

From Level Up! The Guide to Great Video Game Design, 2nd Ed. By Scott Rogers, John Wiley and Sons, Ltd., 2014.

[Taken from Chapter 6: The Three Cs, Part 2: Camera]

Get It Right: Camera Views, Guides



DO YOU HEAR that crashing sound? That is the sound of a video game controller being thrown through a 50" 1080p HDTV plasma panel with a 600Hz subfield drive. And why was this fine piece of technology utterly

pulverized? Because your game has a really bad camera.

Did you know that over 1 billion TVs are destroyed a year because of really bad game cameras? Nothing will cause players to stop playing your game faster than a poor camera. This is why it is so important to get it right.

Get It Right: Camera Views

Choosing the right camera for your game not only is very important for determining how to program the camera, but also impacts how you design your game, map your controls, and create your artwork. It's pretty common for a game to have more than one style of camera, but you should stick with one "main" camera style for the majority of your gameplay and use other camera views only for specific gameplay situations.

A **static camera** does not change position, focal distance, or field of view and stays fixed onto a single screen, location, and image. The earliest video games used static cameras because (a) the

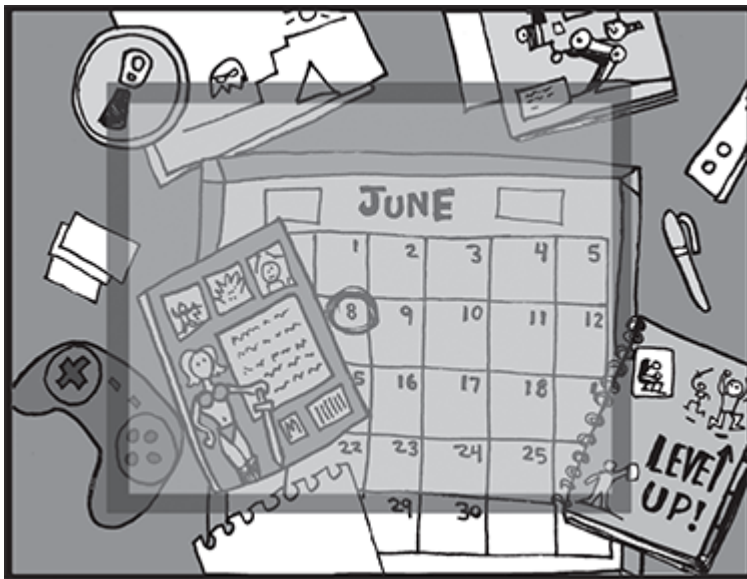


scrolling camera hadn't been invented yet (duh!) and (b) it was easier for players to track all the elements if they were kept to one screen. Early video gamers just weren't that sophisticated. But it didn't take long for them to adapt and evolve ...

Even though the static camera's roots are old school, this view is still very popular in many current games like *Candy Crush Saga*, *Crabtron* and *Plants Vs. Zombies 2*. A clever use of the static camera is to set the mood as found in early survival horror titles like *Alone in the Dark* and *Resident Evil*. These developers not only used the static camera shot to represent a single room, but also used it to set up the camera for maximum effectiveness. It maximizes the game's artwork by requiring only art that would be seen only "from certain angles." An item in a game world that is viewed from only one angle doesn't need a backside, which saves on production and object rendering time.

Another advantage is that you can easily use it to set up story-related events in your game world because you don't run the risk of players looking the other way when it happens. However, you have to be careful with static screens because they aren't very dynamic. Make sure to compensate for this problem with lots of animation and effects to keep your screen lively.

If you aren't satisfied with a camera that stays put, you can always ask your programmer nicely to make it into a **scrolling camera** instead.



Pretend you are looking down at a desktop. Or use this picture if you are bad at pretending. On this hypothetical desktop, you can interact with all the elements on the desk, but hey—you just can't find your pen. In the picture, what you can see is represented by the gray box. By moving or "**scrolling**" the camera (in this case your eyes) to another part of the

desk—voilà!—you find your missing pen beside a book. Amazing!

A scrollable camera offers all the advantages of static camera but with the added advantages of (a) movement, which keeps players engaged in the act of moving the camera, and (b) your ability to hide stuff off-screen or reveal it in a big dramatic way. This is why you will find it being used in many old-school adventure games like *Day of the Tentacle* or the *Monkey Island* series. If you use a scrollable

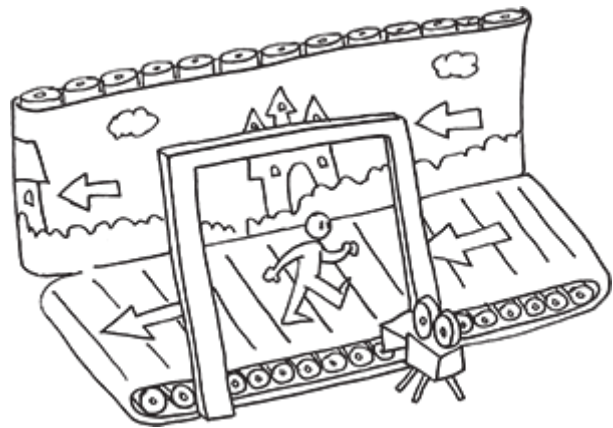
camera with a God-mode or isometric view (which I talk about later), you can simulate a table top to simulate miniature games. This is why a scrollable camera is used in RTS and dungeon crawl games like *Dawn of War* and *Diablo III*. Make sure your controls for moving your camera are simple and relative to your players' controllers. You don't need anything fancy to move a camera around.

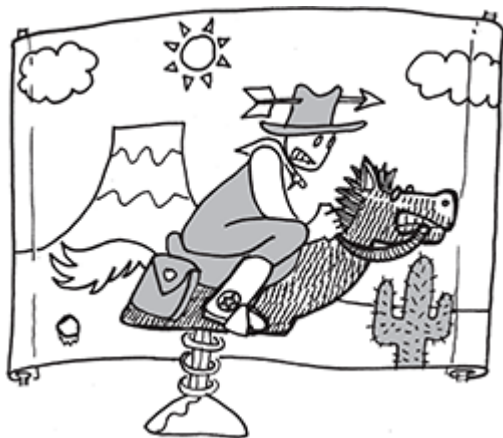
Work with your programmer to **tune the hydraulics** of your camera—the speed at which the camera accelerates/decelerates. The wrong speed can cause the camera to drift past the character or abruptly stop short, which can be very frustrating as the player goes through a process of overshooting back and forth, a condition that eventually leads to insanity and/or a destroyed monitor. Conversely, don't make your camera scroll too slowly. This can be particularly catastrophic in a game in which your little army's platoons are in danger of being wiped out by enemy tanks and your scrolling camera is too slow to get to them in time. Oh, the humanity!

Why not let the players decide what speed they want the screen to scroll? It's an uncommon option in RTS and strategy games, but it could work for any genre of game with a scrolling camera. I suggest making this option a sliding scale with several speeds to choose from. Just "fast" and "slow" won't cut it.

In the beginning, there was the static screen. The static screen was fine for *Invaders from Space* and the Kongs of *Donkey*. And then game players cried out for more. So in 1982, the great Irem descended from the heavens in *Moon Patrol's* purple moon buggy and introduced **parallax scrolling** to the video games world.

As a parallax scrolling camera moves, the world moves with it. This camera view revolutionized video games, allowing game developers to create longer and deeper game worlds in which to play. There are two different ways you can treat parallax scrolling. First is plain ol' scrolling. The camera is controlled by a player's movement: the player essentially stays in the center of the screen as the world moves past him just like in those old-fashioned western films. Giddyup!





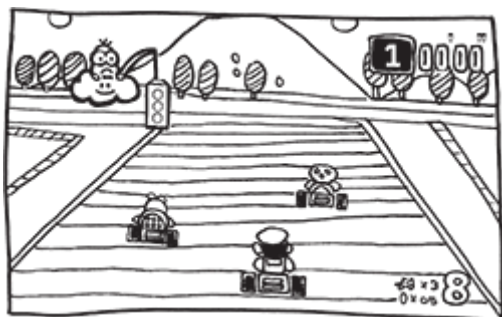
When using this type of scrolling, be careful to play out how your game level loads because your players may be able to “outrun the load.” Always play your game level backward to make sure that your players can’t break your game.

Another type of scrolling is the **forced scroll**. Players are forced to “keep up” with a scrolling camera, which is why it was first used on driving and flying games like *Moon Patrol* or

Scramble. It can be used with 2-D, 2-D parallax and 3-D scrolling games. It became popular with first person shooters like *Operation Wolf* and third person rail shooters like *Panzer Dragoon* and was later used for “chase” sequences like those found in *Crash Bandicoot*. More often than not, if a player fails to keep up with the camera, something horrible (such as death) happens to him. This makes a forced scroll camera great for gameplay where you really want to put pressure on players, but keep in mind that you don’t want to use it consistently in a game—that is, unless your whole game is based on this idea.



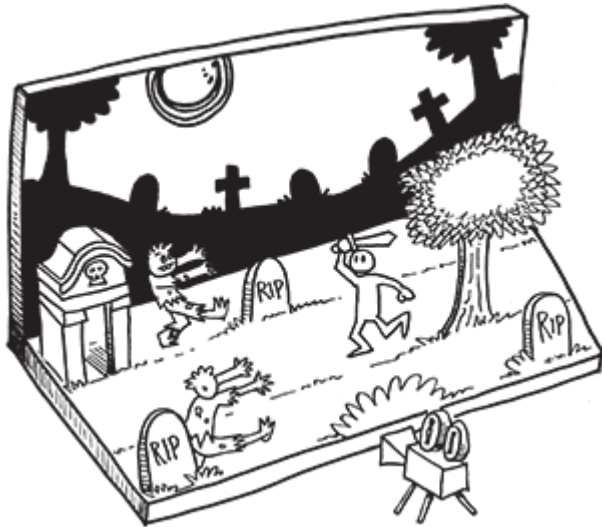
Scrolling games dominated home video games throughout the 1990s (really—I was there). There were tons of them! When you have a bunch of people making a genre of game over and over again, innovation eventually sneaks in. And it happened. Twice, in fact.



The first was **Mode 7**, named after the seventh (out of eight) background layer on the Super Nintendo entertainment system. The hardware would convert 2-D art into a 3-D plane which, when scrolled, created the illusion of a background infinitely moving toward or away from the horizon. Add a forward-facing or rear-facing sprite, and you would create the illusion

of a car or character traveling toward or away from the screen. Excellent examples of Mode 7 can be seen in *Mario Kart*, *F-Zero*, and *Super Star Wars*.

However, designing a level for Mode 7 gameplay can be tricky because your level has no true back wall, only a horizon the players can never reach. While technology has advanced to allow programmers to easily create 3-D worlds without any special graphic mode, the term still remains in use by some (albeit ancient) game developers.



In addition to scrolling the camera, programmers found inspiration from the **multiplane camera** used in traditional animation. This camera gives the illusion of depth by zooming the camera toward and away from the screen. By having a camera that tracked in and out of the **Z-axis**, developers were able to create level designs with parallel paths. Games like Disney's *Hercules Action Game* (Virgin Interactive, 1997) used the multiplane camera to create **bidimensional**

gameplay, the forerunner to what is known as **two and a half D**. A side effect of the zooming effect in bidimensional games was severe pixilation that occurred when the camera zoomed in on a nonscaling sprite. You can still find this effect being intentionally imitated in “retro”-style games.

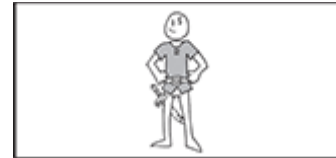
Camera Shot Guide

Now that you've seen all the ways a camera can present the game, it's time to see all the ways the camera can present the story as well. Let's go to the 5-second film school and find out how to set up your camera to get the best shot—just like those Hollywood professionals!

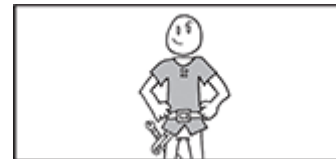
Extreme wide shot (EWS)—This shot shows a character or location from a very far distance. It is perfect for showing castles looming in the distance or a planet-killing space station in orbit.



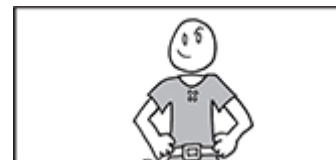
Very wide shot (VWS)—This shot is closer than an extreme wide shot so that you can make out some details. It is usually used for establishing shots of buildings or other large things like space ships or to set the tone that the player is stranded out at sea or in a desert.



Wide shot (WS)—The entire subject (be it a car or person) can be seen in frame on a wide shot. It is usually used when first establishing a main character or vehicle so the player can get a good look at it in its entirety.



Medium shot (MS)—About half of the figure or the subject can be seen in frame—usually your character from the waist up. This means your character doesn't need to wear pants that day.



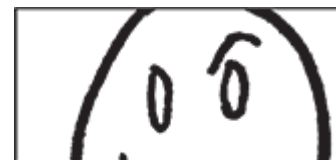
Medium close-up (MCU)—Also called a “head and shoulders shot,” this view is most commonly used when a character is talking. Make sure to animate hands to keep the character on the screen lively.



Close-up (CU)—This is also called a “head shot”: the camera is tight into a character's face to show expression. When you get this close to a CG character model, you start to see flaws (such as in the interiors of mouths or close views on textures). I recommend using this shot sparingly.



Extreme close-up (ECU)—Wham! Right up the nose with this one. This shot is great for focusing on the expression in eyes like in spaghetti westerns and old horror movies. Or you can use it to show details on objects such as puzzle clues or even the puzzles themselves.



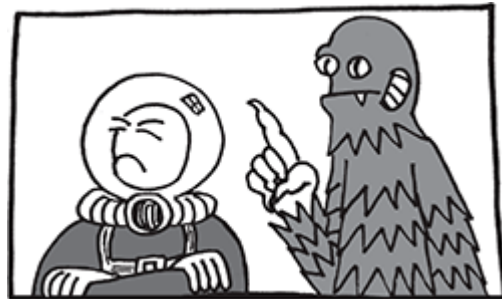
Cutaway—You know when the hero says “I need to get that magic sword” and the next shot is a magic sword? That’s a cutaway. A cutaway can be used for a character reaction shot too.



Cut in—Here, our hero says “I need to examine this clue,” and then the camera shows a close-up detail of the clue. That’s a cut in.



Two shot—This is called a two shot because it features two elements (usually talking characters) shown on-screen at the same time.



Over-the-shoulder shot (OSS)—This shot is taken from over the shoulder of a character. It’s a good opportunity to show hidden things too, such as characters revealing that they have a gun strapped to their back or are crossing their fingers as they promise not to kill the bad guy.



Noddy—In this shot, a character is reacting to what someone else is saying (“nodding” in response to the speaker). You see this shot a lot with news interviews.



Point-of-view shot (POV)—This shot is from the perspective of someone or something. It is usually shown from the eyes of the player but can be from the point of view of a watching enemy, a floating power-up, whatever you want!



Camera Movement Guide

Now that you know what kind of camera you are using and what your shot is going to look like, let’s place the camera to make things look as cool as possible:

Eye level—The camera is aiming level at the eyes of your subject.



High angle—The camera is above the subject, looking down on it. This angle can make things look less impressive. A good shot for showing elements in relationship to each other.



Low angle—The camera is below the subject, looking up at it. This makes things look more menacing or impressive. Great for boss fights.



Worm's-eye view—The camera is literally on the floor looking up, as if a worm were watching the action.



Bird's-eye view—This shot is taken from high up in the sky, as if a bird were watching the action.



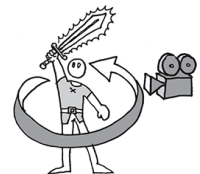
Dutch tilt—We did this trick in *Maximo: Ghosts to Glory*. We wanted our in-game camera to feel slightly creepy or wacky, like shots you see in a Sam Raimi horror movie or the 1960s *Batman* TV show. Tilt the camera so everything seems to be cockeyed. If you do a Dutch tilt subtly, it has a great effect on the player, who realizes something is wrong but isn't sure what. If you do it severely, it really makes things feel screwed up.



Camera Movement Guide

Moving the camera is an art all unto itself. Here are the most common ways to move a camera. See whether you can incorporate these moves into your game camera to make it feel more cinematic:

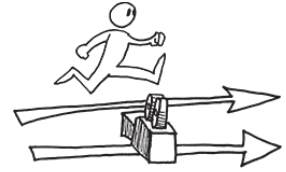
Arc—The camera follows or dollies around a subject in an arc. A common technique is to have the camera arc around the player for 360 degrees if something amazing or wondrous is happening to the character (for example, he's just gained a new power).



Dolly zoom—The camera adjusts the focal length but is moved forward or back to keep the subject the same size on camera. You see this shot a lot in Steven Spielberg movies where a character is amazed by something or comes to a realization that something bad is going to happen.



Follow—The camera moves with the subject. Depending on the style of your cutscene, you can try to make your follow a little shaky as if it were taken by a handheld camera.



Pedestal—The camera moves up to match the subject. Like a tracking shot, but vertical. Having the camera pedestal move past objects in the foreground helps add to the illusion of speed, especially if you are trying to show something rise up suddenly or powerfully.



Pan—The camera moves to the left or the right. Play around with your pans: move them around and past objects. Place items in the foreground to make more interesting shots.



Tilt—The camera's focus is moved up or down, but the camera's position stays the same. Effects like lens flare can make a tilt more interesting.



Dolly—The camera is moved smoothly toward or away from the subject. This is also called a tracking or crab shot. Speed can really make a dolly more interesting—a slow crawl if something is mysterious or suspenseful, or really rocket forward if something is dangerous or dramatic. Play around with starting and ending your tracking before your subject starts moving to make things feel more dynamic.



Zoom—The focal length of the lens changes, giving the illusion of the camera moving. Be careful not to zoom in through items (sorting) or zoom too close to characters or world objects that aren't very detailed. Seeing textures go from fine to pixelated breaks reality for the viewer.

