4-2018

A visual interaction cue framework from video game environments for augmented reality

Kody R. DILLMAN

Terrance Tin Hoi MOK

Anthony TANG
Singapore Management University, tonyt@smu.edu.sg

Lora OEHLBERG

Alex MITCHELL

A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality

Kody R. Dillman Terrance Tin Hoi Mok Anthony Tang Lora Oehlberg Alex Mitchell
University of Calgary, Calgary, Canada National University of Singapore, Singapore
{ kody.dillman, terrance.mok2, tonyt, lora.oehlberg }@ucalgary.ca alexm@nus.edu.sg

ABSTRACT
Based on an analysis of 49 popular contemporary video games, we 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. We demonstrate that this framework can also be used to describe interaction cues for augmented reality applications. Beyond this, we show how the framework can be used to generatively derive new design ideas for visual interaction cues in augmented reality experiences.

Author Keywords
Interaction cues; guidance; augmented reality; game design.

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION
Augmented Reality (AR) systems present digital information atop tracked visuals of the physical world. 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. Many AR scenarios, including tour/museum guides, remote assistance, and games 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. 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. Our interest is in developing recommendations for designers looking to provide interaction and navigational assistance in AR systems.

CHI 2018, April 21–26, 2018, Montreal, QC, Canada.

ACM 978-1-4503-5620-6/18/04…$15.00
https://doi.org/10.1145/3173574.3173714

We 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. 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 attention in the scene (i.e. to suggest the player look at one spot or another). Yet, 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. We do not yet have a formal vocabulary for describing and understanding these interaction cues broadly.

We address two research questions in this work. First, how can we conceptualize these interaction cues, rearticulating the lessons and techniques game designers use to guide players around games? Second, how can we then apply these lessons in the context of augmented reality systems while considering the constraints and inherent limitations of the physical properties of reality, factors that do not necessarily exist in games?

To address these questions, we conducted an exploratory study of 49 video games to understand how visual interaction cues are used to communicate information about the game world to players. Our 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,

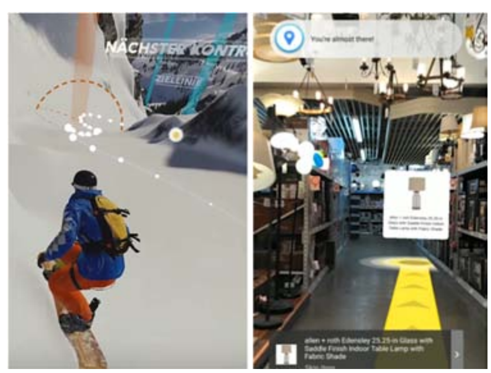

Figure 1. These *Go* interaction cues provide navigation guidance along a path. *Steep* (left) [L15] displays a dotted line in the course; *Lowe's In-Store Navigation,* a mobile AR app (right) [14], uses a bold yellow line.

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 illustrates the use of *Go* cues in the snowboarding game *Steep* [L15] (left) and in *Lowe's In-Store Navigation* app [14]. In both cases, these cues guide the player/user where to go in the environment.

We use this understanding as the basis of a framework that allows us 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 provide guidance to a user. This framework addresses the call by Billinghurst et al. [5] to develop new interaction vocabularies for AR, rather than simply re-using conventions from other domains that are not appropriate for the AR medium.

This work makes two contributions. First, based on a study of video games, we outline a framework that describes the design of cues that provide interaction and navigation guidance to players. Second, we demonstrate how designers can use this framework to describe and design new AR technologies that provide spatial guidance in the real world.

RELATED WORK

We briefly outline related work concerning designing navigation techniques from the AR literature that motivates our present work. We then discuss how frameworks from the games research literature help to address some gaps in the AR space (specifically, the issue of visual design).

Navigation in Augmented Reality. Grasset et al. [9] 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. labeling 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). Our interest is in primary experiences, where the use of the AR display is to provide guidance information. Grasset et al. [9] 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 [13], while others have explored dealing with visual cues that need to be occluded in various ways (e.g. [1–3]). Other researchers have tried visual blending [19].

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

Interaction Cues in Video Games. Bardzell [4] focuses on the design and use of interaction cues across a wide range of video games. 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 [15]: 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 [10,18], as well as player performance [16]. Generally, the work points to increased feelings of immersion as non-diegetic HUD elements are removed (e.g. [10]).

Jørgensen [11] 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, signaling 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. [L18] is a game example that blurs this line, where traditional HUD elements like health are part of the avatar's suit. Similarly, [L8] 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).

Summary. Our framework ultimately builds on the vocabulary introduced by Grasset [9], Bardzell [4], and Jørgensen [11,12]. The principal departure from this 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 thus we argue for its use as a generative framework.

METHOD

Perspective. While our focus on interaction cues comes from our interest in designing effective interaction cues for augmented reality (i.e. as designers), we tackle this question as experienced gamers who play games on both dedicated gaming platforms (Xbox, PlayStation, NES, etc.) and general-purpose computers. One member of our team previously worked in a game company. Thus, we had a wealth of "insider knowledge" of the domain from which we are drawing our insights.

Game Selection. We selected a total of 49 contemporary video games. Our goal was to collect interesting examples with high variance in how cues were designed and used. We used a purposive selection technique, where we selected games that use interaction cues to guide players. We intentionally excluded AR games from the selection, as the space is unnecessarily limiting; the AR community is young, and the current limitations of technology do not allow for meaningful interaction with real world spaces. While we began by identifying games we were familiar with, we were conscious of our personal preferences for game genres, and sought to ameliorate the effect of the potential bias. To this end, we expanded the set of games outside of our personal experiences through recommendations from colleagues (with whom we discussed our research goals). Among these recommendations, we were additionally selective: if a game's interaction cues were already represented in our sample, we did not include the game. The sample we report on represents a mix of first person shooter games, third person adventure games, 3D and 2D platformers, driving games, and puzzle games. Our sample is not intended to be exhaustive; however, it is representative of the wide range of experiences that contemporary game players enjoy.

Method and Analysis. We 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 we had experience with, we replayed some games; for games that we did not have personal experience with, we watched online "walkthrough" gameplay videos. For this latter set of games, we watched the game until we felt we had a clear sense of a player's in-game experience.

We were specifically 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, we focused our 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. We paid attention to both overt aspects of the UI, as well as understated elements. We reasoned that regardless of whether a cue worked well, they were explicitly designed elements (from the perspective of the game designer), and that as designers, we could learn from both successes and failures.

For each game, we identified visual elements that fulfilled our criteria of potentially helping a player navigate the game world. We collected screenshots of each of these, describing how a player would use them, what they looked like, and the context of how they appeared. We used a thematic analysis process, where we iteratively grouped, labeled, discussed and re-labeled categories and axes that described and explained the various cues. This process involved several meetings of all the authors, with the first two authors presenting screenshots to the other authors and discussing the examples of the cues. These categories, labels, and axes were iteratively refined as we added more games into our sample until we found the framework to be relatively stable.

FRAMEWORK: VISUAL INTERACTION CUES IN VIDEO GAMES

Our framework describes the interaction cues we found in our sample of video games 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 when the cue is shown. Table 1 summarizes the dimensions of the framework, relating these to gameplay screenshots in Figure 2.

Dimension 1: Task / Purpose

We observed in our 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 [8] affordances of the environment may suggest more things that can be interacted with than the game 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 and 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.

We 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 2-d illustrates how

Dimension	Category	Description
D1: Purpose	Discover	Informs the player of objects or points of interest in the environment. Figure 2-a: A part of the wall is coloured with slightly off-saturation to indicate to players that the wall can be manipulated [L10].
	Look	Informs the player where to put their visual attention in a timely manner. Figure 2-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 2-i: The added white line and red arches show the player where to go in the race course [L15].
D2: Markedness	Subtle	The cue blends into the environment seamlessly. Figure 2-b: To indicate that the player is being shot at, the enemy's gun is painted with a lit flare [L12].
	Emphasized	An object or surface in the environment is highlighted. Figure 2-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 2-h: A yellow widget painted below the avatar points at a nearby enemy that is suspicious of the player's actions [L16].
	Overlaid	Virtual objects are added atop the viewport, and do not track the view. Figure 2-l: A compass at the top of the player's HUD shows "North" in the game, along with specific points of interest [L2].
D 3: Trigger	Player	The cue is activated by an explicit player action. Figure 2-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 [L13].
	Context	The cue is activated by some implicit player action Figure 2-f: As the player gets close to the door, it becomes emphasized with a highlight around its edges [L6].
	Other/Agent	The cue is activated by some other agent (system or other player) Figure 2-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 [L11]
	Persistent	The cue is always visible. Figure 2-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 1. Summary of the visual interaction cues framework. These dimensions are illustrated by in-game screenshots in Figure 2.

Dragon Age: Inquisition [L3] uses an outlined highlighting cue to emphasize certain artefacts in the environment (here, that the gold pouch can be looted for gold). Figure 2-j shows how *World of Warcraft* [L4] uses a "mini-map" overlay (representing an iconic bird's-eye-view of the entire game world) to show the player where mineable minerals and important characters can be found in the map.

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 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 2-h) to overcome the inherent limitations of, for example, 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).

We 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 2-e illustrates a *Look* cue in *Doom* [L11], 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 2-h shows a *Look* cue where the yellow ring around the player's avatar points toward a nearby enemy position (relative to the player's location). In addition, the red bars indicate that the enemy is currently suspicious of the player [L16].

Go. Finally, games frequently take place in large virtual environments that the player navigates through 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 our 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* [L15] in Figure 1-left).

Dimension 2: Markedness

The second major dimension in our sample corresponds to some ideas first presented in [4,11,12], where the

Figure 2. Screenshots from some of the games from our sample set: (a) [L10], (b) [L12], (c) [L13], (d) [L3], (e) [L11], (f) [L6], (g) [L14], (h) [L16], (i) [L15], (j) [L4], (k) [L5], (l) [L2].

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 [4]. Here, we are strictly concerned with the visual presentation or design of the cue: *Subtle*, *Emphasizing* an object, *Integrated* with the environment, or *Overlaid* atop of 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 3 (top), the level design in *Bioshock* [L1] makes use of drastic contrast in lighting, where the purpose of the cue is to provide a player with a clear destination (*Go* cue). While the cue uses visual contrast, it does not stand out given the in-game narrative.

Figure 2-a shows a *Subtle* cue in *Doom* [L10], 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 3 (right) shows another example from *Dragon Age: Inquisition* [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* [L11] uses flickering lights to attract a player's attention toward certain corridors, supported by the in-game narrative that the base has been destroyed by fire, thus the neon lights are in a half-working state (*Go* cue).

Figure 3. Left, *Bioshock* uses environmental lighting as a *Subtle Go* cue [L1]. Right, *Dragon Age* uses green smoke as a *Subtle Go* cue [L3]. Bottom, left, *Jetpack Joyride* uses a blinking *Overlaid Look* cue to show where the rocket is about to appear on screen (bottom, center) [L9].

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 2-d, *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 2-f shows a highlighted outline effect from *The Witcher 3: Wild Hunt* [L6], 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, we 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 is may be oriented so it is always legible to the player), while others may ignore both.

Figure 2-g shows an *Integrated Discover* cue from *Thimbleweed Park* [L14], where a label appears to tell the player what actions can be taken on the object. Figure 2-i shows a set of pillars in *Steep* [L15]. The pillars are virtual objects placed atop the game world that track the game world to show the player where to go (*Go* cue).

Some first-person shooters make use of the same *Integrated* cue to represent a teammate, but 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 largely context dependent, particularly as it relates to gameplay.

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 we found were represented either as UI widgets (e.g. a compass, bird's-eye-view minimap, aiming reticule), or widgets that made use of the edges of the screen to refer to objects or destinations beyond the edge of the viewport into the world.

Figure 2-k shows a screenshot from *Overwatch* [L5], where red highlighting at the bottom edge of the screen is an *Overlaid Look* cue that tells the player that s/he is being attacked from behind (top edge represents front; right edge represents from the right side, and so forth). This is sometimes represented in the center of the screen as part of the aiming reticule. Figure 2-l 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 artefacts/destinations are relative to the player's current orientation.

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 3-bottom). 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 world with objects that are in front but indicated with an arrow at the upper edge of the screen; however, our surveyed games generally stick with one convention without issue.

We 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 3 (bottom) shows an *Overlaid Look* cue for an object that is out of view; however, when the object enters the field of view Figure 3 (bottom, middle), the cue changes to a *Subtle Look* cue [L9]. 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* [L17]). The intention here may be to provide a *Look* cue even when there is no clear line-of-sight. *Integrated* and *Overlaid* cues were visible regardless of whether there was an obstruction in our sample (perhaps a distinguishing feature of *Emphasized* cues). In our 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.

Dimension 3: Trigger

The third major dimension of our framework considers how the visual interaction cue is triggered into visibility. We identify four levels of trigger based on a player's agency over the trigger, from an explicit act to triggers caused by other agents and finally to persistent cues.

Player. *Player-triggered* visual interaction cues are activated by an explicit action by the player. The example in Figure 2-c shows a player's avatar from *Shadow of the Colossus* [L13] 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 2-g illustrates how a text cue shows when the player hovers his mouse above the toaster oven holding the hot dogs in *Thimbleweed Park* [L14] (*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 our 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 area for a cue), or the player's view in the game world. For instance, in *Thief* [L7], "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 2-k (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 2-j from *World of Warcraft* [L4]) (*Overlaid Discover* cue).

Summary

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 when the cue is shown.

USING THE FRAMEWORK WITH AR

Although we developed this framework by studying and describing interaction cues in video games, we view the primary application of the framework is to be for generating interaction cues ideas for augmented reality. Video games are an ideal starting point to develop these cues, as they have long addressed the issue of guiding players through virtual worlds. Even if techniques do not translate directly to AR guidance in the real world, knowledge of how games have solved the problem can inspire AR designs. We are principally concerned with AR that uses head-mounted displays (where the user's view is strictly defined by the AR display); we will revisit this framework for handheld AR form factors in the Discussion section (below). In this section, we first describe how the framework dimensions apply to AR. Then, we show that the framework provides an effective vocabulary for describing and analyzing interaction cues in current augmented reality applications. Finally, we use the framework to suggest design alternatives for these applications.

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* maps directly to real world situations where it is unclear what artefacts 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 tour context, certain 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, we 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, this refers to a user's movement, where the user's spatial location activates a cue (e.g. information about the entered space). 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 remain always on and require no explicit action by the user.

Markedness. Our framework defines four levels that describe how overt a cue is in its presentation—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 4-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 4-d).

The central departure when applying the framework to AR is this markedness dimension. From a technical perspective, every visual augmentation in AR is either an *Integrated* cue or an *Overlaid* cue. 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 4 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 eye-view overlay; (e) makes it appear as though two of the doors are closed;

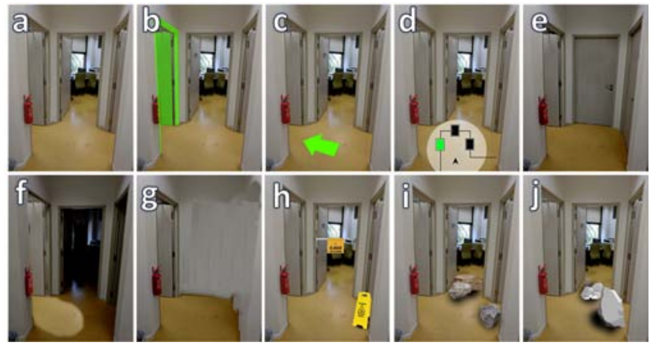

Figure 4. Variations on an imaginary AR interface that provides a Go cue to the door on the left.

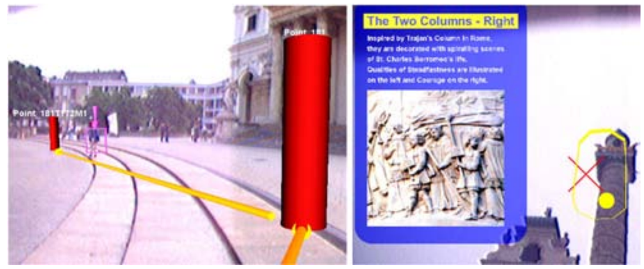

Figure 5. Reitmayr and Schmalstieg's AR tour guide system.

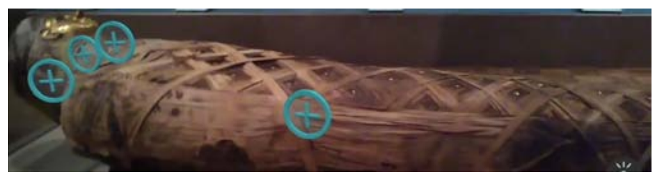

Figure 6. The Lumin Project gives museum-goers an AR experience for navigating exhibits and learning about artefacts. The AR view of artefacts provides *Integrated Discover* cues for more information.

(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 is determined by presentation and fit to the environment (a subjective issue beyond the designer's control).

Describing AR Interaction Cues

Next, we show how our framework can be used to describe AR interaction cues, using two examples of existing AR applications: Reitmayr and Schmalstieg's Vienna Tour Guide [17], and the Lumin project [6].

Example 1 – Vienna Tour Guide System [17]. 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-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 *Emphasize* the path on the ground to the next waypoint. Finally, as illustrated in Figure 5 (right), the system shows a yellow polygon around an annotated building (an *Integrated Discover* cue) that signals to that the user can tap to get more information.

Example 2 – Lumin Project [6]. The Lumin Project is a handheld AR tool deployed at the Detroit Institute of Arts. 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*). In the *User-triggered* information mode (Figure 6), *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.

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

Generating AR Scenarios with the Framework

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

Example 1 – Vienna Tour Guide System [17]: <u>Tina the Quick Tour Guide</u>. 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 on her app 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 [6]: <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. <u>*Fei the Science Fan*</u>. Fei gets the *Look* cue for the show, and wants to attend. She activates the *Integrated Go* cue which looks like 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.

Summary. These scenarios illustrate how the framework functions as a generative tool, allowing us to explore new possibilities for the use of interaction cues in AR. To generate the above examples, we start with the user story, identifying the intended purpose of the cue. Our next consideration is markedness: how visible does the cue needs to be (e.g. Figure 4). Based on the expected usage, we consider different trigger opportunities, identifying the one that matches how we want the user to see the cue. The vocabulary provided by the framework gives us 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. school children), she needs to find her clients immediately; 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.

DISCUSSION

Although consumer-grade Augmented Reality is in its infancy, prior work alongside our generative explorations with the framework provide several points of discussion.

Limitations Presented by the Physicality of AR. Whereas 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. Whereas some games use 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* [L13]). AR designers cannot use this type of cue.

Importance of Discover Cues. Based on existing AR experiences (e.g. [17, 6, 14, 13, 9]), we expect *Discover* cues to be extremely important in the early days of AR: first, not everything in the physical world is interactable or tagged; second, users do not yet have well-developed mental models of the classes of AR applications that might exist (i.e. since certain applications may make some objects more or less likely to be interactive targets); and finally, the visual language of these kinds of interaction cues in AR are not yet well-developed. While the first two issues may end up being overcome over time, it is important for designers to consider how to evolve the visual language. Designers will need to prudently consider the interplay between expectations of the physical environment and the AR application to engender effective designs.

Trade-off: Markedness. The markedness dimension of the framework raises interesting questions about "naturalness" and "jarringness." Users in an AR context are not bound to the content provided by the AR headset. If *Subtle* cues are designed too subtly in an AR context, they could be missed (likely undesirable). Thus, while *Subtle* cues may seem desirable as a goal (as in games [20]), marked cues may be more desirable if the intention is for users to see them.

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 intend 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.

Trade-off: Triggers and Causality. We have outlined a range 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 will 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).

Visibility and "Fit" of Cues. We are also limited as AR designers by current sensing technologies: 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. [1–3,17]). Clipping these visuals properly will aid in interpretability of cues; alternately, designers should consider decorating cues to indicate whether the actual physical object/location *should* be visible (e.g. given known buildings, hills, mountains, etc. in the space). Similarly, we cannot yet accurately use *Emphasized* cues on physical objects. For example, the Vienna Tour Guide (Figure 5) 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 the boulders of Figure 4-i 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. can the user understand cue in context).

While beyond the scope of the discussion here, the near-future ability to render effective *Subtle* cues raises interesting ethical questions: is it right to show someone something that is not present (e.g. doors in Figure 4-e,f), or to visually take away an object that is physically there (e.g. Figure 4-g)? This is particularly important if the alteration of the user's view could lead to accidents or injury.

Impact of Form Factor. With handheld AR, designers should assume users can see both the AR perspective and a real perspective on the world. Here, *Subtle* cues will seem jarring, as users will be able to easily see differences between perspectives. We recommend designers focus on making it clear whether a destination or target is likely to be visible, and from what orientation the target object should be visible. Ignoring these factors draws attention to the implementation of the cue rather than allowing a user to interpret the cue from the augmented view (i.e. that they are overlaid atop the viewport anyway). For example, in the Lowe's App [14] (Figure 1-right), the white placard always faces the user, meaning the cue cannot be used to identify which shelf the product is on. Instead, setting the orientation of the placard to match the target object would allow a user to employ the placard to its full potential.

CONCLUSIONS

Video game designers have developed and honed a visual language for interaction cues. Our interaction cue framework illuminates the roles of purpose, visual design and interaction design for these cues. Further, we find that it can describe interaction cues from AR experiences, and we show how designers can use this framework to generate new designs for interaction cues in AR. Designers of AR experiences as well as those building AR platforms (e.g. [7]) will be able to build from this work to develop a parallel visual language of interaction cues for AR.

REFERENCES

1. B. Avery, C. Sandor, and B. H. Thomas. 2009. Improving Spatial Perception for Augmented Reality X-Ray Vision. In *2009 IEEE Virtual Reality Conference*, 79–82. https://doi.org/10.1109/VR.2009.4811002
2. Benjamin Avery, Bruce H. Thomas, and Wayne Piekarski. 2008. User Evaluation of See-through Vision for Mobile Outdoor Augmented Reality. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality* (ISMAR '08), 69–72. https://doi.org/10.1109/ISMAR.2008.4637327
3. R. Bane and T. Hollerer. 2004. Interactive tools for virtual x-ray vision in mobile augmented reality. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, 231–239. https://doi.org/10.1109/ISMAR.2004.36
4. Shaowen Bardzell. 2008. Systems of Signs and Affordances: Interaction Cues in 3D Games. *EnLeino, O*: 191–209.
5. Mark Billinghurst, Adrian Clark, Gun Lee, and others. 2015. A survey of augmented reality. *Foundations and Trends® in Human–Computer Interaction* 8, 2–3: 73–272.
6. Detroit Institute of Arts. Lumin. *Lumin*. Retrieved September 18, 2017 from https://www.dia.org/about/press/media-kits/lumin
7. Georgia Tech Research Corporation. argon.js. Retrieved September 18, 2017 from https://www.argonjs.io/
8. James J. Gibson. 1977. The Theory of Affordances. In *Perceiving, acting, and knowing: Toward an ecological psychology*, Robert Shaw and John Bransford (eds.). Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 67–82.
9. Raphael Grasset, Alessandro Mulloni, Mark Billinghurst, and Dieter Schmalstieg. 2011. Navigation Techniques in Augmented and Mixed Reality: Crossing the Virtuality Continuum. In *Handbook of Augmented Reality*. Springer, New York, NY, 379–407. https://doi.org/10.1007/978-1-4614-0064-6_18
10. Ioanna Iacovides, Anna Cox, Richard Kennedy, Paul Cairns, and Charlene Jennett. 2015. Removing the HUD: The Impact of Non-Diegetic Game Elements and Expertise on Player Involvement. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (CHI PLAY '15), 13–22. https://doi.org/10.1145/2793107.2793120
11. Kristine Jørgensen. 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*: 78–97.
12. Kristine Jørgensen. 2012. Between the Game System and the Fictional World: A Study of Computer Game Interfaces. *Games and Culture* 7, 2: 142–163. https://doi.org/10.1177/1555412012440315
13. Mark A. Livingston, J. Edward Swan II, Joseph L. Gabbard, Tobias H. Höllerer, Deborah Hix, Simon J. Julier, Yohan Baillot, and Dennis Brown. 2003. Resolving Multiple Occluded Layers in Augmented Reality. In *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality* (ISMAR '03), 56–. Retrieved September 17, 2017 from http://dl.acm.org/citation.cfm?id=946248.946796
14. Lowe's. In-Store Navigation. *Lowe's Innovation Labs*. Retrieved September 18, 2017 from http://www.lowesinnovationlabs.com/instorenavigation/
15. Donald A. Norman. 1988. *The psychology of everyday things.(The design of everyday things)*. Basic Books.
16. M. Peacocke, R. J. Teather, J. Carette, and I. S. MacKenzie. 2015. Evaluating the effectiveness of HUDs and diegetic ammo displays in first-person shooter games. In *2015 IEEE Games Entertainment Media Conference (GEM)*, 1–8. https://doi.org/10.1109/GEM.2015.7377211
17. Gerhard Reitmayr and Dieter Schmalstieg. 2004. Scalable techniques for collaborative outdoor augmented reality. In *3rd IEEE and ACM international symposium on mixed and augmented reality (ISMAR'04), Arlington*. Retrieved September 18, 2017 from https://www.ims.tuwien.ac.at/publications/tr-1882-02e.pdf
18. P. Salomoni, C. Prandi, M. Roccetti, L. Casanova, and L. Marchetti. 2016. Assessing the efficacy of a diegetic game interface with Oculus Rift. In *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 387–392. https://doi.org/10.1109/CCNC.2016.7444811
19. C. Sandor, A. Dey, A. Cunningham, S. Barbier, U. Eck, D. Urquhart, M. R. Marner, G. Jarvis, and S. Rhee. 2010. Egocentric space-distorting visualizations for rapid environment exploration in mobile mixed reality. In *2010 IEEE Virtual Reality Conference (VR)*, 47–50. https://doi.org/10.1109/VR.2010.5444815
20. Jesse Schell. 2014. *The Art of Game Design: A Book of Lenses, Second Edition*. CRC Press.

LUDOGRAPHY

1. 2K Boston and 2K Australia. (2007). Bioshock. Game. [Windows]. (21 August 2007). 2K Games. Novato, CA, USA
2. Bethesda Game Studios. (2011). The Elder Scrolls V: Skyrim. Game. (11 November 2011). Bethesda Softworks. Rockville, MD, USA
3. BioWare Edmonton. (2014). Dragon Age: Inquisition. Game. [PlayStation 4]. (18 November 2014). Electronic Arts. Redwood City, CA, USA

4. Blizzard Entertainment. (2004). World of Warcraft. Game. [Windows]. (23 November 2004). Blizzard Entertainment. Irvine, CA, USA
5. Blizzard Entertainment. (2016). Overwatch. Game. [Windows]. (24 May 2016). Blizzard Entertainment. Irvine, CA, USA
6. CD Projekt. (2015). The Witcher 3: Wild Hunt. Game. [Windows]. (19 May 2015). CD Projekt RED. Warsaw, Poland
7. Eidos Montreal. (2014) Thief. Game. [Windows]. (February 25, 2014). Square Enix. Tokyo Japan
8. Guerilla Games. (2017). Horizon Zero Dawn. Game. [Playstation 4]. (February 28, 2017). Sony Interactive Entertainment. Tokyo, Japan
9. Halfbrick Studios. (2011). Jetpack Joyride. Game. [iOS]. (September 1, 2011). Halfbrick Studios. Brisbane, Australia
10. id Software. (1993). Doom. Game. [MS-DOS]. (10 December 1993). GT Interactive. New York City, NY, USA
11. id Software. (2016). Doom. Game. [Windows]. (13 May 2016). Bethesda Softworks.Rockville, MD, USA
12. SUPERHOT Team. (2016). Superhot. Game. [Windows]. (25 February 2016). SUPERHOT Team. Łódź, Poland
13. Team Ico. (2005), Shadow of the Colossus. Game. [PlayStation 2]. (18 October 2005). Sony Computer Entertainment. Tokyo, Japan
14. Terrible Toybox. (2017). Thimbleweed Park. Game. [Windows]. (30 March 2017). Terrible Toybox. Seattle, WA, USA
15. Ubisoft Annecy. (2016). Steep. Game. [Windows]. (2 December 2016). Ubisoft. Rennes, France
16. Ubisoft Quebec. (2015. Assassin's Creed Syndicate. Game. [Windows]. (19 November 2015). Ubisoft. Rennes, France
17. Valve Corporation. (2009). Left 4 Dead 2. Game. [Windows]. (November 17, 2009). Valve Corporation. Bellevue, WA, US
18. Visceral Games. (2013). Dead Space 3. Game. [Windows]. (5 February 2013). Electronic Arts. Redwood City, CA, USA