws a minimap over the view, representing the face of the bookshelf, with books of interest being rendered as small icons on the map. As the minimap alone was difficult to understand in this case, lines were added, connecting the icon in the minimap to the actual book in the view. (These lines are *Integrated* cues.)

6.3 Shoulder Surfer Detection Prototype (Look)

6.3.1 Task

This Look task (Figure 6.5) aims to make the user aware of other people in the environment while reading a sensitive document in public. The prototype makes the user aware of where others are in the environment and/or whether they are looking at the paper (a social engineering technique called "shoulder surfing")⁸. This is a Look task

⁸ This task is inspired by the work of Brudy et al. [14]

because the user must immediately respond to the cue to protect their personal information.

6.3.2 Implementation

This prototype differs from the *Discover* and *Go* prototypes in its implementation, using Apple iOS [3] devices instead of the Tango setup used in those systems. This *Look* prototype used Vuforia [52], an augmented reality SDK for mobile devices, to register the



Figure 6.5. Shoulder surfer detection (Look) cues. (clockwise from top left) Subtle – a virtual shadow is cast on the document, *Emphasized* – the document is given a red border when someone is peeking, *Integrated* – a virtual light beam is drawn from shoulder surfer to point on the document they are looking at, *Overlaid* – an icon is drawn at the edge of screen indicating where the shoulder surfer is.

marker used on the paper that the user is looking at. As with the other prototypes, this system used Unity [49] as an engine. This application also used the following assets from the Unity Asset Store [50]: The Off Screen Indicator asset for the *Overlaid* cue [41], a model for the *Subtle* cue shadow [2], and an asset for rendering volumetric light for the *Integrated* cue [51]. The user views the document through an iPad, and the position and gaze direction of the "shoulder surfer" is controlled by an iPhone; the iPhone camera acts as the eyes of the shoulder surfer. Position and orientation information from the iPhone is communicated to the user's iPad application via Unity's built-in networking functionality. The iPad app uses that data to render the cues on the user's device.

6.3.3 Cues

Subtle. When another person is near the user's document, the application renders an artificial shadow over the paper, even if the passerby is not casting a physical shadow. People use shadows to understand where objects exist in their surroundings in the real world, making this a good fit for a *Subtle* cue. This cue only lets the user know someone is nearby and does not communicate information about the passerby's gaze direction.

Emphasized. When the user is safe (that is, nobody is shoulder surfing on the document), the prototype outlines the paper in a green border. If someone else gazes at the document, it becomes outlined in red, letting the user know there is a danger of shoulder surfing and that they should protect the document. The actual physical document is outlined by

the app, making this an *Emphasized* cue. I directly ported this cue from the work of Brudy et al. [14], taking it from a large public display to a mobile device screen.

Integrated. When someone is shoulder surfing on the document, the user sees a beam of light, indicating where the gaze of the shoulder surfer is focused. This beam can be followed from the document to the face of the passerby, letting the user see exactly who is viewing the document. This is a new object added to the scene; an *Integrated* cue.

Overlaid. When someone is passing by, the application draws an icon over the view at the edge of the screen. For instance, if the passerby is to the user's right, the icon is drawn on the right side of the display. If they are behind the user, the icon is drawn at the bottom of the screen. Further, if the passerby is gazing at the screen, the icon is shown as an open eye and coloured red. If they are not gazing at the document, it is a closed eye, just making the user aware of the other person's presence in the space. This is Overlaid, as it is strictly drawn over the user's field of view on the 2D plane through which they are viewing the environment.

6.4 Office Mail Delivery Prototype (Go)

6.4.1 Task

This *Go* task (Figure 6.6) guides a mail delivery person through an office space, helping them to deliver letters to specific desks in a set order. The user is given letters with only the name of the recipient, and the app provides guidance to the desk of the recipient. The employees in the office have preferences for where the mail is delivered, such as in a mailbox or on a specific part of their desk. This is a *Go* task because the user must be guided to specific locations in the office.

6.4.2 Implementation

The implementation details of this prototype are the same as for the Bookshelf (*Discover*) task; Google Tango [21, 43] was used as a mobile device, and a special authoring



Figure 6.6. Office mail delivery (Go) cues. (clockwise from top left) *Subtle* – a handwritten label is placed on the desk, *Emphasized* – the drop off location is surrounded by a 3D cuboid, *Integrated* – a flare marks the next drop-off location, *Overlaid* – an on-screen compass orients the user toward the next desk.

application was created to register the targets in the office space. An asset from the Unity Asset Store was used to create the *Overlaid* compass visualization [16], and the beams in the *Integrated* cue were rendered using the Volumetric Light Beam asset [51].

6.4.3 Cues

Subtle. The application decorates the desk that the delivery person must visit with a virtual name tag. This tag is styled as something you might see used as a temporary tag in the real world: A handwritten name on a lined piece of paper, stuck to the desk with masking tape. Since it fits with the environment and context, it is a *Subtle* cue.

Emphasized. The area of the desk where the delivery person must store the mail is outlined with a semi-transparent 3D outline. As with the name tag, only the current target's delivery location is shown. This is *Emphasized*, as the cue highlights an object or surface in the real world.

Integrated. Beams of light or flares mark the next three desks that the delivery person must visit. The brightest flare is the current target, and the faded flares mark future targets, allowing the user to get an idea of where they will be going after they visit the next desk. These are *Integrated* since the beams are brand new 3D objects placed in the scene.

Overlaid. A compass drawn at the top of the user's view guides them to the next desk. When the user centers the icon in the compass, they will move toward the target desk if they walk straight ahead. The distance to the desk is shown as a label below the icon.

6.4.4 Proximity

I extended this mail delivery prototype to combine all of the cues into one view, with the cues being toggled on and off depending on how far away the user is from the target. From the furthest distance, the *Overlaid* compass is displayed to get the user moving in the direction of the target. At a certain distance, the flares are shown (*Integrated*), and the compass view is toggled off. Once the user is in the vicinity of the desk, the name tag is shown (*Subtle*), and the drop-off location is *Emphasized* with the bounding box around the delivery area (e.g. mailbox). An additional *Integrated* line cue is used in this mode to draw a line between the target desks, styled as a sparkling line along the ground (Figure 6.7).

This proximity view was created to explore how these cues could be combined to support each other. Video games rarely only rely on one specific cue for a task, and it is likely that AR applications will need to employ multiple cues for the best communication of the guidance directions.



Figure 6.7. An animated line is drawn between waypoints in the Proximity variation of the Office Mail Delivery prototype. This is an *Integrated* cue.

6.5 Reflection

The primary benefit of using the framework over the naïve approach discussed in Chapter Three is that it provided a more formal process through which to create cues. If I were developing a cue to support the experience of a nature hike and giving a user information about plant life in the environment (*Discovering* objects in the world), I would start by thinking about how these cues could be styled to fit into the environment (*Subtle*). This would ensure that they would not take away from the user's immersion in the environment. I would want this to be something the user could trigger themselves (*User-triggered*), again, to keep the app from interfering with the experience. The framework

allows me, as a designer, to take my needs and map them directly to how the cues are designed.

Using this process, the cues I developed while brainstorming were notably varied; I explored solutions for all categories of each dimension. For instance, I would start by saying something such as, "What is the *Emphasized* solution?". Even if the problem did not call for an *Emphasized* cue, it was still worth exploring what such a cue might have looked like, as part of the brainstorming and design process. My instinct in taking the naïve approach was to create bold, gaudy designs, visually assaulting the user so that they would not miss the instruction. Using the framework, however, forced me to explore a broader range of possibilities in my designs. Further, I feel that working through the process of developing the framework from so many examples of cues from games has provided me with a sort of mental library of samples from which to draw for inspiration when designing new solutions.

The primary concern I raised in Chapter Three was that I did not feel confident that I knew what types of designs I should test against each other to compare the efficiency of AR designs for guidance. As I look at the range of examples in Table 6.1, I would be confident using the framework to now design cues that could be compared with each other.

6.6 Summary

The prototypes presented in this chapter demonstrate how the visual interaction cue framework can be used as a tool to support the development of AR designs for guidance. To create the prototypes, I first defined the task I wanted to support, and then developed cues that satisfied all of the categories of the *Markedness* dimension for that task. These were all created using handheld mobile devices and were developed in the Unity game engine. This process was, for me, an improvement over the naïve approach, as it provided a more formal procedure and ensured broader variations in my designs.

The creation of these prototypes satisfied objective *TO4* (*create augmented reality prototypes as a proof of concept, using the visual interaction cue framework as a guide*), which served to finish addressing *TQ2* (*How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?*). The following chapter examines essential discussion points and outlines potential future work.

Chapter Seven: Discussion and Future Work

This thesis has now addressed all of the thesis questions and objectives offered in Chapter One. The development of the visual interaction cue framework has formalized language to describe visual cues used in video games to guide players, and the creation of the AR prototypes utilizing the framework demonstrates that it can also be used to describe and generate visual cues for augmented reality. In this chapter, I apply the knowledge gained from working through this process to provide design recommendations for the development of cues for AR guidance. Further, I argue that the framework can support design for asynchronous collaborative AR, a currently underdeveloped research space. While questions TQ1 and TQ2 have been tackled, the main research problem, "How should we design visual cues for augmented reality guidance?", can be explored even further. Thus, I also break down the issues that should be addressed in future work. I do this by envisioning how the framework may be extended to describe other types of cues or be applied to other domains, as well as propose the types of evaluations that may be conducted to assess the effectiveness of cues on various criteria.

7.1 General Design Considerations

7.1.1 Limitations Presented by the Physicality of AR

While video game designers have considerable latitude in how to design interaction cues, AR designers are more limited. Physical objects have inherent immutable properties: most are static in shape and are not actuated. While some games use animated/moving objects (e.g. an object that slowly bobs or changes its size to get a player's attention), AR designers are limited to visually rendered cues. Video game designers also have stylistic and artistic license in the designs they use for interaction cues. Some games use cinematic cut scenes to visually re-orient a player with respect to an enemy or other point of interest by taking camera control away from the player and moving it around to give the player a third-person perspective (e.g. boss fights in *Shadow of the Colossus* [L14]). AR designers cannot use this type of cue.

7.1.2 Importance of Discover Cues

Based on existing AR experiences (e.g. [23, 28, 34, 35, 44]), I expect *Discover* cues to be extremely important in the early days of AR for the following reasons:

- Not everything in the physical world is interactable or tagged.
- Users do not yet have well-developed mental models of the classes of AR applications that might exist (i.e. since certain classes of applications may make some objects more or less likely to be interactive targets).

 The visual language of these kinds of interaction cues in AR is not yet welldeveloped.

While the first two issues may end up being overcome over time, it is essential for designers to consider how to evolve the visual language. Designers must carefully consider the interplay between expectations of the physical environment and the AR application to create effective designs.

7.1.3 Trade-off: Markedness

The markedness dimension of the framework raises noteworthy questions about "naturalness" and "jarringness." Users in an AR context are not bound to the content provided by the AR headset and can wander with complete freedom. If *Subtle* cues are designed too subtly in an AR context, they could be missed (which is likely undesirable). Hence, while *Subtle* cues may seem desirable as a goal (as they are in games [48]), it may be more effective to use more marked cues if the intention is for users to see them easily.

The application context also has a role to play here: if the AR experience is intended as a tool, easily visible (i.e. well-marked) cues are probably desirable. In contrast, an AR game designer may instead aim for the player to experience challenge and opt for less marked cues. Similarly, the aesthetics of the environment should be considered: if the location is renowned for its beauty, cues should not interfere with the user's experience of the space.

7.1.4 Trade-off: Triggers and Causality

I have outlined a breadth of ways in which cues can be triggered, ranging from things that are conceptually "close" to the user (*User-triggered*) to things that are conceptually "far" (*Other/Agent-triggered*). *User-triggered* cues are easy for users to understand, whereas *Other/Agent-triggered* cues may be difficult for users to understand since the cause of the trigger may not be visible. To this end, designers should use primarily obvious triggers (e.g. *Persistent, User-triggered*, and *Context-triggered*, but only where context is obvious, such as a physical position in space).

7.1.5 Visibility and "Fit" of Cues

Current sensing technologies limit AR designers: most do not yet capture a high-fidelity model of the scene. Consequently, most AR is unable to properly clip the rendered visual based on what ought to be visible, and simply render atop people and objects in the scene (cf. [7, 8, 10, 44]). Clipping these visuals properly will aid in the interpretability of cues. Similarly, designers cannot yet accurately use *Emphasized* cues on physical objects. For example, the Vienna Tour Guide [44] uses a yellow *Integrated* polygon around points of interest. An *Emphasized* cue that tightly highlights or outlines the tower would be more effective and aesthetically appropriate; however, current consumer-grade technologies cannot track real-world objects with sufficient granularity and fidelity for this. This

explains why the bulk of interaction cues in the AR context are *Integrated* cues, rather than *Emphasized* or *Subtle* cues.

Beyond the technical challenge of rendering photorealistic visuals for *Subtle* cues, a designer needs to consider: the physical context that the cue appears in (e.g. while boulders as subtle cues may not work in an office context, they may be appropriate for certain outdoor contexts); the user's mental model of the environment coming into the situation (e.g. how much does the user already know about the environment, how willing are they to suspend disbelief), and the user's mental model of the artefacts being rendered (e.g. does the user know enough about the situation to understand cue in context).

While beyond the scope of the discussion here, the near-future ability to render useful *Subtle* cues raises interesting ethical questions: is it right to show someone something that is not present, or to visually take away an object that is physically there? This is particularly important if the alteration of the user's view could lead to accidents or injury.

7.1.6 Impact of Field of View

Wearable displays have a notably small field of view (FOV) (typically less than 30 degrees) [12], and mobile phone- and tablet-based AR also only takes up a small amount of a user's FOV. Designers should assume users can see both the AR perspective and a real perspective on the world simultaneously with such low fields of view. Here, *Subtle*

cues will seem jarring, as users will be able to see differences between perspectives easily. Further, a user's foveal vision is generally augmented, but low FOV means that AR views will not augment peripheral vision, so designers will need to compensate for that lack of peripheral vision in some way (for example, by using overlaid cues as in the Off-Screen Guidance prototypes to more explicitly guide to targets not currently in view).

7.2 Designing for Asynchronous Collaborative AR Experiences

Video game designers and video game players engage in an asynchronous communication activity together. When a video game asks a player to move to specific points in the level or to focus on certain elements in the space, the player is following the instructions of a collaborator (the game designer). This is effectively the type of "Different Time / Same Place" communication task with which CSCW is concerned [30], with the game world being the shared "place." All of the issues of asynchronous collaboration persist here. For example, collaborators involved in asynchronous communication cannot ask clarifying questions when instructions are unclear, so instructions must be carefully crafted when they are being produced [29]. Also, synchronous collaborators can convey implicit information through non-verbal cues instead of strictly with explicit annotations, making for more productive communication; asynchronous communication needs to account for this lack of implicit information.

Making the argument that research into augmented reality for collaboration is largely focused on synchronous experiences, Irlitti et al. [29] break down various issues that must be investigated for asynchronous AR collaboration. These use cases involve both a "producer" and a "consumer" collaborator; the producer creates the annotations for use by the consumer. The development and application of my visual interaction cue framework makes a major contribution in addressing the authors' concern that the visualization of situated annotations must be improved through research (visualization of situated annotations⁹). As I have demonstrated in Chapter Six, the framework was, in my case, effective for producing a breadth of solutions. Annotations for collaborative AR have fundamentally been just "simple scribing" for a collaborator to view at a later time (for example, drawing explicit Integrated cues on the view of the environment). Various other levels of markedness should also be explored to improve intelligibility and quality of visualization of cues for asynchronous collaboration.

Apart from merely learning how to visualize the cues, Irlitti et al. are also concerned with how to group and order multiple cues (*temporal ordering and clustering of information*⁹). Both the *Markedness* and *Trigger* dimensions of the framework could be utilized in

⁹ These refer to points on a list of suggested topics of investigation presented by Irlitti et al. [29]

learning how to group and order cues. Implicit information (*passive temporal annotations from user actions and behaviour*⁹) that may be less important could be visualized using more subdued *Integrated* and *Emphasized* cues but could also be visualized as *Subtle* cues. Further, various *Triggers* could be used to limit information overload and display only priority information, depending on the task (as discussed in the proximity variation of the Office Mail Delivery task in Chapter Six).

7.3 Future Work

Developing the interaction cue framework and applying it to the design of visual cues for AR applications has raised some questions, which I believe are worth exploring in future work. Here I speculate about: how the framework could be used to describe other types of cues, how the framework could be applied to domains other than augmented reality, and how the types of cues may be evaluated in future studies.

7.3.1 Application to Other Cues and Domains

The framework may prove useful in learning how to describe different types of (nonvisual) cues in video games. Consider auditory cues. These could be described using the *Markedness* dimension of the framework as follows:

• *Subtle*: These cues would be sounds that are styled to fit the environment. For instance, a bird chirping in a forest could cue a player to the location of a nest they need to visit on a fetch quest.

- *Emphasized:* Auditory cues could be cues that fit in the environment but are enhanced in some way. An example of this would be if the sound of the birds chirping in the *Subtle* example was increased in volume or distorted in some way to make it stand out from other environmental sounds.
- *Integrated:* Entirely new cues that do not fit the environment. The sounds of the birds chirping could be replaced with a type of chime from a musical instrument. This does not exist in a forest and is an artificial sound meant to grab the player's attention explicitly.
- Overlaid: In a game that makes use of 3D sound, flattening the cue into a mono channel and removing it from the 3D space could make this a kind of Overlaid cue. For example, a player receiving verbal directions from another character over a radio could be considered Overlaid (though some new terminology would need to be created to describe this type of cue effectively).

This is merely a preliminary example, but the framework seems to hold up for describing auditory cues. Research should be conducted in the future to learn whether this framework could be used to describe things like auditory or haptic cues. Minimally, it should provide a starting point for such work, and the method I used in this thesis to translate visual cues from games to AR could also be utilized (*TC2*).
The framework could reasonably be applied to other domains and types of augmented reality. As long as a technology is meant to guide the attention of a user and has a robust spatial connection to the environment (physical or virtual), this framework should be useful. Some examples of this might be: AR navigation guided by drone projections [33], proxemic interaction around large public displays [14], and VR house tours for real estate sales [1].

7.3.2 Evaluation

At the beginning of this thesis, I discussed my motivation for taking on this work: comparing AR visualizations for real-world guidance. The naïve approach exposed a substantial issue, which is that I had no idea how to begin designing the visualizations at all. It was clear that I needed to learn more about how cues could be designed before they could be evaluated.

The interaction cue framework provides a kind of guide for beginning these evaluations. For a given task, a range of solutions (e.g. different levels of markedness) can be tested on various criteria. My initial concern was task completion time (which I explored in my pilot study), but criteria such as immersion, interpretability, information overload, and jarringness could all be evaluated as well, depending on the needs of the researchers and designers. I would not expect any one type of design or cue to be superior to all others in all cases, and it is likely that the results would be mostly dependent on the kind of task for which they are designed. For instance, a practical solution for guiding first responders in an emergency situation may not be appropriate for a tablet-based tour guide application.

7.4 Summary

In this chapter, I presented several points of discussion after having worked through the process of developing the interaction cue framework and applying it to AR designs. First, I provided general design recommendations for developing AR for guidance. Then, I argued that video game designers and players engage in asynchronous collaborative guidance tasks and that asynchronous AR designers could use the framework to develop new types of visualizations and triggers for guidance. Finally, I discussed how the framework could be applied to different domains and types of cues, and proposed different kinds of evaluations that might be worthwhile in future research. These types of future work are vital in further addressing the main research problem: *How should we design visual cues for augmented reality guidance*?

In the final chapter, I revisit the questions, contributions, and objectives that I presented in Chapter One and provide closing remarks.

Chapter Eight: Conclusion

In this thesis, I developed a framework to describe the types of visual interaction cues that video game designers use to guide players through virtual worlds. I then used the framework generatively as a tool to help create new cues for augmented reality guidance. In this concluding chapter, I revisit the thesis questions and contributions presented in Chapter One and provide some closing thoughts.

8.1 Thesis Questions

The primary research problem, "*How should we design visual cues for augmented reality guidance*?", was too vague to be manageable. So, I needed to break that problem down into more practicable parts. The approach I took started with finding a related domain, video games, and learning about how game designers handled the issue in virtual worlds. This led to the development of *TQ1*.

Thesis Question 1 (TQ1): What are the ways video games make use of visual cues to guide players through virtual worlds?

To answer the question, I engaged in an in-depth study of video games, learning about how designers use visualizations to guide players through virtual spaces, and then created a framework to describe visual interaction cues in games. This helped me to see that there were three main properties of visual cues for guidance: the *task* they support, the *markedness* (style) of the cues, and how the cues are *triggered* in the player's view. With this tool in my design toolkit, I was able to quite easily transition into *TQ*2.

Thesis Question 2: How can we learn from visual cues in video games to design new augmented reality tools for guidance in the real world?

This question was addressed in Chapters Five and Six by describing augmented reality cues using the language of the framework, then using the framework as a guide to create new visual interaction cues for AR. Reflection on the process led to Chapter Seven's discussion on design guidelines, rounding out *TQ*2.

Satisfying these thesis questions begins to chip away at the primary research problem. Answering the problem question outright is undoubtedly out of scope for this work and is something we will need to do as a community of researchers and designers over years of iteration. However, taken along with all the other research in the areas of design for guidance, it is a meaningful start.

8.2 Thesis Contributions

This thesis makes the following contributions to the areas of human-computer interaction, visualization, augmented reality, and video game design:

97

Thesis Contribution 1 (TC1): Creation of a framework that describes how visual cues are used in game design to guide players through virtual spaces.

This framework can be used by game designers and researchers who have been missing a shared vocabulary to communicate ideas related to visual interaction cues in games. It may also be useful for application to other domains or for describing other types of cues.

Thesis Contribution 2 (TC2): Demonstration of how the visual interaction cue framework can be used to describe and design new AR tools for spatial guidance in the real world.

This shows that the framework is sufficiently stable to be ported to other domains that make use of cues for guidance. More generally, I hope that my success in applying knowledge from one domain to another related domain might inspire other designers to do the same when they face similar issues.

Thesis Contribution 3 (TC3): The creation of augmented reality prototypes imagining new interactions for AR guidance in the real world.

These prototypes demonstrate a breadth of solutions to what will likely be typical AR tasks in the future, and they offer a starting point for discussions on how to best style cues for specific tasks or needs.

8.3 Closing Remarks

New technologies are continually being crafted, but that does not mean we should "reinvent the wheel" every time we come up with something different. In this thesis work, I exposed a challenge: developing new designs for augmented reality guidance. To address that challenge, I applied knowledge from a related domain: video game navigation and guidance. By reasoning that both AR and video games share the issues of using visualizations to guide people through spaces (virtual or physical), I was able to directly apply techniques from game design to augmented reality. The result was a set of AR designs that gave reasonable coverage of the potential range of solutions. While using video games as inspiration is valid, these designs certainly require more rigorous testing to show whether they are effective for AR. Generalizability, however, is not a goal of this work; for my purposes, the creation and application of the framework were certainly successful. By taking inspiration from games and formalizing the process through the use of the framework, I came up with a fuller breadth of designs than I would otherwise have if I had taken a naïve approach. This framework could also be used by designers creating new visual interaction cues for not only video games and augmented reality, but any medium that has a strong spatial connection between the cue and the environment.

References

- [1] 3D Tours and Virtual Reality on sothebysrealty.com: https://www.sothebysrealty.com/eng/virtual-reality. Accessed: 2018-07-15.
- [2] Amanda
 Frost
 Asset
 Store:

 https://assetstore.unity.com/packages/3d/characters/humanoids/amanda-frost 34583?aid=1011lKhG&utm_source=aff. Accessed: 2018-07-24.
 Store:
- [3] Apple: *https://www.apple.com/*. Accessed: 2018-07-24.
- [4] ARCore Overview | ARCore: https://developers.google.com/ar/discover/. Accessed:
 2018-07-24.
- [5] ARKit Apple Developer: *https://developer.apple.com/arkit/*. Accessed: 2018-07-24.
- [6] Arthur, P. and Passini, R. 1992. *Wayfinding: people, signs, and architecture*.
- [7] Avery, B. et al. 2009. Improving Spatial Perception for Augmented Reality X-Ray Vision. 2009 IEEE Virtual Reality Conference (Mar. 2009), 79–82.
- [8] Avery, B. et al. 2008. User Evaluation of See-through Vision for Mobile Outdoor Augmented Reality. Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (Washington, DC, USA, 2008), 69–72.
- [9] Azuma, R.T. 1997. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*. 6, 4 (1997), 355–385.

- [10] Bane, R. and Hollerer, T. 2004. Interactive tools for virtual x-ray vision in mobile augmented reality. *Third IEEE and ACM International Symposium on Mixed and Augmented Reality* (Nov. 2004), 231–239.
- [11] Bardzell, S. 2008. Systems of Signs and Affordances: Interaction Cues in 3D Games. *EnLeino*, O. (2008), 191–209.
- [12] Billinghurst, M. et al. 2015. A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction*. 8, 2–3 (2015), 73–272.
- [13] Braun, V. and Clarke, V. 2006. Using thematic analysis in psychology. *Qualitative research in psychology*. 3, 2 (2006), 77–101.
- [14] Brudy, F. et al. 2014. Is anyone looking? mitigating shoulder surfing on public displays through awareness and protection. *Proceedings of The International Symposium on Pervasive Displays* (2014), 1.
- [15] Chandler, D. 2001. *Semiotics : The Basics*. Taylor & Francis Group.
- [16] Compass Bar Asset Store: *https://assetstore.unity.com/packages/tools/gui/compass-bar-*7244. Accessed: 2018-07-24.
- [17] Darken, R.P. and Sibert, J.L. 1996. Wayfinding strategies and behaviors in large virtual worlds. *Proceedings of the SIGCHI conference on Human factors in computing systems* (1996), 142–149.

- [18] Feiner, S. et al. 1993. Knowledge-based augmented reality. Communications of the ACM. 36, 7 (1993), 53–62.
- [19] Future of First Response: *http://www.futureoffirstresponse.net/*. Accessed: 2018-07-24.
- [20] Gibson, J.J. 1977. The Theory of Affordances. *Perceiving, acting, and knowing: Toward an ecological psychology*. R. Shaw and J. Bransford, eds. Lawrence Erlbaum Associates. 67–82.
- [21] Google: *https://www.google.com/*. Accessed: 2018-07-24.
- [22] Google Maps: *https://www.google.com/maps*. Accessed: 2018-07-24.
- [23] Grasset, R. et al. 2011. Navigation Techniques in Augmented and Mixed Reality: Crossing the Virtuality Continuum. *Handbook of Augmented Reality*. Springer, New York, NY. 379–407.
- [24] Greenberg, S. and Buxton, B. 2008. Usability evaluation considered harmful (some of the time). *Proceedings of the SIGCHI conference on Human factors in computing systems* (2008), 111–120.
- [25] Home · Aero Glass: *https://glass.aero/*. Accessed: 2018-07-24.
- [26] Iacovides, I. et al. 2015. Removing the HUD: The Impact of Non-Diegetic Game Elements and Expertise on Player Involvement. *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (New York, NY, USA, 2015), 13–22.

- [27] IKEA Place augmented reality app: *https://highlights.ikea.com/2017/ikea-place/*.Accessed: 2018-07-24.
- [28] In-Store Navigation: http://www.lowesinnovationlabs.com/instorenavigation/. Accessed: 2017-09-18.
- [29] Irlitti, A. et al. 2016. Challenges for Asynchronous Collaboration in Augmented Reality. Mixed and Augmented Reality (ISMAR-Adjunct), 2016 IEEE International Symposium on (2016), 31–35.
- [30] Johansen, R. 1989. Groupwise and collaborative systems-a big picture view. *Global Telecommunications Conference and Exhibition'Communications Technology for the 1990s and Beyond'(GLOBECOM), 1989. IEEE (1989), 1217–1220.*
- [31] Jørgensen, K. 2012. Between the Game System and the Fictional World: A Study of Computer Game Interfaces. *Games and Culture*. 7, 2 (Mar. 2012), 142–163.
 DOI:https://doi.org/10.1177/1555412012440315.
- [32] Jørgensen, K. 2010. Time for new terminology. Diegetic and nondiegetic sounds in computer games revisited. Teoksessa Grimshaw, Mark (toim.): Game Sound Technology and Player Interaction: Concepts and Developments. Hershey, PA: Information Science Reference. (2010), 78–97.

- [33] Knierim, P. et al. 2018. Quadcopter-projected in-situ navigation cues for improved location awareness. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), 433.
- [34] Livingston, M.A. et al. 2003. Resolving Multiple Occluded Layers in Augmented Reality. *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality* (Washington, DC, USA, 2003), 56–.
- [35] Lumin: *https://www.dia.org/about/press/media-kits/lumin*. Accessed: 2017-09-18.
- [36] Magic in the Making: *https://www.magicleap.com/*. Accessed: 2018-07-24.
- [37] Microsoft HoloLens: https://www.microsoft.com/en-us/hololens. Accessed: 2018-07-24.
- [38] Milgram, P. et al. 1995. Augmented reality: A class of displays on the realityvirtuality continuum. *Telemanipulator and telepresence technologies* (1995), 282–293.
- [39] Neustaedter, C. and Sengers, P. 2012. Autobiographical design in HCI research: designing and learning through use-it-yourself. *Proceedings of the Designing Interactive Systems Conference* (2012), 514–523.
- [40] Norman, D.A. 1988. The psychology of everyday things.(The design of everyday things).Basic Books.
- [41] Off Screen Indicator Asset Store: https://assetstore.unity.com/packages/tools/gui/offscreen-indicator-57062. Accessed: 2018-07-24.

- [42] Peacocke, M. et al. 2015. Evaluating the effectiveness of HUDs and diegetic ammo displays in first-person shooter games. 2015 IEEE Games Entertainment Media Conference (GEM) (Oct. 2015), 1–8.
- [43] Project Tango: everything you need to know: https://www.techradar.com/news/phoneand-communications/mobile-phones/project-tango-everything-you-need-to-know-1322923.
 Accessed: 2018-07-24.
- [44] Reitmayr, G. and Schmalstieg, D. 2004. Scalable techniques for collaborative outdoor augmented reality. *3rd IEEE and ACM international symposium on mixed and augmented reality (ISMAR'04), Arlington* (2004).
- [45] Salomoni, P. et al. 2016. Assessing the efficacy of a diegetic game interface with Oculus Rift. 2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC) (Jan. 2016), 387–392.
- [46] Sandor, C. et al. 2010. Egocentric space-distorting visualizations for rapid environment exploration in mobile mixed reality. 2010 IEEE Virtual Reality Conference (VR) (Mar. 2010), 47–50.
- [47] Saussure, F. de 1983. Course in General Linguistics. 1916. Trans. Roy Harris. London: Duckworth. (1983).
- [48] Schell, J. 2014. The Art of Game Design: A Book of Lenses, Second Edition. CRC Press.
- [49] Unity: https://unity3d.com. Accessed: 2018-07-24.

- [50] Unity Asset Store The Best Assets for Game Making: *https://assetstore.unity.com/*.Accessed: 2018-07-24.
- [51] Volumetric Light Beam Asset Store: https://assetstore.unity.com/packages/vfx/shaders/volumetric-light-beam-99888.
 Accessed: 2018-07-24.
- [52] Vuforia | Augmented Reality: https://www.vuforia.com/. Accessed: 2018-07-24.
- [53] Windows | Official Site for Microsoft Windows 10 Home & Pro OS, laptops, PCs, tablets & more: *https://www.microsoft.com/en-ca/windows*. Accessed: 2018-07-24.

Ludography

- [L1] 2K Boston and 2K Australia. (2007). *Bioshock*. Game. [Windows]. (21 August 2007).2K Games. Novato, CA, USA
- [L2] Bethesda Game Studios. (2011). *The Elder Scrolls V: Skyrim*. Game. (11 November 2011). Bethesda Softworks. Rockville, MD, USA
- [L3] BioWare Edmonton. (2014). Dragon Age: Inquisition. Game. [PlayStation 4]. (18 November 2014). Electronic Arts. Redwood City, CA, USA
- [L4] Blizzard Entertainment. (2004). World of Warcraft. Game. [Windows]. (23 November 2004). Blizzard Entertainment. Irvine, CA, USA
- [L5] Blizzard Entertainment. (2016). Overwatch. Game. [Windows]. (24 May 2016).Blizzard Entertainment. Irvine, CA, USA
- [L6] Bungie. (2014) *Destiny*. Game. [Playstation 4]. (29 September 2014). Activision. Santa Monica, CA, USA
- [L7] CD Projekt. (2015). The Witcher 3: Wild Hunt. Game. [Windows]. (19 May 2015). CDProjekt RED. Warsaw, Poland
- [L8] Eidos Montreal. (2014) Thief. Game. [Windows]. (February 25, 2014). Square Enix. Tokyo Japan
- [L9] Guerilla Games. (2017). *Horizon Zero Dawn*. Game. [Playstation 4]. (February 28, 2017). Sony Interactive Entertainment. Tokyo, Japan

- [L10] Halfbrick Studios. (2011). *Jetpack Joyride*. Game. [iOS]. (September 1, 2011).Halfbrick Studios. Brisbane, Australia
- [L11] id Software. (1993). Doom. Game. [MS-DOS]. (10 December 1993). GT Interactive. New York City, NY, USA
- [L12] id Software. (2016). Doom. Game. [Windows]. (13 May 2016). Bethesda Softworks.Rockville, MD, USA
- [L13] SUPERHOT Team. (2016). Superhot. Game. [Windows]. (25 February 2016). SUPERHOT Team. Łódź, Poland
- [L14] Team Ico. (2005). *Shadow of the Colossus*. Game. [PlayStation 2]. (18 October 2005).Sony Computer Entertainment. Tokyo, Japan
- [L15] Terrible Toybox. (2017). *Thimbleweed Park*. Game. [Windows]. (30 March 2017).Terrible Toybox. Seattle, WA, USA
- [L16] Ubisoft Annecy. (2016). Steep. Game. [Windows]. (2 December 2016). Ubisoft. Rennes, France
- [L17] Ubisoft Quebec. (2015. Assassin's Creed Syndicate. Game. [Windows]. (19 November 2015). Ubisoft. Rennes, France
- [L18] Valve Corporation. (2009). *Left 4 Dead 2*. Game. [Windows]. (November 17, 2009).Valve Corporation. Bellevue, WA, US

[L19] Visceral Games. (2013). *Dead Space 3*. Game. [Windows]. (5 February 2013).Electronic Arts. Redwood City, CA, USA

Appendix A: Game Corpus

The following pages each include an example of a game used in the sample in the creation of the visual interaction cue framework. Each example includes: A screenshot of the game (showcasing at least one visual cue), a description of the cue and how it fits in the framework, and a ludographic reference. Some games span multiple pages.

Game	Developer	Year	Genre
Doom	id Software	2016	First-Person Shooter
Doom	id Software	1993	First-Person Shooter
The Legend of Zelda: Ocarina of Time	Nintendo	1998	Action-adventure
Rayman Legends	Ubisoft	2013	Platform
Diablo II	Blizzard Entertainment	2000	RPG
Fable III	Lionhead Studios	2010	RPG
Dragon Age: Inquisition	BioWare	2014	RPG
Overwatch	Blizzard Entertainment	2016	First-Person Shooter
The Witness	Thekla, Inc.	2016	Puzzle
Lovers in a Dangerous Spacetime	Asteroid Base	2015	Action
Beyond Eyes	Tiger & Squid	2015	Adventure
Thimbleweed Park	Terrible Toybox	2017	Adventure
Day of the Tentacle Remastered	Double Fine Studios	2016	Adventure
Batman: Arkham Asylum	Rocksteady Studios	2009	Action-adventure
Dark Souls III	FromSoftware	2016	Action RPG
Limbo	Playdead	2010	Platform
De Blob	Blue Tongue Entertainment	2008	Platform
Super Mario Kart	Nintendo EAD	1992	Racing
Mario Kart 8	Nintendo EAD	2014	Racing
Bioshock	2K	2007	First-Person Shooter
Mirror's Edge Catalyst	EA DICE	2016	Platform
Shadow of the Colossus	SCE Japan Studio	2005	Action-adventure
Dead Space 3	Visceral Games	2013	Third-Person Shooter
Left 4 Dead	Valve	2008	First-Person Shooter
Epic Mickey: The Power of Two	Junction Point Studios (et al.)	2012	Platform

Game	Developer	Year	Genre
Mark of the Ninja	Klei Entertainment	2012	Platform
Crash Bandicoot	Naughty Dog	1996	Platform
Rocket League	Psyonix	2015	Racing
Angry Birds	Rovio Entertainment	2009	Puzzle
Jetpack Joyride	Halfbrick Studios	2011	Endless Runner
Horizon: Zero Dawn	Guerilla Games	2017	Action-adventure
Thief	Eidos	2014	Action-adventure
Playerunknown's Battlegrounds	Bluehole	2017	Shooter
Superhot	Superhot Team	2016	First-Person Shooter
Wayward Sky	Uber Entertainment	2016	Puzzle
Burnout	Criterion Games	2001	Racing
The Witcher 3: Wild Hunt	CD Projekt	2015	Action RPG
Steep	Ubisoft	2016	Racing
Forza Horizon 3	Playground Games	2016	Racing
Starfox 64	Nintendo EAD	1997	Rail Shooter
Star Wars: Rogue Squadron	Factor 5, LucasArts	1998	Action
Crimson Skies	Zipper Interactive	2000	Action
Furi	The Game Bakers	2016	Action
Left 4 Dead 2	Valve	2009	First-Person Shooter
Assassin's Creed Syndicate	Ubisoft	2015	Action-adventure
World of Warcraft	Blizzard Entertainment	2004	MMORPG
The Elder Scrolls V: Skyrim	Bethesda Game Studios	2011	Action RPG
Fallout 4	Bethesda Game Studios	2015	Action RPG
The Last Guardian	SIE Japan Studio	2016	Action-adventure



Doom (2016)

- First person shooter game with horror elements

Cues visible in the screenshots

- Left: At the top of the screen, there is a compass that updates based on the orientation of the player's viewport. Next major checkpoint shows on compass at top of screen on HUD (arrow), along with distance to the checkpoint (metric is not clear). Framework labels: Go, Overlaid, Persistent
- Right: "Glory kill visualization" When enemy glows, the player can activate a "special kill" function to destroy the enemy in a very gratuitous way. Here, enemy starts glowing blue if out of range, orange in range. When in range, you can initiate an extreme melee attack. Framework labels: Look, Emphasized, Other Agent-Triggered

Cues Not Visible in the screenshots

- Uses off-screen guidance (as in Playerunknown's Battlegrounds)
- Also: Lots of passive elements (like funneling down a linear path and using lighting to guide)

id Software. (2016). Doom. Game. [Windows]. (13 May 2016). Bethesda Softworks. Rockville, MD, USA



Doom (1993 Original)

- First person shooter game with horror elements

Cues visible in the screenshots

- Left: Level lighting is brighter along ideal path. Gently guides the player along the right path to get through level. Framework labels: Go, Subtle, Persistent
- Right: Slightly altered textures on walls indicate secret treasure areas. Framework labels: Discover, Subtle, Persistent

id Software. (1993). Doom. Game. [MS-DOS]. (10 December 1993). GT Interactive. New York City, NY, USA



The Legend of Zelda: Ocarina of Time

- 3rd person action adventure

Cues visible in the screenshots:

- Center: Yellow triangle crosshair overlaid on target lets player know which enemy is currently being targeted (among many). Framework labels: Look, Integrated, Player-triggered
- Center Top: Yellow circle is Navi, a non-player character agent that also shows nearby the targeted enemy. Navi, can fly to points of interest on her own volition. Framework labels; Look, Subtle, Other Agent-triggered.
- Bottom Left: Minimap shown in bottom of screen shows player location in world, along with enemies and other points of interest. Framework labels: Go, Overlaid, Persistent

Nintendo EAD. (1998). The Legend of Zelda: Ocarina of Time. Game. [Nintendo 64]. (21 November 1998). Nintendo. Kyoto, Japan



Rayman Legends

Cues visible in the screenshots:

- Center left: Agent (shining bright yellow) hovers over point of interest and cues player to the fact that the player can initiate some action to help them. Framework labels: Look, Subtle, Other Agent-triggered

Ubisoft Montpellier and Ubisoft Casablanca. (2013). Rayman Legends. Game. [Windows]. (29 August 2013). Ubisoft. Rennes, France



Diablo II

Cues visible in the screenshots:

- Center to bottom left; Minimap shown overlaid on screen show player location in world, along with enemies and other points of interest. Framework labels: Go, Overlaid, Player-triggered

Blizzard Entertainment. (2000). Diablo II. Game. [Windows]. (29 June 2000). Blizzard Entertainment. Irvine, CA, USA. Blizzard North. San Mateo, CA, USA



Fable III

Cues visible in the screenshots:

- Center: Bright sparkling line along on ground, guiding to next waypoint. Directly guides the player to the next location in the world they must visit. Framework labels: Go, Integrated, Player-triggered.

Lionhead Studios. (2010). Fable III. Game. [Xbox 360]. (26 October 2010). Lionhead Studios. Redmond, WA, USA



Dragon Age: Inquisition

Cues visible in the screenshots:

- Center right of character: Loot (bag and coins shown here) highlights on demand, when player searches local area using a kind of radar. Shows the locations of items player can pick up that are near them. Framework labels: Discover, Emphasized, Player-triggered

BioWare Edmonton. (2014). Dragon Age: Inquisition. Game. [Xbox 360]. (18 November 2014). Electronic Arts. Redwood City, CA, USA



Dragon Age: Inquisition

Cues visible in the screenshots:

- Center top: Green column of smoke shows player the location of next point of interest. Allows player to scan the horizon for next point of interest, guiding without UI elements. Framework labels: Go, Subtle, Persistent

BioWare Edmonton. (2014). Dragon Age: Inquisition. Game. [Xbox 360]. (18 November 2014). Electronic Arts. Redwood City, CA, USA



Overwatch

Cues visible in the screenshots:

- Center: Red marks show around aiming reticle. Lets player know they are being damaged, and roughly where enemy is. Framework labels: Look, Overlaid, Other Agent-triggered

Blizzard Entertainment. (2016). Overwatch. Game. [Windows]. (24 May 2016). Blizzard Entertainment. Irvine, CA, USA



Overwatch

Cues visible in the screenshots:

- Left: Tether drawn between player and teammate. Lets the player keep track of position of teammate with whom they are interacting. Right image the tether is still shown even when teammate leaves the field of view. Framework labels: Look, Integrated, Player-triggered

Blizzard Entertainment. (2016). Overwatch. Game. [Windows]. (24 May 2016). Blizzard Entertainment. Irvine, CA, USA



Overwatch

Cues visible in the screenshots:

- Center: Blue "P" diamond icon with DEFEND label. Label shows important vehicle through walls. Direct navigation by always being able to see the labelled vehicle if looking in that direction. Framework labels: Look, Integrated, Persistent
- Center: An outline of the "payload" vehicle is shown just below the DEFEND label. X-Ray view important vehicle or players through walls. Direct navigation by always being able to move toward location. Framework labels: Look, Emphasized, Persistent
- Left Center, Bottom Center: Blue arrows appear, scroll, and then fade out to indicate that a teammate is performing an ability that is supporting the current player (increasing their attack damage). Framework labels: Look, Overlaid, Other Agent-triggered

Blizzard Entertainment. (2016). Overwatch. Game. [Windows]. (24 May 2016). Blizzard Entertainment. Irvine, CA, USA



The Witness

Cues visible in the screenshots:

- Center bottom running left: Cable lights up after a puzzle is solved. Guides the player to the next puzzle if they follow the cable through the level. Framework labels: Go, Subtle, Other Agent-triggered

Thekla, Inc. (2016). The Witness. Game. [Windows]. (26 January 2016). Thekla, Inc. San Francisco, CA, USA



Lovers in a Dangerous Spacetime

Cues visible in the screenshots:

- Center right: Blue enemy shakes and lights up before firing at the player ship. Lets player know an attack is coming so they can prepare. Framework labels: Look, Emphasized, Other Agent-triggered

Asteroid Base. (2015). Lovers in a Dangerous Spacetime. Game. [Windows]. (9 September 2015). Asteroid Base. Toronto, ON, Canada



Beyond Eyes

Cues visible in the screenshots:

- Parts of the level show clearly in distance in the midst of white fog. Shows player the location of important areas or goals, cueing them to move in a specific direction. Framework labels: Go, Emphasized, Context-triggered

Tiger & Squid. (2015). Beyond Eyes. Game. [Xbox One]. (4 August 2015). Team17. West Yorkshire, United Kingdom



Thimbleweed Park

Cues visible in the screenshots:

- Right center: Text label shows up over objects when hovered over. Tells player which objects are interactable, and what the default action is. Framework labels: Discover, Integrated, Player-triggered

Terrible Toybox. (2017). Thimbleweed Park. Game. [Windows]. (30 March 2017). Terrible Toybox. Seattle, WA, USA



Day of the Tentacle Remastered

Cues visible in the screenshots:

- All interactable items are highlighted in pink when player presses a specific key. Lets the player know where all interactable items are, rather than forcing a scan of all the items. Framework labels: Discover, Emphasized, Player-triggered

Double Fine Productions. (2016). Day of the Tentacle Remastered. Game. [Windows]. (22 March 2016). Double Fine Productions. San Francisco, CA, USA



Batman: Arkham Asylum

Cues visible in the screenshots:

- Special X-Ray view, showing enemies through walls, with different colours for enemies and friendlies. Quick view of enemy type, allows player to see upcoming threats and plan paths. Framework labels: Discover/Look, Emphasized, Player-triggered

Rocksteady Studios. (2009). Batman: Arkham Asylum. Game. [Windows]. (25 August 2009). Eidos Interactive. London, United Kingdom. Warner Bros. Interactive Entertainment. Burbank, CA, USA



Dark Souls III

Cues visible in the screenshots:

- Right: Player can see a ghost (red) of other players before their deaths. Lets the player know where upcoming traps or hazards might be. Framework labels: Discover, Emphasized, Context-triggered
- Objects in the world glow orange if they are interactable (ex. Bottom right sword). Player knows which objects can be interacted with. Framework labels: Discover, Emphasized, Persistent

FromSoftware. (2016). Dark Souls III. Game. [Windows]. (24 March 2016). Bandai Namco Entertainment. Tokyo, Japan


Limbo

Cues visible in the screenshots:

- Edges of the screen blur, and objects/enemies are shown with high contrast. Focuses player on action, and makes objects stand out with high contrast. Framework labels: Discover, Subtle, Persistent

Playdead. (2010). Limbo. Game. [Windows]. (10 July 2010). Playdead. Copenhagen, Denmark. Microsoft Game Studios. Redmond, WA, USA



De Blob

Cues visible in the screenshots:

- Grey environment turns bright on interaction. Objects that still need to be visited stand out to player. Framework labels: Discover, Subtle, Persistent

Blue Tongue Entertainment. (2008). De Blob. Game. [Wii]. (22 September 2008). Agoura Hills, CA, USA



Super Mario Kart

Cues visible in the screenshots:

- Bottom: Minimap in a perspective view shows all the competitors on the track. Player knows where all other enemies are. Can strategize and gauge progress. Framework labels: Look, Overlaid, Persistent

Nintendo EAD. (1992). Super Mario Kart. Game. [Super NES]. (27 August 1992). Nintendo. Kyoto, Japan



Mario Kart 8

Cues visible in the screenshots:

- Left: Red shell icon bordered with white circle shows at bottom of screen when someone has attacked. Player knows a threat is coming, and they should try to avoid it. Gives some information about where the threat is coming from. Framework labels: Look, Overlaid, Other Agent-triggered

Nintendo EAD. (2014). Mario Kart 8. Game. [Wii U]. (29 May 2014). Nintendo. Kyoto, Japan



Bioshock

Cues visible in the screenshots:

- Level lighting is brighter along ideal path. Guides the player along the right path to get through level. Framework labels: Go, Subtle, Persistent

2K Boston and 2K Australia. (2007). Bioshock. Game. [Windows]. (21 August 2007). 2K Games. Novato, CA, USA



Mirror's Edge Catalyst

Cues visible in the screenshots:

- Center: Intractable objects show as coloured (red in this image) among white/grey environment. Lets player know what objects can be interacted with. Framework labels: Discover, Emphasized, Context-triggered
- Center to center left: Glowing red line (made up of triangles) shown along a path. Shows player where they can/should move to progress through the level. Framework labels: Go, Integrated, Context-triggered

EA DICE. (2016). Mirror's Edge Catalyst. Game. [Windows]. (7 June 2016). Electronic Arts. Redwood City, CA, USA



Shadow of the Colossus

Cues visible in the screenshots:

- Glint on the player's sword points toward next enemy. Keeps player moving in the right direction so long as they continue checking. Framework labels: Go, Subtle, Player-triggered

SCE Japan Studio. (2005). Shadow of the Colossus. Game. [PlayStation 2]. (18 October 2005). Sony Computer Entertainment. Tokyo, Japan



Dead Space 3

Cues visible in the screenshots:

- Icon (top right) marking teammate position transitions between in and out of viewing angle. Less abrupt change makes it easier to follow icons marking position of point of interest. Framework labels: Look, Overlaid/Integrated, Context-triggered

Visceral Games. (2013). Dead Space 3. Game. [Windows]. (5 February 2013). Electronic Arts. Redwood City, CA, USA



Left 4 Dead

Cues visible in the screenshots:

- Level lighting is brighter along ideal path. Guides the player along the right path to get through level. Framework labels: Go, Subtle, Persistent

Valve South. (2008). Left 4 Dead. Game. [Windows]. (17 November 2008). Bellevue, WA, USA



Epic Mickey 2: The Power of Two

Cues visible in the screenshots:

- X-Ray style outline shows where player should "paint". Contrast with existing world objects shows where player can interact. Framework labels: Discover, Integrated, Persistent

Junction Point Studios, Blitz Games Studios, Heavy Iron Studios. (2012). Epic Mickey 2: The Power of Two

. Game. [Windows]. (18 November 2012). Disney Interactive Studios. Glendale, CA, USA



Mark of the Ninja

Cues visible in the screenshots:

- Enemy vision is shown as a bright cone. Lets player know where they shouldn't go, so they keep from being spotted. Framework labels: Look, Integrated, Persistent
- Enemy laser sight shown when they are targeting. Player knows they are being targeted, and knows to avoid laser sight. Framework labels: Look, Subtle, Other Agent-triggered
- Icon shows above enemy when they spot player. Player is aware of which enemies have spotted them, and which ones to avoid. Framework labels: Look, Integrated, Other Agent-triggered

Cues not visible in the screenshots:

- AI path is exposed to the player. Player knows what move enemy will take, and can plan their route accordingly. Framework labels: Discover, Integrated, Other agent-triggered

Klei Entertainment. (2012). Mark of the Ninja. Game. [Windows]. (7 September 2012). Klei Entertainment. Vancouver, BC, Canada. Microsoft Studios. Redmond, WA, USA



Crash Bandicoot

Cues visible in the screenshots:

- Level lighting is brighter along ideal path. Developers use a "squint test" (discussed in "Level Up" by Rogers). Gently guides the player along the right path to get through level. Framework Labels: Go, Subtle, Persistent

Naughty Dog. (1996). Crash Bandicoot. Game. [PlayStation]. (9 September 1996). Sony Computer Entertainment. Tokyo, Japan

Rogers, Scott. Level Up! The guide to great video game design. John Wiley & Sons, 2014.



Rocket League

Cues visible in the screenshot:

- Arrow points (in 3D) to where the ball is (it's out of the top of the screen). Framework labels: Go, Integrated, Other Agent-triggered
- Circles always appear below the ball (and the size of inside circle is smaller when ball is high off the ground). Framework labels: Go, Integrated, Other Agent-triggered
- Goal to score on has animated white boarder (inset). Framework labels: Go, Emphasized, Persistent

Cues not visible in the screenshot:

- When the ball is far away, it is outlined in white to make it more visible. Framework labels: Go, Emphasized, Context-triggered

Psyonix. (2015). Rocket League. Game. [Windows]. (July 7, 2015). Psyonix. San Diego, CA, US



Angry Birds

Cues visible in the screenshot:

- Path of last shot is shown as dotted white line. When aiming, can correct by using past path as a guide. Framework labels: Look, Integrated, Context-triggered
- (note the little cloud, which also shows where the last bird's power-up was triggered)

Rovio Entertainment. (2009). Angry Birds. Game. [iOS]. (December 11, 2009). Chillingo. Macclesfield, UK



Jetpack Joyride

- Infinite runner game

Cues visible in the screenshot:

- Danger indicator comes up to let the player know some obstacle is about to enter screen. Framework labels: Look, Overlaid, Context-triggered

Halfbrick Studios. (2011). Jetpack Joyride. Game. [iOS]. (September 1, 2011). Halfbrick Studios. Brisbane, Australia



Horizon: Zero Dawn

- Main character actually has an AR device as part of the story.

Cues visible in the screenshot:

- Enemy AI paths show as arrows along ground when in AR view. Framework labels: Discover, Integrated, Player-triggered

Cues not visible in the screenshot:

- Icons show at edge of screen when object being tracked is out of player's field of view. (Ex. Icon above the enemy.) Framework labels: Go, Overlaid, Other Agent-triggered (the game chooses which icons to show and when)

Guerilla Games. (2017). Horizon Zero Dawn. Game. [Playstation 4]. (February 28, 2017). Sony Interactive Entertainment. Tokyo, Japan



Horizon: Zero Dawn

Cues visible in the screenshot:

- Stake and icon on enemy corpse to indicate that it can be looted. Framework labels: Discover, Integrated, Other Agent-triggered
- Points of interest and/or importance are shown on compass at top-center of screen. Framework labels: Go, Overlaid, Persistent

Guerilla Games. (2017). Horizon Zero Dawn. Game. [Playstation 4]. (February 28, 2017). Sony Interactive Entertainment. Tokyo, Japan



Thief

Cues visible in the screenshot:

- Loot shows as highlighted when close enough to grab it. Framework labels: Discover, Emphasized, Context-triggered
- Labels show over loot. Framework labels: Discover, Integrated, Context-triggered

Eidos Montreal. (2014) Thief. Game. [Windows]. (February 25, 2014). Square Enix. Tokyo Japan



Playerunknown's Battlegrounds

Cues visible in the screenshot:

- Points of interest and/or importance are shown on compass at top-center of screen (front). Framework labels: Go, Overlaid, Persistent
- Icons show at edge of screen when object being tracked is out of player's field of view. Framework labels: Go, Overlaid, Context-triggered
- Icons show above teammate's avatar when in view. Framework labels: Go, Integrated, Persistent
- Full screen map with grid to mark locations or points of interest for teammates (behind). Framework labels: Go, Overlaid, Player-triggered

Bluehole. (2017). PlayerUnknown's Battlegrounds. Game. [Windows]. (Early Access (2017)). Bluehole. Bundang-gu, South Korea



Superhot

Cues visible in the screenshot:

- Bullet trails 3D trails that can show player where enemy fire is coming from. Framework labels: Look, Emphasized, Other Agent-triggered
- Bright muzzle flash when the enemies shoot, letting player know they need to dodge. Framework labels: Look, Subtle, Other Agent-triggered

Superhot Team. (2016). Superhot. Game. [Windows]. (February 25, 2016). Superhot Team. Łódź, Poland



Wayward Sky (PS VR)

- Developers call it a "look and click VR adventure"

Cues visible in the screenshot:

- Interactable elements are highlighted with a white stroke when cursor is over them. Framework labels: Discover, Emphasized, Context-triggered

Uber Entertainment. (2016). Wayward Sky. Game. [Playstation 4]. (October 10, 2016). Uber Entertainment. Kirkland, WA, US



Wayward Sky (PS VR)

Cues visible in the screenshot:

- Arrow is overlaid on ground plane to guide player. Framework labels: Discover, Integrated, Other Agent-triggered

Uber Entertainment. (2016). Wayward Sky. Game. [Playstation 4]. (October 10, 2016). Uber Entertainment. Kirkland, WA, US



Burnout

Cues visible in the screenshot:

- 2 phase turn prediction (first just green arrow, then turn sign). Framework labels: Go, Overlaid, Context-triggered

Criterion Games. (2001). Burnout. Game. [Playstation 2]. (November 1, 2001). Acclaim Entertainment. Glen Cove, NY, US



The Witcher 3: Wild Hunt

Cues visible in the screenshot:

- Door is highlighted, showing how to progress through the level. Framework labels: Go, Emphasized, Player-triggered

CD Projekt. (2015). The Witcher 3: Wild Hunt. Game. [Windows]. (19 May 2015). CD Projekt RED. Warsaw, Poland



Steep

Cues visible in the screenshot:

- Checkpoints show as pillars of light (with orange being the next one). Framework labels: Go, Integrated, Context-triggered
- Icons show around screen indicating positions of other characters in the world. Framework labels: Look, Overlaid, Context-triggered

Ubisoft Annecy. (2016). Steep. Game. [Windows]. (December 2, 2016). Ubisoft. Rennes, France



Steep

Cues visible in the screenshot:

- An ideal path through the world is represented as a spotted white line. Framework labels: Go, Integrated, Persistent

Ubisoft Annecy. (2016). Steep. Game. [Windows]. (December 2, 2016). Ubisoft. Rennes, France



Forza Horizon 3

Cues visible in the screenshot:

- This is a "braking line", which shows you where to be, and turns yellow/red to cue you to slow down to take turns. Framework labels: Go, Integrated, Context-triggered
- 3D minimap shows routes through the course. Framework labels: Go, Overlaid, Persistent
- Checkpoints are shown as labels in world with "stakes" in the ground. Framework labels: Go, Integrated, Context-triggered

Playground Games. (2016). Forza Horizon 3. Game. [Windows]. (September 27, 2016). Microsoft Studios. Redmond, WA, US



Starfox 64

Cues visible in the screenshot:

- Off-screen target shows as a "help" arrow, pointing off screen. 2 arrows point inward when target is on screen. Framework labels: Look, Overlaid, Other Agent-triggered
- When on screen, target has a large coloured marker on it. Framework labels: Look, Integrated, Other Agent-triggered (some external event would trigger the indicator)

Nintendo EAD. (1997). Star Fox 64. Game. [Nintendo 64]. (June 30, 1997). Nintendo. Kyoto, Japan



Star Wars: Rogue Squadron

Cues visible in the screenshot:

- Mini map shows an orientation cone guiding to next checkpoint. Framework labels: Go, Overlaid, Persistent

Factor 5, LucasArts. (1998). Star Wars: Rogue Squadron. Game. [Windows]. (December 3, 1998). LucasArts. San Francisco, CA, US



Crimson Skies

Cues visible in the screenshot:

- Off-screen target is shown as a live image of the actual target in the world. Framework labels: Go, Overlaid, Context-triggered
- Target indicator turns into a thin blue outline when it enters the field of view. Framework labels: Go, Integrated, Context-triggered

Zipper Interactive. (2000). Crimson Skies. Game. [Windows]. (September 17, 2000). Microsoft. Redmond, WA, US



Furi

Cues visible in the screenshot:

- Area of effect of upcoming attack is painted on the ground. Framework labels: Look, Integrated, Other Agent-triggered

The Game Bakers. (2016). Furi. Game. [Windows]. (July 5, 2016). The Game Bakers. Montpellier, France



Left 4 Dead 2

Cues visible in the screenshot:

- Character outlines show as different colours based on health needs. Framework labels: Look, Emphasized, Other Agent-triggered

Valve Corporation. (2009). Left 4 Dead 2. Game. [Windows]. (November 17, 2009). Valve Corporation. Bellevue, WA, US



Assassin's Creed Syndicate

Cues visible in the screenshot:

- When you've been spotted by an enemy, this ring visualization comes up. Colour (amount of red) seems to indicate some threat level. Framework labels: Look, Integrated, Other Agent-triggered

Ubisoft Quebec. (2015). Assassin's Creed Syndicate. Game. [Playstation 4]. (October 23, 2015). Ubisoft. Rennes, France



Assassin's Creed Syndicate

Cues visible in the screenshot:

- Eagle vision: Lets you see enemies and objects through walls. Enemies show red, friendlies blue, objects white. Framework labels: Discover, Emphasized, Player-triggered

Ubisoft Quebec. (2015). Assassin's Creed Syndicate. Game. [Playstation 4]. (October 23, 2015). Ubisoft. Rennes, France



Blizzard Entertainment. (2004). World of Warcraft. Game. [Windows]. (November 23, 2004). Blizzard Entertainment. Irvine, CA, US

- Image From: http://wowwiki.wikia.com/wiki/Minimap



The Elder Scrolls V: Skyrim

Cues visible in the screenshot:

- Points of interest and/or importance are shown on compass at top-center of screen. Framework labels: Go, Overlaid, Persistent

Bethesda Game Studios. (2011). The Elder Scrolls V: Skyrim. Game. [Windows]. (November 11, 2011). Bethesda Softworks. Rockyville, MD, US


Fallout 4

Cues visible in the screenshot:

- Bright green line drawn to next objective in the world in VATS view. Framework labels: Go, Integrated, Player-triggered.

Bethesda Game Studios. (2015). Fallout 4. Game. [Windows]. (November 10, 2015). Bethesda Softworks. Rockyville, MD, US



The Last Guardian

Cues visible in the screenshot:

- Barrels that can be interacted with are decorated with white/blue light. Framework labels: Discover, Subtle, Persistent

SIE Japan Studio. (2016). The Last Guardian. Game. [Playstation 4]. (December 6, 2016). Sony Interactive Entertainment. Tokyo, Japan

Appendix B: Thematic Analysis Raw Data

The following spreadsheet provides an example of how the game examples were analyzed and grouped. The groupings are not the final groupings, but this gives a snapshot of the process of categorizing elements of cues. This is a large file, so the first image shows the entire spreadsheet, with subsequent images zooming in on sections of the document.

Catalogue of Game Design Things

Mail You yuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu		Anariy Concept Comparing and the set was in the down integrand Anary Concept Anar	Galaxy Bit Angle Main Yang, Yang Yang, Yang Yang Galaxy Galaxy Distry Nan Yang Yang, Yang Yang, Yang Yang Galaxy Galaxy Galaxy Distry Nan Yang Yang, Yang Yang Galaxy	Sector Sector Sector Sect	Observation Operation of the second of the sec	State State State State State Sta
"Andd a coutor example? Ro Httern	design ender the ender of the e	Constantiation of experiments in the original sector of the original	Scholarski kan serie gjudiner konduktion konduci 0. 740 Scholarski kan serie gjudiner konduktion konduci 0.00 740 Scholarski kan serie kan seri	test page transmission in transmission of test page test page transmission of test page transmission of test page tra	Table bit is for any intervention Building to the set of th	
rang and an and an and an	And And Any Kategorated South Chargorated South	rausic 200 rausic	An and a sector of the sector	ay 30 ay 30 asis 30 asis 30 ay 30 ay 30 ay 30 30 30 30 30 30 30 30 30 30 30 30 30 3	pluoonnes 200 avit 300 be de 300 be	Pa
Spatial Nond agets Spatial Spatial Nond agets Nond agets Spatial	Kondigeo, Kondigeo, Dispetic Dispetic Spatial Spatial Dispetic Dispetic	Spatial Spatial Digget Spatial Spatial Digget Spatial Cinget Nonol spatial Nonol spatial Nonol spatial Nonol spatial	Constant Con	Nond quick Spatial Nond quick Nond quick	3D Diege si Spatial Dieget al Dieget al Nondiget Nondiget Nondiget Nondiget Nondiget Nondiget	perties
Contastual Persi Sterit Persi Sterit Persi Sterit Contastual Contastual	Physein initia Contactual Contactual Contactual Contactual Contactual Contactual	Presi Szert Cortastud Cortastud Cortastud "unicowe "unicowe Presi Szert Cortastud Cortastud Cortastud Cortastud Cortastud Cortastud Cortastud Cortastud	Corrabual Corrabual Corrabual Corrabual Physicing Physicing Physicing Physicing Physicing Physicing Physicing Physicing Corrabual Corrab	Contactual Contactual Contactual Persi storit Contactual	Displaying Contastual President Contastual Contastual Contastual Contastual Contastual President Contastual President President President President Contastual President Preside	
*check *check In In In/Dut In/Dut		in out out in out	n n n n n n n n n n n n n n n n n n n n	Infort Occ Inandod	INDOR IN NOR IN NOR IN NOR	
Pertadu Une Une	Predect I hold and "card (Guide province)" that is address to be "chear the	"Hadrink by Down (P). The Unit Provide Area *1.20 April 2019 for particular *1.20 April 2019 for particular *2.20 April 2019 for particular Complete Cold April 1910 for particular Com Vignet (1914 for the Theory Ind Stee particular Com Vignet (1914 for the Theory Ind Stee particular Stee Particular Com Vignet (1914 for the Theory Ind Stee particular Stee Particular Stee Particular ************************************	The the construction to the Lag Construction of the technology of technology of the technology of technology of the technology of technolo	Stoon whether every hind war rot. Proc. Net Stood gene biggend. Technically teammate, but the player trince hind of gene-biggend. Technically teammate, but the player trince bind of gene-biggend. Technically teammate, but they avenue team hind of gene-biggend. Technically team avenue, but they are transmission hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene-biggend. Technically team avenue, but they are team hind of gene biggend. Technically	Control Contro Control Control Control Control Control Co	* Georder Ex Connor This make
dure		~		ens on services to the HLD & a pulking attent on the s all he between players a tool ing it, and it's the response to an ever purale is available to play (Otheri wae ins purale is available to play (Otheri wae ins	v(Yibika S) in chuna or Nghi (kood) ghang Li a ad ume) ant on colo	y;2 things – tanget (the thing, we're navig vigate to a contain "point", by using a "line s things a bit muddy
the a partemple of user of users codes or object. Poindus dout & to date Hold yord (Doubly 2 Allowed they) unredening)	na vysi firke teer ym Och jidir e rennin)	stare - board reners ware first a breed agent is transmission. The scalar possible path processors, l'heady.) Ye watation models the whenders, where explore bob waterion materials basis out not belief in the scalar possible the whenders, where explore the scalar possible out not belief in the scalar possible the scalar possible out of the scalar possible out of the scalar possible out of the scalar possible out of the scal		Constraints place many management of each part of the stars of place in the stars of the sta	tions Gan all Consensionation reviews a neurol Teleperior of the service of the	Prop. Prop. e.* * Pathu, Dezinatori Rutra'i herezî Dişecci i (neresî) Tikl
		under y pirt ("e, unte d'Orek)		nnna cuquer carrier x xono non oz le guldeg atterion'if i want toj attr. (local on sc enented) – Hgh	 Providence2 "Agent cauld "Lonch leave (Inwork geometry makes alter of s guide to arrest/n 	perties: ex. Penásterit vs. Summoned, slíhá to table 2 in spore wwy le 3 is about 49 stuff — mos ty pomi

Catalogue of Game Design Things

Game	Slide #	What's happening?	What does it do for me?	Disc/Aim/Nav	Conspicuousness
Doom (2016)		1 Compass at top of HUD	Tells me where next waypoint is	Go	Overlay
Doom (2016)		1 Enemy highlights when glory kill is possible	Tells player an object is in interactable state, and if they're close enough to i	r Look	Emphasis
Doom (Original)		2 Level lighting is brighter along ideal path	Gently guides the player along the right path to get through level	Go	Invisible
Zelda: Ocarina of Time		3 Crosshair overlaid on target	Lets player know which enemy is currently being targeted (among many).	Look	Integrated
Zelda: Ocarina of Time		3 Agent hovers over point of interest	Makes it clear that something important is close to player, and they should s	Look	Invisible
Zelda: Ocarina of Time		3 Minimap shown in bottom of screen	Shows me my location in world, along with enemies and other POIs	Go	Overlay
Rayman: Legends		4 Agent hovers over point of interest	Cues player to the fact that an agent can initiate some action to help them	Look	Invisible
Diablo 2		5 Minimap shown overlaid on screen	Shows me my location in world, along with enemies and other POIs	Go	Overlay
Fable 3		6 Line drawn on ground, guiding to next waypoint	Directly guides the player to the next location in the world they must visit	Go	Integrated
Dragon Age: Inquisition		7 Loot highlights on demand, when player searches local area using a kind of radar	Shows the locations of items player can pick up that are near them	Discover	Emphasis
Dragon Age: Inquisition		8 Column of smoke shows player the location of next POI	Allows player to scan the horizon for next POI, guiding without UI elements	Go	Invisible
Overwatch		9 Red marks show around reticle. (360 deg)	Lets me know if player is being damaged, and roughly where enemy is	Look	Overlay
Overwatch		11 Red X shows on reticle when enemy is hit.	Visual feedback		
Overwatch		12 Tether drawn between player and teammate	Lets the player keep track of position of teammate with whom they are inter-	r: Look	Emphasis
Overwatch		13 X-Ray view of players or vehicles through walls.	Quickly distinguish between player/vehicle types using outline. Direct naviga	t Look	Emphasis
Overwatch		14 Icons show around screen when being boosted or healed	Lets the player know that a teammate is acting on them in some way	Look	Overlay
The Witness		15 Cable lights up after a puzzle is solved	Guides the player to the next puzzle if they follow the cable through the leve	l Go	(Invisible or Emphasis)
Lovers in a Dangerous Spacetime		16 Camera pans to show an unlocked object	Lets the player know where to go after the unlock action is completed	Go	*unclear (something different? Ma
Lovers in a Dangerous Spacetime		17 Enemy shakes a lot before it fires at the player	Lets the player know that an attack is coming, so they can prepare	Look	Emphasis
Beyond Eyes		18 Parts of the level show clearly in distance among fog of war	Shows player the location of important areas or goals, cueing them to move	Go	Emphasis
Doom (Original)		19 Textures on wall are slightly different	Indicates secret rooms with which the player can interact	Discover	Invisible
Thimbleweed Park	;	20 Text label shows up over objects when hovered over	Tells player which objects are interactable, and what the default action is	Discover	Integrated
Day of the Tentacle (Remake)	:	21 All interactable items are highlighted in pink when player presses a specific key	Lets the player know where all interactable items are, rather than forcing a	s Discover	Emphasis
Batman: Arkham Asylum		22 Special X-Ray view, showing enemies through walls, with different colours for ener	r Quick view of enemy type, allows player to see upcoming threats and plan p	aDiscover	Emphasis
Dark Souls 3		23 Player can see a ghost of other players before their deaths	Lets the player know where upcoming traps or hazards might be. (Take cauti	Discover	Emphasis
Dark Souls 3	:	23 Objects in the world glow orange if they are interactable.	Player knows which objects can be interacted with (without doing a big sear	c Discover	Emphasis
Limbo		24 Edges of screen blur, and objects/enemies are high contrast.	Focuses player on action, and makes objects stand out with high contrast.	Discover	Invisible
Splatoon		25 Paint is splattered over a grey environment.	Objects that still need to be painted stand out to player.	Discover	Invisible
De Blob	:	26 Grey environment turns bright on interaction.	Objects that still need to be visited stand out to player.	Discover	Invisible
Super Mario Kart	:	27 Minimap in a perspective view shows all the competitors on the track.	Player knows where all other enemies are. Can strategize and gauge progres	: Look	Overlay
Mario Kart 8	:	28 Icon shows at bottom of screen when someone has attacked.	Player knows a threat is coming, and they should try to avoid it. Gives a bit o	f Look	Overlay
Bioshock	:	29 Level lighting is brighter along ideal path	Gently guides the player along the right path to get through level.	Go	Invisible
Mirror's Edge		30 Interactable objects show as coloured among white/grey environment.	Lets player know what objects can be interacted with, and seems to guide a	Discover	Emphasis
Mirror's Edge		30 Glowing line or running "ghost" figure shown along a path.	Shows player where they can/should move to progress through the level.	Go	Integrated
Shadow of the Colossus		31 Glint on the player's sword will point toward next enemy.	Always keeps player moving in the right direction. Player must continue cheep	lGo	Invisible
Dead Space 3		32 *diegetic hud elements		N/A	
Dead Space 3		33 Icon marking teammate position transitions between in and out of viewing angle.	Less abrupt change makes it easier to follow icons marking position of POI.	Look	Overlay/Integrated

Game	Slide #	What's happening?	What does it do for me?	Disc/Aim/Nav	Conspicuousness
NHL 17	3	*puck visualization for shot		N/A	
Epic Mickey	3	5 X-Ray style outline shows where player should "paint".	Contrast with existing world objects shows where player can interact.	Discover	Emphasis
Mark of the Ninja	3	Enemy vision is shown as a bright cone.	Lets player know where they shouldn't go, so they keep from being spotted.	Look	Integrated
Mark of the Ninja	3	⁷ Enemy laser sight shown when they are targeting.	Player knows they are being targeted, and knows to avoid laser sight.	Look	Invisible
Mark of the Ninja	3	Icon shows above enemy when they spot player.	Player is aware of which enemies have spotted them, and which ones to avoi	Look	Integrated
Mark of the Ninja	3	Guard AI path is exposed to the player.	Player knows what move enemy will take, and can plan their route according	Discover	Integrated
Mark of the Ninja	3	Position where you were last spotted is shown as a ghost (Dark Souls style)	Gives player their history, so they can change their tactics.	*unknown	
Crash Bandicoot	3	8 Level lighting is brighter along ideal path. Developers use a "squint test".	Gently guides the player along the right path to get through level.	Go	Invisible
Rocket League	3	Rings on the field with size based on distance when ball is off the screen (may	t Gives player an idea of how high the ball has bounced, so they can predict wh	Go	Integrated
Rocket League	3	3D arrow points to the ball when it is off screen.	Gives a player an idea of the position of the ball when it's out of view.	Go	Integrated
Rocket League	4	Motion lines around the goal. These are white animated lines.	Makes the goal stand out in the play field.	Go	Emphasis
Angry Birds	4	Previous shot's trajectory is shown, with a small indicator on the position when	ELets the player plan the next shot based on the performance of the previous of	Go	Integrated
Jetpack Joyride	4	? Warning shows as an icon at edge of screen when rockets are coming. Change	Shows player the y-axis position of something they'll need to avoid, with som	Look	Overlay
Horizon: Zero Dawn	4	Enemy patrol paths are shown as arrows drawn on the ground.	Player knows what move enemy will take, and can plan their route according	Discover	Integrated
Horizon: Zero Dawn	4	*OSG		Look	Overlay/Integrated
Horizon: Zero Dawn	4	Compass is shown at top of HUD, with POIs displayed on top of it.	Lets player set direction to get to a specific POI.	Go	Overlay
Horizon: Zero Dawn	4	I cons show over POIs, with lines attaching them (stakes) to the POI as an object	The stake on the icon makes it easier to see in the 3D world which game obje	Discover	Integrated
Horizon: Zero Dawn					
Assassin's Creed (various)	4	Enemy patrol paths are shown as lines drawn on the ground.	Player knows what move enemy will take, and can plan their route according	Discover	Integrated
Battlegrounds	4	Player can bring up a large map overlay to set and view POIs on a grid. A small	Player navigates the world as they would with an actual map.	Go	Overlay
Battlegrounds	4	5 *OSG		Look	Overlay/Integrated
Battlegrounds					
Superhot	4	Trails are drawn from enemy bullets back to their guns.	Player can localize the point of the threat by following the trail back to the so	Look	(Invisible or Emphasis)
Superhot	4	Bright white muzzle flash shows when enemies fire their weapons.	Player is made aware of the position of an imminent threat.	Look	Invisible
Wayward Sky	4	Items that are interactable are highlighted when they are hovered over.	Indicates that the player can perform some action with this object.	Discover	Emphasis
Wayward Sky	4	*icon on plane to guide forward		Discover	Integrated
Robo Recall					
Burnout					
Burnout	5	*turn prediction (2 phase)		Go	Overlay
Final Fantasy XV	5	*camera forces view of POI		Go	*unclear
Skyrim					
Steep	5	*checkpoints as pillars (and volumes)		Go	Integrated
Steep	5	*OSG		Look	Overlay/Integrated
Steep	5	*drawing a path to follow		Go	Integrated
Forza Horizon (3?)	5	*braking path	*extra high nav (tells you location and ideal speed)	Go	Integrated
Forza Horizon (3?)	5	*perspective minimap		Go	Overlay
Starfox 64	5	*OSG (bad osg)		Look	Overlay/Integrated
Star Wars: Rogue Squadron		*perspective crosshairs		N/A	
Star Wars: Rogue Squadron	5	*minimap cone		Go	Overlay
Crimson Skies	5	*video OSG		Go	Overlay/Integrated
Furi	6	*area of effect warning for attacks		Look	Integrated
Left 4 Dead 2	6	*teammate outline based on health		Look	Emphasis
Assassin's Creed: Syndicate	6	*ring pointing to enemy after you've been exposed		Look	Integrated
Assassin's Creed: Syndicate	6	*see through walls in eagle vision		Discover	Emphasis
World of Warcraft	6	*minimap shows objects of different types in the world		Discover	Overlay
Skyrim	6	*compass at top of HUD		Go	Overlav

	Properti	es			
Game	2D/3D	Diegesis	Displaying	In/Out/Oc	c Property Notes
Doom (2016)	3D	Non-diegetic	Contextual	In and Out	
Doom (2016)	3D	Spatial	Contextual	In	
Doom (Original)	3D	Diegetic	Persistent	In	
Zelda: Ocarina of Time	3D	Spatial	Contextual	In	Not sure whether game- or player- triggered. (Define better.)
Zelda: Ocarina of Time	3D	Diegetic	Contextual	In	Fairy automatically hovers over things that are important.
Zelda: Ocarina of Time	3D	Non-diegetic	Persistent	In/Out	Minimap is weird for View (in/out/etc.), but it gives info about things in and out of the viewing angle usually. (Cound be interesting
Rayman: Legends	2D	Diegetic	Contextual	In	
Diablo 2	2D	Non-diegetic	Player-initiated	In/Out	Shows locations of POIs both on and off screen. Kind of isometric view instead of proper 2/3D.
Fable 3	3D	Diegetic	Contextual	In	I *think* it's player triggered.
Dragon Age: Inquisition	3D	Spatial	Player-initiated	In	May also be off-screen. // check this
Dragon Age: Inquisition	3D	Diegetic	Persistent	In	
Overwatch	3D	Non-diegetic	Contextual	In/Out	Shows whether enemy in in view or not.
Overwatch					
Overwatch	3D	Diegetic	Contextual	In/Out	
Overwatch	3D	Spatial	Persistent	Occ	
Overwatch	3D	Non-diegetic	Contextual	In and Out	Kind of game-triggered. Technically teammate, but the player isn't controlling it, and it's the response to an event in the game.
The Witness	3D	Diegetic	Contextual	In	Kind of game-triggered. Cable always shows, but lights when the next puzzle is available to play. (Otheriwse inactive)
Lovers in a Dangerous Spacetime	2D	Diegetic	Contextual	Out	Hmm these are harder to class. (Come back to it.)
Lovers in a Dangerous Spacetime	2D	Diegetic	Contextual	In	
Beyond Eyes	3D	Diegetic	Contextual	In	I think the game exposes certain areas as you explore (not persistent)
Doom (Original)	3D	Diegetic	Persistent	In	
Thimbleweed Park	2D	Spatial	Contextual	In	
Day of the Tentacle (Remake)	2D	Spatial	Player-initiated	In	
Batman: Arkham Asylum	3D	Diegetic	Player-initiated	check	*I think this is mostly Occlusion.
Dark Souls 3	3D	Diegetic	Contextual	In	
Dark Souls 3	3D	Spatial	Persistent	In	
Limbo	2D	Diegetic	Persistent	In	
Splatoon	3D	Diegetic	Persistent	In	
De Blob	3D	Diegetic	Persistent	In	
Super Mario Kart	3D	Non-diegetic	Persistent	In/Out	
Mario Kart 8	3D	Non-diegetic	Contextual	Out	
Bioshock	3D	Diegetic	Persistent	In	
Mirror's Edge	3D	Diegetic	Persistent	In	Diegetic, like Horizon: Zero Dawn. But doesn't need to be.
Mirror's Edge	3D	Diegetic	Contextual	In	Diegetic, like Horizon: Zero Dawn. But doesn't need to be.
Shadow of the Colossus	3D	Diegetic	Player-initiated	In and Out	
Dead Space 3	3D				
Dead Space 3	3D	Non-diegetic	Contextual	Trans	
Left 4 Dead	3D	Diegetic	Persistent	In	

	Properti	es			
Game	2D/3D	Diegesis	Displaying	In/Out/Oo	cc Property Notes
NHL 17	3D				
Epic Mickey	3D	Spatial	Persistent	In	*Spatial is tricky, because it's 2D. But I think it's the 2D equivalent of spatial.
Mark of the Ninja	2D	Spatial	Contextual	In	
Mark of the Ninja	2D	Diegetic	Contextual	In	
Mark of the Ninja	2D	Spatial	Contextual		
Mark of the Ninja	2D	Spatial	*unknown		
Mark of the Ninja	2D	Spatial	*unknown		
Crash Bandicoot	3D	Diegetic	Persistent	In	
Rocket League	3D	Spatial	Contextual	Out	*is it only out? Not persistent?
Rocket League	3D	Spatial	Contextual		* could be in and out at least out
Rocket League	3D	Diegetic	Persistent	In	
Angry Birds	2D	Spatial	Contextual	In	Game-triggered, since it only shows last shot?
Jetpack Joyride	2D	Non-diegetic	Contextual	Out/Trans	
Horizon: Zero Dawn	3D	Diegetic	Player-initiated	In	Diegetic, but doesn't need to be.
Horizon: Zero Dawn	3D	Non-diegetic	Contextual	*check	
Horizon: Zero Dawn	3D	Non-diegetic	Contextual	In/Out	
Horizon: Zero Dawn	3D	Spatial	Contextual		Game-triggered, since icons change? And show up when enemies die?
Horizon: Zero Dawn	3D				
Assassin's Creed (various)	3D	Diegetic	Player-initiated	In	
Battlegrounds	3D	Non-diegetic	Player-initiated	In/Out	
Battlegrounds	3D	Non-diegetic	Contextual		Persistent? In/out/trans? *check
Battlegrounds					
Superhot	3D	Diegetic	Contextual	In	(Could be persistent? Need to define it a bit)
Superhot	3D	Diegetic	Contextual	In	
Wayward Sky	3D	Spatial	Contextual	In	
Wayward Sky	3D	Spatial	Contextual	In	*check this.
Robo Recall					
Burnout					
Burnout	3D	Diegetic	Contextual	In	
Final Fantasy XV	3D	Diegetic	Contextual	In	
Skyrim					
Steep	3D	Spatial	Contextual	*check	
Steep	3D	Non-diegetic	Persistent	*check	
Steep	3D	Spatial	Persistent	In	
Forza Horizon (3?)	3D	Spatial	*unknown	In	
Forza Horizon (3?)	3D	Non-diegetic	Persistent	In/Out	
Starfox 64	3D	Non-diegetic	Contextual	In/Out	
Star Wars: Rogue Squadron					
Star Wars: Rogue Squadron	3D	Non-diegetic	Persistent	In/Out	
Crimson Skies	3D	Non-diegetic	Contextual	In/Out	
Furi	3D	Spatial	Contextual	In	Kind of Iso instead of proper 3D
Left 4 Dead 2	3D	Spatial	Contextual	*check	Persistent, but game triggers health colours?
Assassin's Creed: Syndicate	3D	Spatial	Contextual	In/Out	Guides attention to location whether enemy is in or out of viewing angle.
Assassin's Creed: Syndicate	3D	Diegetic	Player-initiated	In/Occ	I'd consider this "in viewing angle". Also occlusion, because of X-Ray. Other stuff happens for tracking stuff out of viewing angle, but
World of Warcraft	3D	Non-diegetic	Persistent	In/Out	
Skyrim	3D	Non-diegetic	Persistent		
		-			

	This makes things a bit muddy.		Table 3 is about AR stuff mostly geometry? Point, Line, Area, Volume
Game	Geometry (Table 3)	Notes	 Persistent/Summoned/Contextual? (Maybe call this visibility?)
Doom (2016)	Point?		
Doom (2016)	Area? (Entire character highlighted)	Glory Kill	"Agent could usually be replaced by an icon or something maybe just more of a "point"
Doom (Original)	Volume (lighting is a volume)		
Zelda: Ocarina of Time	Point		* I don't know if Static vs. Dynamic is too interesting. Most of these things could be done for both.
Zelda: Ocarina of Time	Point		
Zelda: Ocarina of Time	HUD	Can we think about the	minimap as an area? Maybe that stuff that's being overlaid on HUD should be thought of differently. In world, geometry makes a lot of sense. On HUD, we may think of it diffe
Rayman: Legends	Point		
Diablo 2	HUD	Shows over the entire v	iew. HUD elements don't quite fit into the geometry thing. (I guess it guides to points. Some might guide to areas?)
Fable 3	Line	Shows as a sparkly line	through world.
Dragon Age: Inquisition	Point	A radar kind of visualiza	tion causes all the loot in an area to highlight.
Dragon Age: Inquisition	Point (depends on scale)	Practically guides the ch	naracter to just a point in the world. As you get closer, this may look more like an area. This is shown in a cutscene before it shows in the world, to prime the player.)
Overwatch	Point	Guides player attention	to source of damage.
Overwatch	Point? (More like HUD. Is it guiding attention?)		
Overwatch	Line (Draws a line between players.)	Is this active guidance?	Depends on player's goal, maybe? If player just wants a sense of where the teammate is, it could just be guiding attention? If I want to go to that teammate, then it's guidanc
Overwatch		Is tracking enemies gui	ding attention? Or active guidance? Or just something else? This is maybe where types of objects matter. (Locations vs. enemies) Highlighting POIs?
Overwatch			
The Witness			
Lovers in a Dangerous Spacetime			
Lovers in a Dangerous Spacetime			
Beyond Eyes			
Doom (Original)			
Thimbleweed Park	Area		
Day of the Tentacle (Remake)	Area		
Batman: Arkham Asylum			
Dark Souls 3			
Dark Souls 3			
Limbo			
Splatoon			
De Blob			
Super Mario Kart			
Mario Kart 8			
Bioshock			
Mirror's Edge			
Mirror's Edge			
Shadow of the Colossus			
Dead Space 3			

Dead Space 3 Dead Space 3 Left 4 Dead

Game	Geometry (Table 3)	Notes
NHL 17		
Epic Mickey	Volume	Volume - because the entire volume of the subtracted object can be interacted with.
Mark of the Ninia		
Mark of the Ninia		
Mark of the Ninja		
Crash Bandicoot		
Rocket League	Volume	This actually practically guides you to a volume. (The ball.) The visualization considers the whole volume, where many just consider a point (i.e. centre of the object).
Rocket League		Both visualizations make it possible to see track the ball's volume.
Rocket League	Area	
Angry Birds		
Jetpack Joyride	Point	
Horizon: Zero Dawn		
Assassin's Creed (various)		(Not sure which game this example is from)
Battlagrounds		hot sure when guine this example is notif.)
Battlegrounds		
Battlegrounds		(This is space
Battlegrounds		(TITS IS LYDIC)
Supernot		
Supernot		
Wayward Sky		
Wayward Sky		
Robo Recall		
Burnout		
Burnout		
Final Fantasy XV		
Skyrim		
Steep	Point and Volume	This is a great example of using a volume as a collision object.
Steep		
Steep	Line	
Forza Horizon (3?)	Line	Path plus colour to indicate ideal speed. (Encoding 2 different things in one element.)
Forza Horizon (3?)		
Starfox 64		
Star Wars: Rogue Squadron		
Star Wars: Rogue Squadron	HUD (But guides player to an area.)	
Crimson Skins	Point	
Chillison Skies	Area	
Full	Alea	
Lett 4 Deau 2 Assessing Creed, Sundiante		
Assassin's creed: Syndicate		
Assassin's Creed: Syndicate		
World of Warcraft		
Skyrim		

* Geometry: 2 things -- target (the thing we're navigating to or calling attention to) a Properties: ex. Persistent vs Summoned, Onscreen/Offscreen/Transition, Diegetic vs. Non-Diegetic, Static vs. Dynamic, Actor/Agent or Social Ex. Can avagate to a certain "point", by using a "line" * Path vs. Destination? Po This links to table 2 in some way.

Appendix C: Prototype Sketches



DISCOVER - SHOW ALL BOOKS BY A SPECIFIC ANTHOR

emphasized - CrLOWING HIGHLIGHT (PULSE) Cue.



DISCOVER - SHOW ALL BOOKS BY A SPECIFIC ANTHOR

Integrated - virtual arrows above books Cue



Stickers -s could be used in real life. (so, subtle) Subtle - Sticker on shelf below back.

DISCOVER - SHOW ALL BOOKS BY A SPECIFIC ANTHOR



LOOK TASK-SHOULDER SURFER DETECTION

emphasized - outside of document one. Mighlights when someone is hoking.

LOOK TASK-SHOULDER SURFER DETECTION





LOOK TASK-SHOULDER SURFER DETECTION

Overlaid cue- i con at edge of screen (off-screen guidance)



LOOK TASK-SHOULDER SURFER DETECTION

Subtle - Show shadow on document. Cue

GO TASK - DELIVERING MAIL IN AN OFFICE.



emphasized - Mail boy is highlighted. are

GO TASK - DELIVERING MAIL IN AN OFFICE.



integrated-pillars of light mark desks Cue (first one bright. Next ones dark)



GO TASK - DELIVERING MAIL IN AN OFFICE.

Overlaid - Compass drawn at Gue top of view.

GO TASK - DELIVERING MAIL IN AN OFFICE.



Subtle - Nametag displayed on desk. Che

Appendix D: Co-Author Permission



July 27, 2018

University of Calgary 2500 University Drive NW Calgary, AB T2N 1N4 Canada

I, Anthony Tang, give Kody Dillman permission to use co-authored work from our publication, "A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality" for his MSc thesis.

Sincerely,

Anflier

Anthony Tang (/ Associate Professor University of Calgary

Faculty of Arts and Social Sciences Department of Communications and New Media



Alex Mitchell Department of Communications and New Media Faculty of Arts and Social Sciences Block AS6, #03-22 11 Computing Drive, Singapore 117416 Tel: (65) 6516-3021 Fax: (65) 6779-4911 E-mail: alexm@nus.edu.sg

6 August 2018

To whom it may concern,

Re: Permission to use co-authored work for MSc thesis

I, Alex Mitchell, give Kody Dillman permission to use co-authored work from our publication, "A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality" [1] for his MSc thesis.

Please feel free to contact me for further information.

Sincerely,

Alex Mitchell Assistant Professor, Department of Communications and New Media

[1] Kody R. Dillman, Terrance Tin Hoi Mok, Anthony Tang, Lora Oehlberg, and Alex Mitchell. 2018. A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 140, 12 pages.

Riock AS5, 403-41, 11 Computing Drive, Singapore 117416 (rel: (65) 6516-4670; Fex: (65) 6779-4511 Weosite: www.fac.nuc.edu.og/cnm



11 September 2018

University of Calgary 2500 University Drive NW Calgary, AB T2N 1N4 Canada

I, Lora Oehlberg, give Kody Dillman permission to use co-authored work from our publication, "A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality" for his MSc thesis.

Sincerely,

Love Ochlleg

Lora Oehlberg

July 30, 2018

Terrance Mok 79 Edgehill Cr. NW Calgary, AB T3A2X5 Canada

I, Terrance Mok, give Kody Dillman permission to use co-authored work from our publication, "A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality" for his MSc thesis.

Sincerely,

1_m

Terrance Mok