Examining the Essentials of Stealth Game Design

Youssef Khatib

Game Design Bachelor Thesis, 15hp
Game Design and Graphics, spring semester, 2013
Supervisors: Ernest Adams, Hedda Gunneng
 Examiner: Steven Bachelder
Abstract
Through looking into the inner workings of stealth centric games, this paper aims to find out the essential components of this type of videogames. Examining the history of such games and the design principles of stealth centric games in relation to the participating player this paper will methodically examine games in the light of the arguments of industry professionals. After that a framework is extracted, identifying the principal core components of stealth centric game design.
Table of contents

1. Introduction .............................................................................................................................................. 1
   1.1 Purpose ............................................................................................................................................... 2
   1.2 Question ............................................................................................................................................. 2
   1.3 Scope of work .................................................................................................................................... 2

2. Background ............................................................................................................................................... 3
   2.1 Avatar Means ..................................................................................................................................... 4
      2.1.1 Avatar Characteristics .................................................................................................................. 4
      2.1.2 Movement .................................................................................................................................... 4
      2.1.3 Visibility ..................................................................................................................................... 5
      2.1.4 Tools ............................................................................................................................................ 6
   2.2 Avatar Challenges ............................................................................................................................. 6
      2.2.1 Guards ......................................................................................................................................... 6
      2.2.2 Environmental Hazards .............................................................................................................. 6
      2.2.3 Security Measures ..................................................................................................................... 6
      2.2.4 Alarm state .................................................................................................................................. 7

3. Method ...................................................................................................................................................... 8

4. Analysis .................................................................................................................................................... 9
   4.1 Principles of stealth game design ........................................................................................................ 9
      4.1.1 Designing for stealth centric play ............................................................................................... 9
      4.1.2 The importance of Artificial Intelligence .................................................................................. 10
      4.1.3 AI Decision making ................................................................................................................... 13
      4.1.4 Overall level design considerations .......................................................................................... 16
   4.2 Perspectives: Camera models and player overview ................................................................................. 18
      4.2.2 Context sensitive camera placement .......................................................................................... 19
      4.2.3 Mixing and matching camera models ....................................................................................... 19
      4.2.4 Acquiring information beyond the camera model ...................................................................... 20

5. Discussion .................................................................................................................................................. 21
   5.1 The AI guard and player relation ...................................................................................................... 21
   5.2 Overall Sequence of stealth play ....................................................................................................... 22
5.3 Beyond shadow space ................................................................................... 22
5.4 Interconnecting challenges .......................................................................... 23
6. Conclusion ...................................................................................................... 24

Further Research ............................................................................................... 25

Glossary ............................................................................................................. 26

Bibliography ...................................................................................................... 27

Table of figures

Figure 1 (Partial failure margin in combat centric games) – (Smith, 2006) ................. 9
Figure 2 (Partial failure margin in stealth centric games) – (Smith, 2006) ....................... 10
Figure 3 (Multiple viewcones) – (Leonard, 2003).............................................................. 11
Figure 4 (Stealth) – (Smith, 2002).................................................................................... 12
Figure 5 (A finite-state machine – Behavioral states illustrated by nodes which are join by arrows explaining possible transitions between states.) – (Kehoe,2009) .......................................................... 13
Figure 6 (Basic relationship requirement) – (Smith, 2002) .............................................. 15
Figure 7 (Open-ended level structure) – (Smith, 2006) note: image has been edited for the purpose of this paper .................................................................................................................. 17
Figure 8 (Methods of pacing) – (Smith, 2006) note: image has been edited for the purpose of this paper .................................................................................................................. 22
1. Introduction

This paper will look into the inner workings of stealth centric games to find out the essential components of this type of videogames. The paper will examine the history of such games and the design principles of stealth centric games in relation to the participating player.

To do this, one needs to find a suitable definition of what stealth centric games are. The most common way is to define stealth games as genre. Game design books categorize stealth games as a sub-genre of action games. This way of categorizing is present for example in Ernest Adams and Andrew Rollings, *The Fundamental of Game Design* (2007) and Scott Rogers *Level Up! The Guide to Great Video Game Design* (2010). However this way of categorizing games can cause some confusion due to the way the action genre is defined.

The retailers began organizing their shelves along these lines. Publishers created product plans based on them. Gamers learned to prefer one genre over another, and to identify themselves as fans of shooters or platformers or real-time strategy. (Adams, 2009)

In the article *Sorting Out the Genre Muddle*, Ernest Adams explains the origin of game genre usage as symptom of the videogames industry’s growth. As more money was being made, the cost and investments of developing new games increased rapidly. Only large publishers had access to store shelves, making publishers more cautious of the content being developed and less willing to use creative approaches. So videogames eventually settled into a set of genres recognized as: sports, strategy, racing, fighting, action, role-playing, and so on.(Adams, 2009)

Stealth games are currently recognized as a sub-category of the action genre. This causes problems because of the use of the word genre in marketing.

An action game is one in which the majority of challenges presented are tests of the player’s physical skills. (Adams, Rollings, pp.436, 2007)

The unregulated structure for the definition of the action genre causes problems, due to its lack of clarity. Therefore this paper will turn towards the MDA (Mechanics, Dynamics, Aesthetics) Framework (Robin Hunicke, Marc LeBlanc and Robert Zubek, 2004) as a reference point for defining stealth centric games. In the MDA framework, the design of a game can be divided into considerations of a game’s mechanics, dynamics and aesthetics. Mechanics are functions that trigger actions in the game space, as several mechanics interacting with each other this interaction is referred to as the dynamics of a game. These two components are often built and designed according to a desired aesthetic goal.

Randy Smith defines stealth centric games as having the following common aesthetic goal: “[To] create the illusion of a securely guarded area that the player can sneak through by virtue of leveraging their unique abilities and tools to create and exploit security flaws” (2006). In Ernest Adams’s definitions of challenge types, stealth is a sub-category for conflict challenges where the player is “avoiding being seen” (Adams and Rollings, pp.23, 2007). From the above one can
devise a more precise definition what stealth games are. This will be used as a base for differentiating stealth centric games from other conflict centric games.

1.1 Purpose

The purpose of this paper is to familiarize the reader with stealth-centric games and to differentiate them from action games, by examining similarities between stealth games and the aesthetics they aspire to, as reflected in their mechanics.

1.2 Question

What are stealth centric games, how are they designed and what attributes are essential for adhering to their aesthetic goal?

1.3 Scope of work

There are many instances where these games deviate from their stealth centric approach. This happens when they introduce combat centric scenarios for the purpose of conveying the overarching narrative of the gameworld and a prime example is when boss encounters happen in which players are presented with a set of expectations unlike the default challenges present in the game. This paper will only focus on the main procedure of play for these games and not the atypical challenges found in boss battles.
2. Background

Stealth games are widely thought to have started with Metal Gear for the MSX2 home computer in 1987. However the earliest known game with stealth elements was released as early as 1981 with Castle Wolfenstein (Muse Software). Worthy of note is that many of the mechanics introduced early on, like hiding dead bodies and frisking guards for items, later disappeared from newer games for some time and did not resurface until decades later.

The year 1998 saw the release of several titles that made use of stealth systems. Many of these games became commercial successes allowing this style of play to reach new audiences previously unaware of this type of game systems. During this time, many of these games evolved further, by borrowing mechanics and elements from other non-stealth centric games.

By the turn of the millennium, stealth centric games started gaining momentum with releases like Thief II: The Metal Age (Looking Glass Studios, 2000), Metal Gear Solid 2: Sons Of Liberty (Konami, 2001) and Tom Clancy's Splinter Cell (Ubisoft, 2002). Many of the early mechanics were re-introduced and Artificial Intelligence systems became increasingly important for stealth play.

There is an abundance of articles from game critics, from the enthusiast press, and industry professionals about this type of game. However, the information presented in such articles is often subjective in nature, since they seldom have an analytical approach in organizing information.

The coming sections will examine the features and traits an avatar might possess to avoid detection. There are also obstacles that generally work against the player to create challenges. These come in the form of patrolling guards, security measures such as security cameras, alarm triggering traps and other environmental hazards. This paper derives these features from stealth centric games that have made use of them in a recurring manner through several released titles. The background is divided into two major sections; the first one representing the player and their actions. The second section presents dynamic and static obstacles occupying the game space.
2.1 Avatar Means

2.1.1 Avatar Characteristics

The avatars featured in these games have a certain set of common characteristics. For example the avatar the player inhabits in *Tom Clancy’s Splinter Cell* (Ubisoft, 2002) acts in a way that makes him seem dexterous and agile but also incapable of fending off multiple armed opponents in a direct confrontation. Thus the player has to learn to use the avatar’s agility attributes in order to “avoid being seen” (Adams and Rollings, 2007) and “[leverage] their abilities and tools to create and exploit security flaws” (Randy smith, 2006) in order to progress.

The avatar’s physical abilities can often be used to dispose of threats, overcome certain obstacles and hide behind cover. In *Metal Gear Solid 3: Snake Eater* (Konami, 2004) the game uses a grappling system that allows the player to quickly knock guards unconscious. In *Assassin’s Creed* (Ubisoft, 2009) the game uses an automated navigation system that allows the avatar to transition seamlessly between running on the ground and scaling buildings.

2.1.2 Movement

How movement is designed varies greatly in these games and is dependent on how the design approaches the subject of sneaking.

Contemporary stealth centric games require the player to manipulate the speed at which they travel through the game-space, without alerting any threats. This type of games offers mechanics to alter the pace of movement by implementing different walk and run modes. These modes can either work as different states or in transition from one to the other. Furthermore, these modes can alter the avatar’s stance earlier games of stealth. Tended to only have one mode of movement; this meant that in early games guards generally did not have complex systems for sensory perception. (See: artificial intelligence section 6.1.2)

*Hitman: Codename 47* (Eidos, 2001) makes use of a mechanic called “Sneak mode”(Eidos, *Hitman: Codename 47 Manual*, 2001)This is a mode of movement, which allows the avatar to move silently without detection. The game contains several movement modes all of which emit varying levels of sound relative to the speed of movement. However, when the player enters “sneak mode” the avatar’s sound emissions are reduced.

Speed management in *Tom Clancy’s Splinter Cell* (2002, Ubisoft) functions in a particular manner, whereby the player controls the avatar’s movement speed within a threshold. The function is similar to that of a accelerator pedal in automobiles. At the highest rate of movement the avatar emits the highest possible sound while moving, while the opposite is true for the lowest. The avatar profile also alters this, as changing the stance of the avatar also changes the movement pattern. This means that an avatar moving in a crouch stance (i.e. lower profile) emits less sound.
2.1.2.1 Cover systems

Hiding behind cover is a way of avoiding detection. In early stealth games, the player positioning themselves behind a vision-obstructing object was enough to avoid detection. Over time, taking cover came to develop into a mechanic, which allowed the avatar to press himself against these vision-obstructing objects. These objects vary in their shapes and sizes, as long as they facilitate adequate cover from the guard’s sight (Metal Gear Solid, Konami 1998).

This further developed into a mechanic where designated objects acted as cover, enabling the player at a push of a button to position the avatar behind cover. Movement modes were also added to these objects, allowing players to strafe along covers for instance (Tom Clancy’s Splinter Cell, Ubisoft 2002).

2.1.3 Visibility

How a game communicates whether the player is visible to the guards or not works differently in every game. This is governed through rules that are applied in the gameworld. The most common one is cover. Allowing the avatar to press himself against a wall to peek around corners, and to crouch behind smaller objects is a common way to obstruct the guard’s sight of the avatar.

In stealth games developers try to give the guards simulated sensory perceptions such as vision, hearing and sometimes smell to create a human like awareness. For further information see the Artificial Intelligence section 6.1.2.

2.1.3.1 Concealment

Concealment, unlike cover, is a way for the avatar to blend into the surroundings, whether it be through camouflage (Metal Gear Solid 3: Snake Eater, Konami 2004), the use of shadows for hiding (Tom Clancy’s Splinter Cell, 2002) or the use of crowds to blend into. (Assassin’s Creed II, Ubisoft 2009) When the avatar is concealed, special rules apply restricting the avatar’s behavior. For example in the situation in which a guard is facing the avatar’s general direction, while incapable of sensing the avatar’s presence due to concealment, sudden movements on the part of the avatar will cause the avatar to be detected. This serves as an abstraction of how quickly the human eye can react to movement. However while this is applicable to concealment by shadows and camouflage, sudden movements will not cause detection in the case of crowds. Crowds cycle constantly, moving around and sometimes move in groups that the avatar can move among while maintaining his concealment. With the potential exception of crowds, concealment generally does not protect the avatar from incoming gun fire or other attacks.
2.1.4 Tools

Gadgets and firearms available to the player increase their options when they face game challenges. Traditionally stealth games rely on gadgets that help the player gauge the dangers they are about to encounter. These tools can range from radars that cover entire areas (*Metal Gear Solid*, Konami, 1998) to snake cameras that let the player peek around corners and under doors (*Tom Clancy’s Splinter Cell*, Ubisoft, 2002). Gadgets that create distractions are often used to divert attention from the avatar’s planned direction of entry. These tools usually work as sound causing devices, such as coin that be thrown which attracts a guard’s attention to its landing spot (*Hitman: Codename 47*, Eidos Interactive, 2001). Firearms are often presented in these games as a last resort in case the avatar is detected and defensive measure has to be taken and sometimes used as a way to neutralize guards standing in the avatar’s path of progression.

2.2 Avatar Challenges

2.2.1 Guards

Guards are generally the main dynamic obstacles in a stealth centric game; they can patrol certain routes or have static placements. Sometimes they also march in groups (*Assassin’s Creed II*, 2009, Ubisoft). Upon the detection of the avatar they usually react with hostility, often attacking the avatar while alerting fellow guards to the avatar’s location. If the avatar manages to escape their sight, they initiate a search for the avatar, using the senses and intelligence they have been given.

2.2.2 Environmental Hazards

These hazards exist as a threat that can give away the avatar’s location to a guard’s senses, for example loud flooring (*Thief: The Dark Project*, Looking Glass Studios 1998) or water puddles (*Metal Gear Solid*, Konami, 1998) that emit sounds when trod upon. Other hazards can serve to break the concealment such as motion sensitive lights reacting to the avatar’s movements by lighting up darkened areas. (*Tom Clancy’s Splinter Cell: Pandora Tomorrow*, Ubisoft, 2004) There are also life-threatening hazards such as land mines or floor traps (*Metal Gear Solid*, Konami, 1998).

2.2.3 Security Measures

Security cameras are often used to detect intruders in stealth games (*Metal Gear Solid*, Konami 1998). Other measures also include laser beams (*Tom Clancy’s Splinter Cell: Pandora Tomorrow*, Ubisoft, 2004), motion sensors (*Mark Of The Ninja*, Klei Entertainment, 2012) or a trip alarm device that gives out a loud sound. These threats are designed to trigger the alarm state upon their activation.
2.2.4 Alarm state

The alarm state is usually triggered upon the avatar’s detection by patrolling guards. This phase generally causes an increase in difficulty by changing the guard’s idle patrolling, to a more hostile behavior. In early stealth games, once the player had been detected, only the guards patrolling in the area were aware of the avatar’s location. As technology advanced, guards can trigger general alarms, alerting guards outside the present area.

Many stealth centric games employ alarm states differently. Some have implemented multiple steps that need to be performed by a guard before the alarm can be raised. In *Metal Gear Solid 2: Sons of Liberty* (Konami, 2002) a guard has to first spot the avatar, call his commander on the radio and ask for backup. During this sequence, the player has time to react in order neutralize the guard or destroy the radio to hinder the alarm from being triggered.

*Tom Clancy’s Splinter Cell* (2002, Ubisoft) populates some levels with alarm switches on various walls that the guard has to interact with in order to trigger the alarm. However whether this is the first course of action or to shoot at the player first can vary depending on how the artificial intelligence system works.

Alarm states tend to take on different stages. In *Metal Gear Solid* (Konami, 1998) alarms have multiple stages. The first stage (*Alert Phase*) directs all guards at the avatar’s location. The second stage (*Evasion Phase*) is where guards lose track of the avatar’s location and start actively searching, until a certain time threshold has been reached, which causes the guards to abandon their search.
3. Method

The paper will go through design principles presented by industry professionals. This is done by examining stealth centric game design principles and the design of artificial intelligence. One can develop a general perception of what stealth centric design is by also examining the dynamic relationship of guard and avatar. Level design will then be considered examining the architecture behind level design for stealth games in order to grasp why stealth based levels need to function in a specific way. From this we can extract a set of common design requirements which stealth centric games need to achieve their aesthetic goal, but also the technical requirement that has to be present for stealth play to work.

Games and level designer Randy Smith has held two Game Developers’ Conference (GDC) talks. The first talk, in which he discussed the open-endedness of stealth gameplay, is called “Design Fundamentals of Stealth Gameplay in the Thief series “(2002). The second talk was held in 2006 and goes by the title “Level Building for Stealth Gameplay”. It discusses requirements of stealth levels.

Additionally Christopher W. Totten has a thought-provoking approach to level design, in which he breaks down rooms into spaces which trigger different human emotions depending on certain conditions. While Totten writes about level design in general, his work touches on the relevance of game space to establishing an approach to stealth centric level design.

Tom Leonard and Donald Kehoe both write about Artificial Intelligence design for games. Leonard goes into how an Artificial Intelligence Sensory system works. He worked on artificial intelligence sensory system for Thief: The Dark Project (Looking Glass Studios, 1998). Kehoe writes in a more general manner on Artificial Intelligence design for videogames. However the article also goes through the consequence of using AI state machines that work to shift the guard’s behavior in the game-space relation to the avatar’s actions.

These people’s work will be used in analyzing and organizing the information needed to establish the building blocks of stealth centric games. The paper will also look at camera models as an information gathering tool that the player makes use of during play.
4. Analysis

4.1 Principles of stealth game design

This section covers aspects of game design relevant to stealth centric design, examining the principles relied upon by industry professionals to create stealth centric gameplay, discussing the camera model’s importance as a window of communication and information gathering mechanic. This paper will also look into artificial intelligence design in the case of guards and their behavior.

4.1.1 Designing for stealth centric play

During GDC San Jose in 2006, Randy Smith, Level Builder and Game Designer on Thief (Looking Glass Studios, 1998, 2004) held a talk called “Level Building For Stealth Gameplay”.

Smith defines a common aesthetic goal for stealth centric games as:

[To] create the illusion of a securely guarded area that the player can sneak through by virtue of leveraging their unique abilities and tools to create and exploit security flaws. (Smith, 2006)

In combat centric games the avatar can take punishment from their opponents until their health gauge (partial failure margin) is depleted for the failure condition to be met, while stealth centric game’s partial failure margin’s threshold is considerably smaller as the failure condition is the player’s inability to remain undetected. (Smith, 2006)

Meeting this failure condition of stealth forces the player out of stealth play and into either fleeing or engaging in open conflict.

Figure 1 (Partial failure margin in combat centric games) – (Smith, 2006)
4.1.2 The importance of Artificial Intelligence

To examine this further one needs to examine the avatar’s relation to their *artificial opponents* i.e. the guards.

Stealth poses a considerable problem in the design of artificial opponents for war games. In a game with no stealth, the AI-driven opponent has access to the complete state of the game world; to include stealth, you[the game designer] must restrict the opponent’s knowledge, limit its attention, leave it ignorant of whole regions of the game world. You decide what the AI opponent does and doesn’t know and define what steps it takes if any to gain further information. (Adams, Rollings, 2007)

While this quote originally talks about games in the real-time strategy genre, it is still true for stealth centric games.

The guards are driven by an Artificial Intelligence (AI) and the AI must gather information in order to find the possible whereabouts of the avatar. Generally the guards aren’t constantly searching for the avatar as they often are oblivious to the avatar’s existence. However to be able to react to the avatar’s presence these guards need means to gather information about the possible presence of a threat. The means comes in the form of artificial senses.

In the *Gamastura* feature “Building an AI Sensory System: Examining The Design of Thief: The Dark Project”, Tom Leonard explains the abstracted *senses* of an *artificial intelligence* as the following:

> The term "senses" in game development is a useful metaphor for understanding, designing, and discussing that part of the AI that gathers information about items of interest in the simulated environment of the game. Non-player characters visually presented as humans, animals, or creatures with eyes and ears in a realistic three-dimensional space lend themselves well to the metaphor. (Leonard, 2003)

The gathered information is acquired by the senses in a way that they stimulate the guard’s awareness of other entities in the game (Leonard, 2003).
This allows guards to have a kind of sensory perception of their surroundings, often in the form of vision, hearing and sometimes smell. However the artificial nature of these senses means that they are only stimulated by specific occurrences in the environment.

Eyesight can be mimicked by giving the agent a field of view dictated by a view cone that starts from the guard’s head and stops depending on how far away the guard is meant to see. Multiple view cones can also be added to emulate multiple layers of vision like peripheral and direct vision (Leonard, 2003).

Leonard uses the following figure to explain this concept:

![Figure 3 (Multiple viewcones) – (Leonard, 2003)](image)

Donald Kehoe presents the following solutions for common problems that might occur, when it comes to an AI’s vision in the *Intel Software* article *Designing Artificial Intelligence for Games (Part 1)*.

In more complex games, you may need to account for the player or other entities being hidden by some sort of cover. For this type of game, you may need to perform a ray trace (sometimes referred to as a ray cast) to see whether something has blocked the potential target. A ray trace is a mathematical way of checking if a ray intersects anything, starting from a single point and going in a set direction. (Kehoe, 2009)

Kehoe also goes into the implementation of hearing and smelling senses:

At first blush, it may seem like sound is no different than sight. If you can see an entity, certainly you can hear it, too. It's true that if your agent [guard] has spotted an entity, the agent can actively detect anything that entity does until it is no longer in sight. However, adding an extra level of hearing to your agents can help make sight work more effectively. Tracking the noise that entities make as a level of perception is key to any stealth-based game. (Kehoe, 2009)
Adding the sense of smell to a game is relatively easy: Give each entity in the game a distinct smell number and strength. The strength of the smell determines two factors: the radius of the smell and the size of the scent trail left behind. Active player entities often keep track of their last few positions for a number of reasons [……]. One reason could be to help entities with smell. As the player entity [avatar] updates the trail, the strength of the smell diminishes as the trail grows "cold." When an agent [guard] with smell is updated, it needs to check for smells like it would check for sound [to check the smell] radius and check [if] walls [are blocking the guards smell radius]. (Kehoe, 2009)

Scent trails for the guards from the avatar, allow the guards a more realistic reaction to the avatar’s disappearance. This indicates which direction the guard should start searching for the avatar’s hideout position, or alternatively their direction of escape. Over the course of time while the guard is searching the area, actively looking for the avatar, the trail can start weakening, allowing the guard to transition more naturally to his idle patrol pattern and move back to his post.

Equipping a guard’s AI with senses such as vision, hearing and smell leads to a game system in which the guards are capable of reacting to the avatar’s actions in the environment. Randy Smith calls these actions expressions performed by the player. These expressions are what make the avatar more or less detectable by the guard’s AI-controlled senses, creating a relationship between player and AI which results in stealth play (Smith, 2002).

Smith explains this using the following Venn diagram: (see, fig.4)
The avatar has goals to fulfill and uses his movement expressions (i.e. actions) to reach these objectives. Once the objective is fulfilled the avatar has attained success (in stealth play). However, using these movement expressions, affects their detectability either by producing sound or increasing their visibility to the guards. This happens in relation to the avatar’s location in environment the avatar is treading during play.

Depending on the avatar’s behavior guards has to use their senses to detect and find the appropriate reaction to the avatar’s actions. This is done through the AI component of a guard where they engage in decision making.

### 4.1.3 AI Decision making

Finite-state machines are a method of modeling software’s systems relationships and operations. Desired behaviors tend to be represented by nodes which are interlinked by arrows to display their relationships. These relationships may range from progressive step by step procedures, to more advanced interconnected series of operations.

The core concept behind AI is decision making. To execute these choices, the intelligent system needs to be able to affect the entities using the AI system. (Kehoe, 2009)

A finite-state machine can for example be used as a framework to determine which state of behavior a guard will be in. When certain criteria have been met, a new state of behavior can be entered.

A “state” can represent physical conditions that the entity is in, or it can represent emotional states that the entity can exhibit. In this example, emotional states are nothing like a true AI’s emotional states but predetermined behavior models that fit into the context of the game. (Kehoe, 2009)

These states are triggered by the avatar’s expressions, for example the avatar causing a loud noise while treading on a loud surface.

Kehoe uses the following figure to explain the states a guard can take on in “a game with stealth elements” (Kehoe, 2009)

![Figure 5](image-url)
He continues to explain each state in the figure:

**Idle** –

In this state, the entity is passively standing around or walking along a set path. Perceptions are low. Player sounds are not often checked for. Only if this entity is attacked or “sees” a player directly in front of it will its state change to a higher level of awareness. (Kehoe, 2009)

**Aware** –

This entity is actively searching for intruders. It checks often for the sounds of the player and sees farther and wider than an idle entity. This entity will move to the Intrigued state if it notices something out of place (something to check for), such as open doors, unconscious bodies, or spent bullet casings.” (Kehoe, 2009)

**Intrigued** –

This entity is aware that something is up. To demonstrate this behavior, the entity will abandon its normal post or path and move to areas of interest, such as the aforementioned open doors or bodies. If a player is seen, the entity goes to the Alert state.” (Kehoe, 2009)

**Alert** –

In this state, the entity has become aware of the player and will go through the actions of hunting down the player: moving into range of attack, alerting fellow guards, sounding alarms, and finding cover. When the entity is within range of the enemy, it switches to the Aggressive state.” (Kehoe, 2009)

**Aggressive** –

This is the state where the enemy has engaged in combat with the player. The entity attacks the player when it can and seeks cover between rounds of attack (based on attack cool-downs or reloading). The entity only leaves this state if the enemy is killed (return to normal), if the enemy moves out of firing range (go back to the Alert stage), or if the entity dies (go to the Dead state). If the entity becomes low on health, it may switch to the Fleeing state, depending on the courage of the specific entity.” (Kehoe, 2009)

**Fleeing** –

In this state, the entity tries to run from combat. Depending on the game, there may be a secondary goal of finding health or leaving the play area. When the entity finds health, it may return to the Alert state and resume combat. An entity that “leaves” is merely deleted.’ (Kehoe, 2009)

**Dead** –

In some games, the state of death may not be completely idle. Death or dying can have the entity “cry out,” alerting nearby entities, or go into a knocked-out state, where it can later be revived by a medic (and returned to a state of Alert).’ (Kehoe, 2009)

The field of AI is a complex area of research. AI for games takes on different forms depending on the needs of the game designed, ranging from simple sets of rules for computer-controlled entities to more advanced adaptive systems. Applying AI concepts to games is a necessary way to increase the believability of the virtual characters created in electronic entertainment, but it is not an impossible challenge. (Kehoe, 2009)
The AI state machine of the guard provides the possibility for the guard to adopt different patterns of behavior in the occurrence of different events; whether it be the detection of the avatar or noticing suspicious activity in their vicinity. This allows for a certain dynamic to emerge that meets the aesthetic goal (according to MDA) for the stealth presented at the start of this paper.

the player must move to succeed, but moving makes them detectable, and the AI senses pick up on that, possibly eventually leading to the failure state. (Smith, 2002)

![Diagram](image)

Figure 6 (Basic relationship requirement) – (Smith, 2002)
4.1.4 Overall level design considerations

In the *Gamasutra* article “Designing Better Levels through Human Survival Instincts” (2011) Christopher W. Totten divides the various areas a player navigates in a level into different game spaces that, due to perception of the space and other factors, trigger different human emotions.

The following terms are definitions for spaces chosen for their relevance to this paper.

- **Prospect Space** “Prospect Space describes a spatial condition that is wide open, within which the occupant is exposed to potential enemies.” (Totten, 2011)
- **Refuge Space** “Refuges are places like caves and tree covered areas where early humans could look out into the Prospect spaces of wilderness and evaluate potential threats.” (Totten, 2011)

Totten uses those two spaces to argue that players can make use of a Refuge Space to look for a Secondary Refuge Space and move between these spaces. He then writes about how this concept can be utilized in stealth centric games by creating sequences where players moves from refuge to prospect to secondary refuge in a repeating pattern to comply with the aesthetic of stealth (2011).

I would argue that MGS’s [Metal Gear Solid series] levels are actually based upon the Refuge-Prospect-Secondary Refuge sequence, as the stealth gameplay requires you to move from hiding place to hiding place. This type of gameplay changes mundane environmental elements like corners and lockers into safe places differentiated from the Prospect areas of the level with guards and cameras. (Totten, 2011)

Totten also mentions the concept of **Shadow Space** which “creates the perception that one room is actually two: areas within the Shadow and areas in the light.” Referring to Tom Clancy’s *Splinter Cell* (Ubisoft, 2002) for instance, which makes use of shadows as a means of concealment, creating a second space within the room which functions as a one-way mirror. This means that one can view the content of the outside room from within the shadow; however an AI cannot see the contents of the shadow space while viewing from the outside.

There is a difference between designing levels for combat centric and stealth centric games. In combat centric games it is valid to create challenging rooms and assume all players will experience some degree of failure. However in a stealth centric game, designers can’t assume the player will fail. Instead the game should provide the means for the player to experience little to no failure (Smith, 2006).

This means stealth levels should refrain from having challenges where the security actually is impenetrable, unless it is an area the player is not supposed to have access to. This becomes a balancing act where if security is too tight, the player will be incapable, however if there are too many means that support the player, the game will cease to be stealthy (Smith, 2006).
Randy Smith suggests that having open-ended levels allows the player more flexibility, improvisation and player expression during play sessions. Levels become more open-ended when they permit more paths to be taken when creeping through areas. This makes the level more of a playground and makes the game less of an authored experience for the player. However the larger number of valid routes the player can choose to breach security, the more obstacles that have to be implemented in order to balance the player experience (Smith, 2006).

The following figure (fig.7) shows a room where it is possible for the player to take multiple paths in order to pass through. The room contains two guards, a patch of loud flooring and two torches that light up certain areas of the room. These are the main challenges the player has to deal with in this room. Shadows are however present, which the avatar can use for concealment. The challenges work to repel the player away, while he finds his way around the environment and the shadow serves as means to avoid detection. However which paths are apparent to the player is very dependent on how the environment is communicated through the camera model used in the game.
4.2 Perspectives: Camera models and player overview

The perspective a game is presented from affects the way the game is going to be played. It decides how all information that is going to be transmitted to the player is presented and possibly interpreted. The following sections will discuss how camera models and user interfaces are applied to stealth centric games, in order to form an overview of the situations the player is going to work with while they advance from one challenge to another (Adams, 2007, pp. 241-248).

4.2.1.1 Camera Models

This paper uses the following explanation of the word camera model:

If a game includes a simulated physical space, or *game world*, then it almost certainly uses graphics to display that space to the player. The user interface must display the space from a particular angle or point of view. Designers usually imagine that a hypothetical camera is pointed at the virtual space, creating the image that the player sees. The system that controls the behavior of this imaginary camera is called the *camera model*. (Adams, Rollings, 2009)

4.2.1.2 Third person perspective

The third person perspective model allows the player to see his avatar’s physical form in relation to the game space, while playing. There are two common variations present in stealth centric games. One is a so called the *top-down* view. This variation places the camera far above the avatar allowing the player to see a vast part of the environment at the same time. The camera in some games also follows the avatar along the environment. (Metal Gear Solid, Konami, 1998) From a design standpoint this camera system allows the player to quickly get an overview of the environment the avatar is traversing. For example; the player can easily observe the patrol patterns carried out by guards in relation to themselves as an object within the game space (Adams, Rollings, pp.256).

How far the player can see is dependent on the camera’s distance from the player, generally, the further away, the more one can see. This can also mean that the perspective limits the player’s ability to anticipate challenges outside the camera’s field of view. This leads to the possibility of the player feeling unfairly treated if a guard is able to attack from a distance beyond the top-down camera’s view (Adams, Rollings, pp.256).

The second variant is called a *3D third-person* view, and is essentially a camera system where the camera rotates around the player avatar, using the player avatar as a center of gravity. (Tom Clancy’s Splinter Cell, Ubisoft, 2002) The camera’s default position is usually somewhat over the avatar, in some instances they are set from an over-the-shoulder-view allowing the player to view what is in front of the avatar. This model also allows the player to see a little of their near surroundings. In this camera model the player often controls the rotation of the camera deciding which angle is the most appropriate at any given time during play (Adams, Rollings, pp.457).
4.2.1.3 First person perspective

In this camera model the camera is placed in the player avatar’s head, so as to give the illusion that the player is seeing the world from the avatar’s “eyes”. This model gives the player the ability to see what is happening ahead, allowing the game to have threats such as snipers in their levels without the player feeling at a disadvantage. As the player turns the camera as if they were the avatar’s body, the player can focus on what they think is important. While it can seem realistic this view only lets the player see things in the direction the avatar is facing (Adams, Rollings, pp.242, 456, 2007).

In a first person game, the player's sense of body is weak, and the player seen by an opponent they do not see often feels cheated. (Leonard, 2003)

This leaves the player vulnerable to threats coming from all other directions, than the one they are facing. This leads the game to be designed in order that all threats must come from one general direction that the player should focus on in order for the player to not feel mistreated (Adams, Rollings, pp.242, 456, 2007).

4.2.2 Context sensitive camera placement

A context sensitive camera model is a camera that views the area the player is present in from a position that changes depending on where the avatar is in the region. The camera moves and rotates to keep the avatar in view while the player is traveling through the environment. In some games it does this smoothly, but the camera can suddenly snap to another camera position as well, depending on the game’s needs. (Metal Gear Solid 2: Sons Of Liberty, Konami, 2002) “If the camera moves –especially if it jumps suddenly—the player will become disoriented and is likely to make mistakes.” (Adams, Rollings, pp.458, 2007)

4.2.3 Mixing and matching camera models

It is common for stealth centric games to switch between multiple camera models. This can be done in order to compensate for the main camera model’s limitations. One common example is the use of a 3D third person camera model for navigation around environments, while allowing the player to switch into a first person camera model for precision aiming with firearms (Metal Gear Solid 2: Sons of Liberty, Konami, 2001) or as a way of observing what is happening further ahead, which can allow the player to gauge threats early on (Metal Gear Solid, Konami, 1998).
Other examples involve switching from first to third person camera model when the avatar is pressed against the wall for cover (*Deus Ex: Human Revolution*, Square Enix, 2011). This can allow the player to peek around the edges of a wall without exposing themselves to threats. There are instances where games switch from a third person camera model to a context sensitive camera model. This may happen for example when the player enters a narrow space where a 3D rotating camera will not function well and can potentially get stuck behind a wall (*Assassin’s Creed*, Ubisoft, 2007).

### 4.2.4 Acquiring information beyond the camera model

According to the book; Fundamentals Of Game Design, the challenge type of stealth centric games is “[to]avoid being seen” (*Adams, Rollings, 2007*) This means that in order for the player to avoid being spotted by guards, the player plans and observes his surroundings before attempting to move through an area to another. Information gathering is required before the player takes action. The camera models presented above allow the player to comprehend most of the information needed for the player’s progression. However when the player has to move around a blind corner, the potential threats are not obviously exposed to the player. This gap of information can be remedied by providing the player with tools, to help them gain information from areas they cannot directly observe from a comfortable distance.

A closed door can represent a concealed area for the player that they eventually have to enter. Many mechanics have been designed to address this, all with more or less the same result; being able to see what is happening on the other side, by letting the player peek through a key hole, giving the player limited visibility of the room ahead (*Hitman 2: Silent Assassin*, IO, 2002).

Other solutions can be slipping in a camera under the door, allowing the player to look in from a low profile and adjust the camera’s orientation to reveal more of the surroundings (*Tom Clancy’s Splinter Cell*, Ubisoft, 2002). There are tools such as goggles that allow the player to see through opaque objects or detect heat signatures from guards (*Tom Clancy’s Splinter Cell*, Ubisoft, 2002).

On some occasions the camera mechanics have evolved into a sticky-camera (sticks to walls) that can be launched into high risk detection areas, where the player has little to no overview before entering.

As stealth games tend to have more open levels, this becomes important. Mini-maps can be used to compensate for the camera model’s limitations in three dimensional spaces, as it can be difficult for a camera model to address all the information coming from all three dimensions at once.
5. Discussion

5.1 The AI guard and player relation

The patrolling guard exists to counter the avatar’s ability to move freely through the game space. Using their *senses* the guards are capable of covering territory directly through their vision and indirectly through other *senses* such as hearing and smelling. This is a general requirement for stealth centric design to work.

A practical solution to making an exceedingly static stealth centric space is to populate the space in question with guards. However it is important not to create a disproportionate level of challenge outside of the play style’s aesthetic ambition.

Guards have a variety of states that reflect their local game space status, at any given time. These are built purely around the avatar’s actions. These actions are interpreted through sensory systems in the guards, resulting in action causation.

> A game sensory system must be designed in a way that is subservient to the game design and efficient in implementation. The senses need only be as sophisticated as is needed to be entertaining and robust. The result of their work must be perceivable and understandable by the player. (Leonard, 2003)

A stealth game is in need of an AI oblivious to the complete state of the game. These AI agents gather information using their own means. AI has a limited perception of their surroundings, but by allowing them to gather bits of information and put them together they form a decision supporting what they have perceived through their own means. This is important because the player needs to perceive themselves as an unknown threat to the security of the AI dominated game space.

When the guard detects suspicious activity in their vicinity their *senses* become more vigilant. However this suspicion has to be updated with new stimuli in order not to fade. This indicates that the *senses* of guards can be taken advantage of in stealth centric games as a means of advancement. Derailing the guard from the patrol path by creating distracting stimuli for the guards to avert their attention is an example of this.
5.2 Overall Sequence of stealth play

Totten’s argues that stealth centric games (using Metal Gear Solid, Konami, 1998) have the play sequence of Refuge-Prospect-Secondary Refuge (2011); Randy Smith puts forth a method relevant to this when he talks about designing the pacing of such sequences. (See, fig.8) he uses two examples for this, in the example on the left side, the player slowly and cautiously advances from refuge to refuge, while avoiding and always keeping an eye at the guard’s patrol route. In right side example, the player has to anticipate each approach beforehand, and move swiftly between stops to avoid being detected by either the guard patrolling in front of them or the one behind them (2006).

![Diagram of stealth play sequence]

**Figure 8 (Methods of pacing) – (Smith, 2006) note: image has been edited for the purpose of this paper.**

5.3 Beyond shadow space

The concept of shadow space can take on different forms beyond the concept of a static shadow where the player can have an outlook, without being in danger of detection. In *Assassin’s creed II* (Ubisoft, 2009) crowds are used by the avatar to blend in among civilians. This attribute essentially makes crowds act as a mobile shadow space. The use of camouflage in *Metal Gear Solid 3: Snake Eater* (Kojima Productions, 2004) can also be seen as a form of shadow space, where the avatar can use the majority of the environment as locations to avoid detection. In short, shadow space provides concealment which can take on different forms. The avatar can be partially concealed from the AI as well. Both *Tom Clancy’s Splinter Cell* (Ubisoft, 2002) and *Metal Gear Solid 3: Snake Eater* (Kojima Production, 2004) have an indicator in place that communicate to which degree the avatar is concealed from the guards.
5.4 Interconnecting challenges

There seems to be a general understanding that players need to plan every encounter by observing what is ahead of them. However, creating open-ended level structures may also cause many of these challenges to interconnect. For example, in fig. 7 reaching the middle of the room does not only mean the possibility of dealing with loud flooring as well as revealing light sources. Additionally, the two guards patrolling are also a threat as they survey the area. This means that the player is (depending on the patrol pattern) in constant danger of detection from two sides. These challenges can, however, be avoided by using the shadowy side corridor, by doing so the players avoid the interconnecting challenges i.e. the intertwined guards, loud flooring and revealing lights.
6. Conclusion

The following list outlines what this paper has come to find are the essential core components of stealth centric games:

- **Movement**: in stealth centric games the purpose of movement goes beyond the avatar’s navigation of the game space. Multiple movement modes are often implemented allowing the player a higher degree of control over the avatar’s detectability.

- **Cover or concealment**: to counter visibility the game needs have an environment that provides the avatar with the means for hiding, whether be it crowds in the city street to blend among or a crate to hide behind.

- **Guards or Security measures**: These are the main adversaries the avatar has to deal with; they need to be present enough for the area to seem securely guarded, but not to the point where security is impossible to breach.

- **Alarm state**: is the failure condition of stealth play and sometimes the game itself, signaling the need for the player to decide whether to flee or fight.

- **Basic sensory system**: as the adversaries of these games do not have access to the complete state of the game, they need to have some means of detecting the avatar’s presence through at least binary detection using vision, hearing or smelling.

- **Aesthetically demanding levels**: The levels in these games have to be constructed with all of the mentioned above in mind, abiding by the aesthetics of stealth which is: “[To] create the illusion of a securely guarded area that the player can sneak through by virtue of leveraging their unique abilities and tools to create and exploit security flaws” (Smith, 2006).

With this, one can plausibly develop a stealth centric game that can make use of the aesthetic that defines this type of games. For example one can apply the aesthetic of stealth to a game where the player takes control of a car, avoiding speed cameras and police patrols. This is significant because it shows that the aesthetic of stealth can take on a variety of settings and themes, as long as the players themselves act as hidden threats to the current alarm state, thereby avoiding detection.

These listed components however, are only a base set over which all other discussed attributes of stealth can be added to widen the partial failure margin in play, thus expanding the grey zone where the player may operate while adhering to the aesthetic of stealth centrism.
Further Research

Going beyond this paper’s conclusions, it is possible for us to further our insights into stealth games. Through examining further other stealth centric games we could come to learn more about how these games are played.

As these games tend to be open-ended, allowing the player to use multiple approaches to progress through challenges. This also means that there is bound to be a variety of applicable play styles to stealth centric games. The potential of identification of these play styles will be explored in my future work through analysis of contemporary titles released within a set of years. By doing so, one can also reach a reliable perception of the current state of this type of games, expanding the scope of possibilities for future development of stealth centric games.
Glossary

Avatar: “A fictional character in a game with whom the player identifies as the personification of herself with in the game world. The character need not be human; it may even be a vehicle.” (Adams, Rollings glossary G1, 2007)

Game world: “An imaginary universe in which the events of the game takes place. Most computer game worlds are simulated two- and three-dimensional spaces containing characters and objects.” (Adams, Rollings glossary G5, 2007)

Game space: The space in which the player interacts with other actors or entities directly.

Stealth: The themed aesthetic goal set forth by a designer, for a player to achieve within a game space; to act and approach challenges in a covert manner.

Sneak: The procedure of movement to the context of Stealth; to move in a quiet and cautious fashion.

Avatar profile: This refers to the avatar’s current visual composition; be it respective of stance, shape or color.

Mini-maps: “A small, dynamically updated map of the game world, usually displayed in the corner of the screen in the primary gameplay mode, for quick reference. Also sometimes called radar screen” (Adams, Rollings, glossary G7, 2007)

Entity: “A datum or collection of data that describes some objects, character, quantity, or state of affairs” (Adams, Rollings, glossary G4, 2007)

Boss encounter: “A large and particularly difficult challenge that must be overcome, typically the last one required in order to complete a level of a game” (Adams, Rollings, glossary G2, 2007)
Bibliography


Interactive, E. (Publisher). (2001). Hitman; Codename 47 [Video Game].


Konami (Publisher). (1987). Metal Gear [Video Game].

Konami (Publisher). (1998). Metal Gear Solid [Video Game].

Konami (Publisher). (2002). Metal Gear Solid 2: Sons of Liberty [Video Game].


Studios, L. G. (Publisher). (2000). Thief II:The Metal Age [Video Game].

Ubisoft (Publisher). (2002). *Tom Clancy’s Splinter Cell* [Video Game].

Ubisoft (Publisher). (2002). *Tom Clancy’s Splinter Cell: Pandora Tomorrow* [Video Game].

Ubisoft (Publisher). (2007). *Assassin’s Creed* [Video Game].

Ubisoft (Publisher). (2009). *Assassin’s Creed II* [Video Game].