Video: Observer Motion and 3D Layout

[00:01] [slide 1] The next pair of videos addresses an important way that we use information about image motion, which is to understand our own movement relative to the environment, and to recover the three-dimensional layout of objects in the scene. This first video provides some essential background for the problem and the second video offers a possible solution.

[00:22] When we watch a movie scene like Luke and Leia in the forest, we're sometimes presented with the viewpoint of the person moving through the scene, like in the snapshot on the right, and we vividly sense the direction of motion of the observer, which is directly toward the tree in this case. At this moment, the image is expanding rapidly in our field of view, and there's a point in the center that doesn't move - it's the point that we're heading directly toward. In the upper left corner, shows a picture where we can imagine an airplane landing on a runway, and at this moment in time, it's heading directly toward the location that I highlighted with the red dot in the middle. The arrows show the movement in the image away from this point, which is referred to as the focus of expansion. Visual features further away from the focus of expansion move faster in the image, as conveyed by larger arrows in the diagram, and in the forest snapshot, there's more blurring in the outer parts of the image. Relative motion also enables us to sense the relative depths of surfaces in the scene, as illustrated in the diagram in the upper right, of a person in a moving train. Closer objects move faster across the observer's field of view, and more distant surfaces move more slowly. In this video, we'll explore the connection between image motion and the movement of the observer and depth in the scene. In the next video, we'll describe a simple method to compute observer motion and scene structure from image motion.

[02:15] [slide 2] The instantaneous movement of the observer can be described as the combination of a translation in three dimensions and rotation around the three coordinate axes. We'll use the parameters Tx, Ty, Tz to denote the movement of the observer in the x, y, and z directions, and the parameters Rx, Ry, and Rz for the amount of rotation around the x, y, and z axes, as shown in the diagram. For the observer motion problem, we'll assume a coordinate frame that's sitting on the eye, but this picture spreads things out in a way that helps you understand the meaning of each parameter more easily. Also for each location (x,y) in the image, we can specify a distance Z to the nearest surface in that visual direction. Solving the observer motion problem means starting with an image velocity field like the one shown on the left here, and computing the 6 parameters of the observer's motion and the depth of the surface at each location in the image. A key part of this problem is recovering the direction of motion of the observer, referred to as the observer's heading. Especially when you're engaged in high-speed activities, like driving on a highway, skiing down a steep slope, riding a speed bike through the forest, you need to control your heading direction very precisely, because a lapse in judgment can lead to disaster in a very short time.

[03:49] [slide 3] So we can ask, how good are human observers at judging heading? In perceptual studies of this ability by Bill Warren and his colleagues, subjects viewed computer

displays of the simulated movement of an observer along a ground plane of randomly positioned dots, toward a location on the horizon. The picture on the bottom shows each dot with an attached line segment that indicates its direction and speed of movement over a brief time window. In each trial of the experiment, a blank screen was first displayed, with a single mark on the screen that served as a fixation point. The subject fixed their eyes on this mark and the dots appeared and moved for a brief time. After the motion stopped, a vertical bar appeared on the horizon near the heading point used to generate the pattern of movement of the dots. The observer had to indicate whether the simulated heading in the motion display was to the left or right of this bar. The empirical question is, how small can we make the angle between the simulated heading and the direction to the vertical bar, in order for the subject to judge reliably, whether the movement was to the left or right of the bar. The answer is that we can perform this task reliably when the simulated heading direction is only 1 or 2 degrees of visual angle away from the direction of the bar. To get a feel for just how accurately we can sense our heading, the blue and red arrows in the bottom right corner have a difference in direction of about 2 deg. So we can distinguish between heading in the blue direction vs. heading in the red direction, at least 75% of the time. That's incredibly high precision. But, heading direction is easy to compute in the scenario I described here - why do I say that?

[06:15] [slide 4] Let's say the black dot here is the observer's true heading point. If the observer is just translating through the scene, there will be an expanding pattern of motion in the image, directed away from the focus of expansion at the observer's heading point. The image velocities will all lie along this set of dashed green lines here that intersect at the focus of expansion. So a possible strategy for determining the heading point is to compute this point of intersection of the lines containing all the image velocities. There's likely to be error in our computed image motions, so our lines of motion won't perfectly intersect at one point, so we can find a point that best captures the intersections of lots of these lines, as we did when we were solving the motion measurement problem. But, when we're doing a high-speed activity like driving on a highway, we typically don't keep our eyes glued on the heading point, we're looking around and tracking things with our eyes, like other cars, highway signs, things on the side of the road. And in this scenario, we're rotating our eyes, which makes the pattern of image motion on our eye more complex. Can we still recover our heading direction accurately when the eyes are rotating?

[07:47] [slide 5] This situation was simulated in the perceptual experiments in two ways. In one case, before the dots started moving, one of the dots on the ground was highlighted, and the subject was asked to track that dot when the motion started. It might, for example, be the dot here that I drew with the red arrow. In this situation, the observer is physically rotating their eyes as the dots move, and our visual system can sense this rotation of the eyes. But note that as the dot moves on the display, and the observer tracks it, the dot remains stationary on the eye, right in the center of the observer's field of view, and everything else moves relative to that. Now consider a second case, where a point is again highlighted at the start of the trial, like the red dot in the bottom display here, and the subject is again asked to keep their eyes fixed on

that dot, but this point then remains stationary on the display, and the rest of the dots in the scene move as if the eye were rotating. In this case, the experimenters are simulating the pattern of movement that's created when the observer translates while rotating their eyes, but the observer's eye is actually stationary, so we don't directly sense any rotation of the eye in this case. So they're trying to determine just from purely visual information, without the information about the physical rotation of the eyes, can we still recover our direction of heading accurately. One of the things you can see from the line segments here in the bottom, is that the pattern of image motion is more complex in this situation, it's not just expanding outward from the heading point, which is still on the horizon near the vertical bar. So can we still recover the true heading direction from this more complex pattern of image motion? The answer is mostly yes, the simulated rotation can't be too large, but we can still recover our heading with the same accuracy of about 2 deg. So now let's return to the observer motion problem with this information in hand.

[10:27] [slide 6] We said that we want to compute the parameters of the observer's translation and rotation, and the depth at each image location, from the image motions. Here I show the image motions that arise from a particular translation and rotation of the observer relative to a scene where the observer is moving toward a wall that has a square object floating out in front of the wall. The upper left diagram shows the velocities at a grid of locations that would be generated for a particular speed of translation toward the wall. The red dot is the heading point, and the red dashed square is the outline of the surface in front. The translation on its own generates an expanding pattern of velocities away from a focus of expansion at the heading point, and the speeds of image motion increase further away from the heading point, as we saw before. If you look carefully, you'll also see a jump in speed at the border of the square surface. We saw earlier that surfaces closer to the observer move with higher speed, so in the vicinity of the border, the speed of movement of the square in front is higher than the speed of movement right next door on the wall in back. The right figure on the top shows the image motions that would be generated if the observer were just rotating their eyes to the left - the surface texture would shift to the right in the image as the eye rotates to the left, as the little segments at each point show here. If the observer translates toward the wall while rotating their eyes at the same time, the resulting motion on the eyes is the sum of the motion due to the translation and the motion due to rotation. This resulting pattern of motion is shown on the bottom. At each location on the grid, the velocity vector resulting from the observer's translation was added to the velocity vector resulting from the observer's rotation, to yield the final velocity displayed at that location.

[12:57] I'll show this vector addition for one sample location taken from the upper right corner of the image, and I'm going to show it to the side. Imagine that at a particular location marked with the black dot here, the observer's translation gives rise to the velocity shown as the red vector, which I'll labeled vT. Suppose the rotation of the eyes adds this green velocity here, labeled vR. The final motion at this location is the sum of these two vectors. To add the vectors, we'll redraw the green vector at the endpoint of the red vector, and the sum is the vector from the start of the

red vector to the end of the green vector that I copied, which is the blue vector that's labeled vT + vR. So now let's observe the final velocity field here more closely. The true heading point is again marked with the red dot, and the border of the square is shown with the dashed red lines. There are two locations here that look like a focus of expansion, that I circled in purple - one is near the left side of the square and the other is on the left side of the image. But these locations don't correspond to the observer's heading point - they're places where the velocity due to the observer's translation and the velocity due to their rotation happen to cancel out and produce a point of no motion. You can also see here around the borders of the object that there are differences in velocity from one side of the border to the other. Any strategy for finding the observer's heading that just looks for an expanding pattern of motion would fail here, because we have locations here that look like a focus of expansion that aren't the observer's true heading point. The aim of the solution to the observer motion problem is still to compute the parameters of motion of the observer and the depths of surfaces everywhere, but we need to do this in the general situation where the observer is both translating and rotating, which can give us a more complex motion pattern, like the one shown at the bottom here.

[15:35] [slide 7] To accomplish this task, we need to know more specifically, how the image velocities depend on the information we want to compute. This next slide shows the equations. We have the same parameters of movement and depth, and Vx and Vy on the left refer to the horizontal and vertical components of the 2D velocity at each location (x,y) in the image. The nice thing here is that there's two separate parts, one of which depends only on the translation parameters and the depths, and the other depends only on the rotation parameters. The translational component of motion in the red box here is really the most important part, because knowing your direction and speed of movement, and knowing where object surfaces are in space is so critical for tasks like navigating through the environment.

[16:35] [slide 8] So we'll finish this video by looking at the translational component of motion in more detail. First, I wrote it out here in a form that draws attention to the fact that the depths depend on location in the image, so consequently, the velocities also depend on image location (x,y). With depth in the denominator here, what happens as depth increases? As Z increases, the values of Vx and Vy get smaller. Remember that diagram at the beginning of the person in a moving train, seeing closer objects moving by very fast and distant mountains moving very slowly - the equations tell us why. A couple more important points - the translation parameters Tx, Ty, and Tz all appear as ratios, over depth - we have Tx over Z, Tz over Z, and Ty over Z. What this means is that you can never compute your actual speed of motion and the actual depths of surfaces, from image motion - you can only compute these ratios. You could be moving twice as fast through a scene where objects are twice as far away, and the motion on your eye will be the same. And finally you can ask, where does the actual focus of expansion appear in the image, given the observer's translation? Well, we know that at the FOE, there's no motion, so in these expressions here, what we can ask is, where are the numerators yielding a value of zero? Setting the numerators to zero tells us that the coordinates of the FOE are the

ratios Tx/Tz and Ty/Tz. If we're just moving straight ahead, for example Tx and Ty will be zero, and the focus of expansion will be right in the middle of the image, where x and y are zero.

So you now have the background that's useful for exploring a method for solving the observer motion problem, and you'll that see in the next video.