[SQUEAKING]

[RUSTLING]

[CLICKING]

BERTHOLD HORN: Have another go at this demonstration using a direct HDMI connection. And let's see what happens. Can't play graph. OK, so let's first go back up here. Just make--

So that's what we've seen before. And now let's try-- OK, that's a bit bitter. So this is a webcam looking down at the keyboard. And when I hold it still, you can see A, B, and C are near zero. They're small on the right side there.

And the time to contact, bottom right, is some large number that sometimes negative, sometimes positive. And now if I move it away from the keyboard, it should go green on C. The third one is the C component-- 1 over time to contact. As I approach the keyboard, it should go red, meaning danger.

So there's that. And, of course, it's independent of the texture. So I can do the same thing on any surface. If I try and move it in x only, then the first bar would be large. And that's sort of true.

I haven't got it oriented exactly right. But then if I move it in this direction, the second bar-- that's B. OK, so that's working a little bit better than it was last time.

Start off with a correction. Somebody last time pointed out a sign era. So we had that discussion about different ways of thinking about time to contact, one of which was what we were discussing and another one was rate of change of size.

And from perspective projection, we had this equation. And then we cross multiplied. And we got that. And-- sorry, equals. And then we took the derivative to get-- because this is a product and this is a constant. So we get 0.

Then-- and so there's a minus sign there. And that makes sense because when the dzdt is positive, things are moving away. And in that case, the image of those things is shrinking-- negative dsdt. All right.

We are busy talking about perspective projection and how we can use it in various ways. And in particular, we were busy talking about vanishing points. So, again, to explain what those are-- so here's our imaging system. There's an image plane. And there's a center of projection.

And out in the world, we have a bundle of parallel rays-- vectors that are filling space. They're all parallel. And one of them is special because it goes through the center of projection.

So out of all of these parallel rays-- and, of course, in much of human construction, we have parallel lines. Happens to be a fairly efficient way of doing things. And unless you're trying to be very artistic and build this data center, you'll have a lot of parallel lines.

So these could be edges of buildings, corners, edges of windows. Anyway, one of them is special in that it goes through the center of projection. And it hits the image plane at a particular point.

And we can use that point as representative. That's a way we can talk about which set of parallel lines we're talking about. And that's called the vanishing point. And the reason is that, first of all, for this particular line, you're looking straight at that line. And you just see a point.

And what about the other lines? Well, if you go far enough out on any of these parallel lines, their projection into the image will come closer and closer to the projection of this line. Why? Well, because there's a decrease in magnification from distance in the world to distance in the image.

The further way we go, we have-- f over z. And so if Z becomes very large, then the magnification becomes very small. And so any difference between these is reflected in the image by a smaller and smaller distance.

So these other rays actually are also image, these other lines-- as lines. And those lines have to go through the vanishing point. So as I move outward, I guess I have the arrows reversed in the two cases. But as I go outward along these lines, I am coming closer and closer in the image to that vanishing point.

So that's the idea of a vanishing point. And we can exploit that because it allows us to determine relationships between coordinate systems. And it allows us to calibrate the camera. So those are two things we'll briefly discuss today. And we started yesterday.

So, first, let's talk about the camera calibration problem. So here's my image plane. Here's the center of projection. And now suppose that we live in a world of rectangular objects.

And so each rectangular object has sets of parallel lines-- three sets of parallel lines-- I guess four of each of them. And so they define a coordinate system. They define three directions.

And I can pick parallel lines that go through the center of projection. So I-- OK, so out here is some object. And it has parallel lines, like that. And I pick, out of the family of parallel lines in that group, one that goes through the center of prediction. And so let's call that direction x.

So that's a coordinate system, which is just parallel translation of the coordinate system on the object. Those are the-- so for the moment, we're ignoring translation. We just were going to worry about orientation.

So then I project those three into the image plane, as we did over there. And, well, in my case, some of them-- some of them may go out of the format. And often, the vanishing points are not in the part of the image that you're actually sensing.

But that doesn't matter. We're only interested in their position in the image plane. So let's suppose that-- I don't know, we can call these a, b, and c. Or we can call them r2 and r3. So if I have my picture of the cube, as I did last time, I can define three vanishing points just by extending those parallel lines.

Now, how accurately do we know them? Well, that's another story. We'll need to know how accurately we can determine lines.

But let's suppose for the moment that we've got these three vanishing points. So we can imagine the little diagram in here of some rectangular object and highly distorted because, actually, in practice, those vanishing points will tend to be further out. If I draw them this closed in, I'm going to get a lot of, quote unquote, "perspective distortion." Of course, the term distortion is sort of odd because basically this is what perspective projection does.

It's not like the effect of radial distortion in an image, which is an undesirable property that warps the image plane. OK, what do we do with that? Well, one thing we can do with that is try and figure out where the center of projection is.

So that's where we were going last time. So we have a coordinate system that's in the imaging plane, in the image device. And we're trying to find the relationship between the coordinate system in the object and the coordinate system in the image plane.

And then we're trying to find out where the center of projection is. So a couple of terms-- so one thing is the point that is perpendicular below the center of projection. So we draw the perpendicular from the center of projection down here. And that's called the principle point.

And as we indicated, you'd like that to be at the center of the array. But it won't be very accurately. And to do accurate work, you need to know exactly where it is. So that's two numbers-- row and column in the image plane.

And then the third number is the height of the center of projection above. And we call that f. And that is to remind us that in the lens system, that's the focal length. It's typically slightly larger than the focal length.

Anyway, so there are three degrees of freedom. They're three numbers we need. And one way to think about it compactly is we're trying to find out where that point is.

And that could be a difficult task to do by physically disassembling the camera. For a start, in a cell phone camera, those distances are very small. So you'd have to measure them very accurately. And it probably won't be the same after you reassemble the camera.

So how do we do this? Well, if we connect the central projection to the vanishing points in the image plane, we have three vectors, which are basically these three vectors up here. Well, try and draw it that way maybe.

And so therefore, they're at right angles to each other. And where does that come from? Well, our assumption is it's a rectangular object. If it were some other object and we knew what the angles were, we could use that as well. It would be slightly less convenient.

But so we now know that we're looking for a point up here such that if you stand there and you look down into the image plane, the directions to these vanishing points will be right angles to each other. So that's the task. We move around in this space to find that place.

And so let's start in 2D. We already mentioned this last time, that the angle made by the diameter of a circle from points on its circumference is the right angle. So conversely, the locus of all the places you could be from which those will be at right angles to each other is a circle. This diagram will come up later when we're talking about photogrammetry and ambiguity in imaging of surface terrain because if you have two landmarks and they appear at right angles to each other, you might think that that would tell you where you are in the airplane.

But actually, no, because you could be anywhere on this circle and you would see them at right angles to each other. OK, so that's the 2D version. And the 3D version, of course, is you just spin this around its axis. And you get a sphere.

So that constraint on the position of the center of projection is just that it lies on a sphere. So what sphere? Well, for a start, it lies on the sphere where R1 is one end of the diameter and R2 is the other end of the diameter.

So I can imagine that we draw a sphere with r1 and r2 as the diameter. And so it goes above and below the image plane. And the center of projection must lie on that. Now, of course, that's not enough to tell us where it is. So we have a second sphere.

We connect, say, r2 to r3. And that's the diameter of a second sphere. And we intersect those two spheres. So what's the intersection of two spheres? It's a ring.

So now we have a ring. That means we haven't really solved the problem quite because we still have an infinite number of possibilities. So we use a third.

We use a sphere with diameter there r3 and r1. And now we intersect those. And we're left with how many solutions?

Two. Right. OK, and so there's a two remaining two way ambiguity. And in this simple case, it's simply that when we're up here, we get the same right angle condition than if we're mirror image below the image plane. Well, in our case, we know that there's a physical constraint, which is that the center of projection has to be above the image sensor so that its imaged so that the second solution can be eliminated.

OK, so want to just talk a little bit about some of this is all very simple and basic. But it's good to remind ourselves of these things. So we'll start off talking about linear equations.

And whenever we can, we're trying to reduce things to linear equations because we know how to solve those. And so geometrically, what are they? Well, they're straight lines.

And what equations do that correspond to? Well, there's this old chestnut, which has some real problems. But that's one way of writing the equation for a straight line.

Then we can just say, well, it's a linear equation like that. And in this case-- so here we've got two parameters, m and c. And what is that?

That tells us that the family of straight lines is a two parameter family. If we look at this, there are three parameters. How can that be? Well, because there's a scaling we can do.

So true, they're three numbers. But actually, if you divide through by any one of them, you get the same line, and you're down to 2 degrees of freedom. So it's a 2 degree of freedom world. And then we can go-- probably get the signs wrong but-- make sure I get them the right way around.

Sine theta x minus. OK, so what's that? Well, that's another way of parameterizing a straight line, which we'll use quite a bit. And you can check that for sine theta equals 0, this is that y equals rho. So theta equals 0 is that line.

And for theta equals pi over 2, we get x is minus rho. So that's this line-- and so on. And two parameters-- so here we got the world of lines is parameterized through theta and rho instead of over here, where it's parameterized in terms of m and c. And why is that useful?

Well, the thing up there is that if your line happens to be parallel to the y-axis, then m is infinite. So there's a singularity, which we avoid in this representation. Anyway, straight lines in 2D-- the linear equation's in x and y.

But now we're dealing with 3D. So let's talk about that. And so now we're talking about planes. And, of course, planes also represented by linear equations, except now in three unknowns.

And one reason I introduced this notation is because we can generalize that to 3D. OK, so we can write the equation of the plane as that. So it's a linear-- I mean, if I expand out this dot product here, I get linear equation in x, y, and z.

And it's just convenient to write it that way. But if I wanted to, I can write that a, b, c, d. So that makes it look like they're 4 degrees of freedom. But, of course, they aren't because of the same scale thing. If I multiply a, b, c, and d by 2, I have the same plane. So, actually, it's three degrees of freedom.

And I can also see that from this representation because n is a vector-- so three numbers. But it's a unit vector. So there's one constraint. So there are only two degrees of freedom. And then I have rho.

So 2 plus 1 is 3. So the family of planes in the 3D world are three dimensional, which is an interesting duality because it means that there's a mapping between planes and points in 3D, just as there's a mapping between lines and points in a 2D case. So I can either plot theta rho or mc.

Oh, and by the way, if I like that representation, then you can now see the similarity where in the 2D case, this vector here has an x component that increases as x-- oh, it becomes negative larger as theta increases. So that's the minus sine theta term. And the y component is large when theta is 0 and then get smaller. And that's the cosine component.

So you'll see that this equation is the same as n dot r is rho or something. And, of course, that's the same equation we have over there, just in 3D. OK, so that's all pretty obvious. Now, back to our camera calibration problem.

So one way to approach this is to think of this as the intersection of three spheres. And so let's talk about that a little bit. So I mentioned last time this problem of multilateration, which comes up in robotics, where, for example, we have the distances to a number of Wi-Fi access points. And let's say we have three.

And you'd think that should allow you to compute where you are. And it does. So let's solve that problem first.

So we're trying to intersect three spheres. So how can we talk about a sphere? So here's a sphere. The i-th sphere. We'll have three of them. So rather than write three equations, I'll write that.

And it's just that the magnitude of that vector difference is the radius of that sphere. And so we'll get three equations like this and try and combine them. Now, if I want to write this out, I can just write this because the definition of that magnitude is just the square root of the dot product of those two vectors. OK, then I can multiply this out to get that equation.

And so for every sphere, I'm going to get a second order equation like this. And we mentioned last time that, therefore, by Bezout's theorem, we may have as many as eight solutions unless the equations have special structure. Well, we can exploit that by considering a second sphere-- the same equation, just i instead of i.

And by subtracting them, we can get rid of that annoying second order term. And so we end up with a linear equation. And that's always preferable. So we get 2r dot rj minus ri.

Then I need to say what that is. So ri squared is-- OK, so these things on the right-hand side are just constants. They're the distances are measured and the distance from the origin of the two reference points of the centers of those spheres.

And what's important is that on the left-hand side, I've got r dot something. Well, that's a linear equation in the components of r. And any time I can reduce quadratics to linears, I'm happy.

So OK, now we've gotten it for one pair of spheres. But we actually have more. So we can repeat this exercise for the other combinations. So when I put it all together-- OK, and then I have all of these constant terms, right? Because the transpose says I've taken a column vector and turned it into a row vector.

So the first row of the matrix is this difference, r2 minus r1. And multiplying this matrix by this vector is taking the dot product of this difference and that vector. So that just corresponds to what I've got over here. Well, there's a factor of 2. I forgot a factor of 2. Do that.

OK, and then similarly, the second row, I've taken a column vector, turned it into a row vector. And so the second term in the result of multiplying this matrix by this vector is to take the dot product of this vector with that vector. So that's the same equation just with i and j changed. And I do that a third time. And I get that.

And hurrah, I got three linear equations and three unknowns. What could be better? OK, there are some people shaking their heads. So why is that wonderful if it's true?

Well, because I know how to solve linear equations. And there's only one answer-- et cetera, et cetera. But wait, we said there were two answers. So something's wrong.

So what's wrong with this? Why will this fail? So when do linear equations not have a unique solution? When there's redundancy, when the rows in the matrix are not independent, when the matrix is singular when the determinant is 0-- all different ways of saying the same thing.

Now, how can I be sure that's the problem? Well, if I just add up these three rows, what do I get? I got zero. So the three rows are indeed not linearly independent.

The third row doesn't tell me anything new because I could have got it from the first two just by-- if I add or subtract. If I take this one and subtract that one, I get this one. And the same with the right-hand side. So it's all consistent.

So, yes, this statement is true but it's not giving me a solution because this is a singular matrix. So we always have to check that not only do we have enough equations and unknowns, but that we can actually solve the problem. Yeah?

Right. OK, so his statement was that we think there should be two solutions. But if the matrix is singular, there's actually an infinite number of solutions I can construct along a line, any number of solutions I like. So what happened? What's going on?

Well, what we did was we manipulated the equations. And we got some more equations. But we threw away the original equations. And that may or may not be legitimate.

In this case, we've lost-- something that satisfies these three equations does not necessarily satisfy these three second order equations. So satisfying this equation doesn't guarantee that we actually have a solution. So this is another thing, another cautionary tale.

And you will see this sometimes in papers. Great, we manipulate equations. We get some equations we can solve. And then, oh, actually, we're not only getting the solutions we're supposed to get. But we're getting other stuff.

And so in this case, it's perfectly legitimate to derive these equations. But you can't then throw away the original equations. And in particular, in this case, we can use two of them. But we need to keep one of the quadratic ones.

So we can keep, for example, the first two equations over there and keep the third one of these. And that's perfectly legitimate. Now, we got two linear equations, one quadratic, and be Bezout's theorem, we would get 1 times 1 times 2 solution maximum, i.e. 2. And that fits with what we're expecting.

OK, so let's deal with this in another way. So what we've got is constraints like this. And we can write them out the way we did over there and then subtract them. And what we're going to end up with is-- and we can derive two more equations like this. But let's just stop and look at this for a second. What was this? R2.

OK, so all I've done here is I've subtracted these two and reorganized the terms a little bit. So what does this tell me? Well, that product being equal to 0 means that the two vectors are perpendicular.

So that's one important thing, that r minus r2 is perpendicular to r3 minus r1. The other thing we can notice is that the plane goes through r2. So first of all, this is a linear equation.

So by our discussion over there, it represents a plane in 3D and passes through r2 because if r equals r2, this is 0. And so that satisfies this equation. OK, so we got two important properties. And now if we go back to our vanishing points in the image plane-- I'll draw it again so-- r1.

So what this is saying is that this particular equation is a plane that is perpendicular to r3 minus r1-- so r3 minus r1. That's this vector. So this plane has this as a normal perpendicular. And so for a start, it tells you that that plane is perpendicular to the image plane.

So the solutions are on that plane that's perpendicular to the image plane. But which of these planes is it? Because just saying that this is the normal perpendicular to the plane, we could have lots and lots of planes. Well, that's the second statement. It passes through r2.

So we're popping back and forth between algebra and geometry because you can do all this just algebraically. But you don't really get much insight. And it's much more fun to look at it geometrically.

Now, of course, I picked this particular combination. I could pick two other combinations. So what do I get from those? Well, one of them is going to give me a plane that is perpendicular to r3 minus r2 and passes through r1. So that's that one.

And then there's a third one, which is going to be a plane perpendicular to r1 minus r2 and passing through r3. And so ta-da, I got a solution. So what is that called, by the way? Triangulation. OK.

Well, so with triangles, there are a lot of special points that have names. And I don't know. There are at least six. I'm sure people have come up with other ones.

For example, there's the outcenter, which is the center of a circumscribed circle. Then there's the incenter, which is the center of a inscribed circle. Then there's the centroid, which is the average of the three sets of coordinates. And this one is the orthocenter.

I don't know. There are a few more intersections of bisectors and whatever. Anyway, this is the one we want. And it's obtainable easily by solving linear equations.

So we're partly done. We're not quite done, because now we know where it is in the image plane. So this is the principal point, the one we were talking about that the perpendicular we dropped from the center of projection into the image plane. That's where that is.

What are we still missing? Well, we're missing f. So we know that the solution is along a line that's coming perpendicular out of the image plane. Why? Well, because we're intersecting these planes that are each individually perpendicular to the image plane. So they'll produce a line.

And by the way, of course, I don't need all three planes. I just need to intersect two. Yeah, OK. Let's start with r.

So the unknown center of projection is out here. And this is the perpendicular I dropped from r into the image plane. And the equations I'm solving are these equations. And the thing I'm exploiting is that all of these have a 0, a z component.

Why is that? Well, because in terms of the coordinate system I'm using, which is row and column in the image, the height is 0 in the image plane. The center of projection is out here at some non-zero z. But these points are actually in the image plane, which has height 0 relative to the image plane.

So none of these vectors have a third component. And so all of these equations here, I can think of, really, as equations in two vectors, not just x and y. And I only need two of those to solve for x and y.

But that's just algebra that represents this geometric insight. So one of the things that-- what do you do with this? Well, camera calibration.

But you can also do some fun things with it otherwise. So, for example, if you take an image and you find this point, and you find that it's not in the middle of the image, like here's your picture and you do this construction and you find the vanishing points are, I don't know, outside the image. And then you discover that, oh, the center of projection is, I don't know, here-- the principal point.

Well, then whoever took this picture either had a very funny camera or they cropped you out of the picture. Very commonly done when relationships break up. You take this great picture where you look really good. Then you cut out the other person.

And so that's one of the kind of lighthearted-- or maybe not so lighthearted, in that case-- uses for this technology. Another one is to try and question whether an image is an original or has been modified. And this came up in did Admiral Peary get to the North Pole or not?

And his proof was partly in the form of photographs. And what you can do-- and we know what camera he used. So we know the focal length, et cetera. And, well, we also know what altitude the sun should be at that time of year.

So you can do some photogrammetry and discover that, well, for example, one of the important pictures has been cropped. So this is the kind of technology that will let you do that. And I guess the Photogrammetric Association, which is into this sort of stuff, picked that up. And they published a book that kind of questions his claim to have reached the North Pole.

I'm interested in that partly because there are wonderful hoaxes in exploration, a lot of which are easy to understand. Someone spent years raising money, years finding people to work with them. They get within 150 miles of the North Pole and there's no one around.

Are they going to come back and say, no, I didn't make it? Well, if they're really, really honest, that's what they say. But if they sent back the only other guy that knew how to operate a sextant to measure the altitude of the sun, they might just be tempted to say, yeah, I got there.

Anyway, so doing this kind of vanishing point analysis can sometimes alert you to problems with image manipulation. But we're using it for calibration instead. So we can write down the two linear equations for that point. But it's not-- I mean, you know how to solve two linear equations.

So we still have to find f. But that's OK because we now know x and y. And all we need to know is the third component of that vector. And we end up with a quadratic. And we know how to solve a quadratic. So I won't do that.

But here's another interesting thing. I kept on saying that in typical cases, the vanishing points will be outside the frame. And so one thing we might want to do is, given this positioning of the vanishing point, can we say something very quickly about the f, the focal length? Well, here's a-- let's take a really simple case.

So here, the vanishing points-- and they happen to be equally spaced in the image plane. We just turned the cube so that we're looking at the three faces equally. And let's suppose that this distance is, I don't know, v for vanishing point. And then the question is, what's if?

The general approach to this involves plugging in the x and y we get from this and then solving a quadratic in f. But maybe we can do this without all of that because it's a special case. We should be able to figure this out.

So we can think of this in terms of corner of a coordinate system. OK, so we have 1, 0, 0; 0, 1, 0; and 0, 0, 1. And we need to know the distance from the origin to this plane. And, of course, it'll vary depending on where we are.

But somewhere out here is the point that is closest to the origin. And it's the one we indicated by distance rho from the origin. So the question is, how far is that point from the origin?

Well, presumably it's a point where-- what's another variable name? A, a, a. It's symmetric. So you would imagine that at that point should have the same x, y, and z coordinates.

And so the dot product of that with the unit normal to this plane should be 1. So the unit normal to-- the perpendicular to this plane comes symmetrically straight out equally in x, y, and z. And to make it a unit vector, we have to divide by the square root of 3, right? Because we've got 1, 1, 1.

And so I think this is that A is 1 over square root of 3 because it's 3 times-- yeah, OK. So this is also what we called rho before. OK, so that's the distance from the origin to this point.

And that's going to be our f. But what is v? Well, in this diagram, this is v.

So in this case, v is square root of 2. And f is 1 over square root of 4. So there's a relationship which is that v is square root of 6 times f, or f is v over square root of 6.

So in this special case, we can easily calculate what the focal length or the principle distance is. And it's substantially larger than-- sorry, I got this the wrong way around. F should be-- no, it is right.

And so v will typically be substantially larger than f, the principle distance. And so often the vanishing points will be outside the frame that we've actually captured an image of. And this is a special case.

We can solve it in a general case. It's just algebra. So that's application of vanishing points to camera calibration.

Now I want to talk about another application. I mentioned the case where we just slap a cell phone camera onto a car's window. And we want to relate the images we're seeing to some three-dimensional world coordinate system just by identifying features in the image.

So what sorts of things can we see in the image? Well, if it's on a straight road, we'll see the curb, and we'll see road markings. And those are supposedly parallel.

And so they will produce a vanishing point that we can detect in the image. If we're lucky, there's also a horizon. So we get a second constraint out of that.

And what we want to know is how is this camera oriented relative to the road and relative to gravity. So that's the kind of problem we're trying to solve. And I guess we got rid of that also.

So this is really about orientation. So the transformation between a world coordinated system that's lined up with the road and gravity to the camera coordinate system is translation-- there's some shift-- and rotation. And we're going to focus here on just recovering the rotation.

So where do we start? Well, the same diagram. Let's suppose we're lucky and we actually have all three vanishing points. In the application I mentioned, we don't. But let's take the easy case, where we have all three vanishing points.

So we got v1, v2, v3, r1, r2, r3. And now that we have a calibrated camera, we just connect those up to the center of projection. We now know where the center of projection is. That is we've got the three numbers. We've got two for the principal point and one for the principal distance-- or if you like, just the coordinates of the center of projection.

I don't know. Let's call it p or something. We've called it r over here. OK, then we know that the edges of this rectangular object that we're looking at to get the vanishing points have directions that are just defined by these lines. That is-- let's call this the x-axis. And then there's another one, which will be the y-axis.

And here we go. So they look a little bit funny because of the way I picked the vanishing points. But they're supposed to be at right angles to each other.

And, of course, the first thing I can do is check. I've got some algorithm that finds the vanishing points. I have calibrated the camera, supposedly. I connect these up. And I take the dot products. And they better be small. They're unlikely to be exactly 0.

So that's the first thing, to check that they are, in fact, at right angles to each other. And then I have now-- what do I have? I have the unit vectors in the object coordinate system measured in the camera coordinate system. So my definition of x, y, and z is still in this coordinate system over here. So I'll do this-- T minus r1.

And, of course, once I know that they're parallel and they're unit vectors, I can just compute them by normalizing-- et cetera. So, important to understand that those x, y, and z unit vectors are in the camera coordinate system. OK, so now suppose I have some point in the object.

And let's see, did I call these primes? And it's going to be-- yeah, they're vectors. So that's in my camera coordinate system.

And what is that vector in the original object coordinate system? So everything now is measured in the camera coordinate system. Oh, I shouldn't be-- OK.

So it's alpha in the direction of the x-axis and beta in the direction of the y-axis and gamma in the direction of the z-axis. So what is the vector in the object coordinate system? Where are its components?

Its x component is alpha and y component is beta. So this vector here that I've written in the camera coordinate system corresponds to this vector in the object coordinate system. So, I mean, that's the definition.

We have three axes in that world. And we express the position of a point in terms of a weighted sum of those three directions. And so in my camera coordinate system, this is what it looks like. In the object-- the rectangular block coordinate system, it looks like that.

So what's the transformation? So I got-- let's see. R is-- so let's see. The T, again, means I'm transposing a column vector into a row vector. So the first row of that matrix is the unit vector in the x direction laid out as a row.

And so the first component of r over here is the dot product of this x unit vector and r prime-- this guy. And well, that's what you see in that equation. The second component is the product of y with this guy, and so on.

So that's my transformation vector between the two coordinate systems. We'll be doing more of this. So don't panic.

So that's a very important matrix. And it represents the orientation of one coordinate system relative to the other. And I claim this matrix is also normal. What does that mean?

That means that the rows are perpendicular to each other. If you take the dot product of the two rows, you'll get 0. Take the dot product of those two rows, you'll get 0. And by construction, that's true because we made a unit vector. We're assuming that they represent the axis of the coordinate system, so they're perpendicular.

And then it's also normal that each row has magnitude 1. And we did that by construction because we constructed a unit vector. So amongst other things, we're now led to understand that rotation is represented by also normal matrix and quite a few photogrammetric tasks involving finding that matrix.

And in this case, we were able to do it rather straightforwardly because we explicitly could determine the direction of the coordinate axis in the object written out in terms of the coordinates of the camera. And, of course, if we want to, we can invert this. So if we've got-- maybe we need to go the other direction. Maybe we know the coordinate r and we want to find the coordinate our prime.

Well, and it turns out that for rotation matrices, we can show that they're actually just the transpose. So that follows from the property of orthonomality. So in this case, going back and forth between the two coordinated systems is particularly easy.

Onto something else, finally. So we spent quite a bit of time talking about where perspective projection and all of the stuff that's connected with it, including the derivatives, which gave us the motion field. And then we talked a little bit about vanishing points and exploiting them for camera calibration. And let's go back now to the other part of the puzzle, which is brightness and what we can do with it. How can we exploit measurements of brightness?

We already talked about foreshortening. So here's a small facet of the surface of an object. And we can specify its orientation by talking about the unit normal. And then he is an observer.

And let's call this direction the viewing direction. And there's also illumination. In the case there's just one source of illumination, we'll call that s for source. And, well, we can draw some angles here.

So this is theta I for incident angle and this is theta E for emitted angle. And, well, that's not enough. We need we need another angle to fully describe that situation, because if we just specify the incident and the emergent angle, we can spin one of these around.

So we need some azimuth angle. And we'll talk about this later. But for the moment, just imagine we take this ray from the light source and we projected down into the plane, and we take this ray to the view end, we project it down into the plane. And then we measure this azimuth angle.

Now, the observed brightness is going to depend on those parameters. It's going to depend on the material of the surface, one. And it's going to depend on the light source. And it's going to depend on those angles.

An extreme case is a mirror, where there's only one direction where you see anything. The reflected ray goes off in a particular direction. And unless your camera or your eye happens to be there, it's dark.

But the more interesting cases, where we have a piece of yellow note paper. And pretty much any direction I look at it from, it has the same brightness. And we'll talk about that more later. We're going to greatly simplify this story right away by talking about the illumination.

And we've talked about foreshortening. And so we know that if we have a patch of a certain area, it's going to get less and less power from the light source the more it's tilted relative to the light source direction. And the foreshortening makes the apparent area be the true area times the cosine of that angle.

And so just in terms of power getting in, that's the magic number. Well, if cosine theta i is less than 0, that's not true because that would mean you get negative power. So what does cosine theta i being negative mean? That means it's greater than-- the angle is greater than pi over 2.

And so that means that you've turned the surface to face away from the sun. So we should really be saying max of cosine theta and 0. But that's so tedious, we're just going to implicitly say that.

And then the surface may or may not reflect the light. It may absorb some of it. And it may reflect differently in different directions. But let's take a really simple case. Let's start off with that-- not to say that we're going to be stuck with that.

OK, so this is a model of a matte surface that's very unmirror-like. It reflects light in various directions. And it has the special property that no matter what direction you look at it, it has the same brightness. And we'll talk about exactly how we measure brightness.

So for the moment, all we're going to make use of is let's imagine there's a surface where the brightness only depends on how much power is going in. And therefore, it depends on cosine theta i. And so how can we exploit that?

Well, we make a brightness measurement. And it's going to be proportional to that. Let's not worry about the proportionality factor for the moment.

So the brightness is proportional to that dot product in this diagram between the normal and the-- right? Because n dot s is cosine theta i. And so does that allow me to determine the surface orientation?

So can I solve this for-- see, where I'm going with this is I'd like to recover the shape of the surface. And one way I can do that is to look at every little facet and figure out if its surface orientation, which is little different from-- you might think that, well, we'll just construct the depth map. We'll find some way of estimating the depth at every point in the image.

But that's something we can't do from a monocular image. We'd like to be able to recover shape from monocular images as well. So we're trying to recover n. Can we do it from this? Well, this is just one constraint.

And the surface normal has 2 degrees of freedom, right? It's the unit vector. We want constraints. So 3 minus 1 is 2.

So we got two unknowns. And we only have one equation. So that doesn't work.

But let's imagine that we're in an industrial situation. We have control of light sources and so on. And we can take a second image with a second position for the light source. And you saw that in the slides that we did at the beginning.

Well, that's already looking better because now we have 2 constraints and 2 degrees of freedom. We have a match between the number of unknowns and a number of equations. But they're not linear.

And so where's the nonlinearity come from? Well, the thing is that if we're solving for n, we need to enforce the constraint that it's a unit vector. And one way of thinking about it is we're trying to solve these three equations for n. And these are nice and linear. And this one isn't. So by Bezout's theorem, there may be two solutions.

Another way to see that is suppose that you measure the brightness, and therefore you can estimate cosine theta i. And therefore you can estimate theta i. What do we know?

Well, let's suppose we know the directions of the light source. And you know this is the direction to the light source. And then I know that there's a certain angle between the surface normal and the light source.

Well, that doesn't give me the answer because it could be-- I can rotate this around s. And so I actually have a whole cone of possible directions. So this is s. And this is theta i. And my normal could be any one of these on that cone.

Now, if I have a second measurement, second light source, that gives me a different cone of directions. And if the normal has to be on both of them, well that means that I look for the intersection of those cones. And there will be two.

And, of course, if I intersect two cones, I get two lines. But I have the additional constraint that it has to be unit normal. So it's on a unit sphere. So I then take those two lines intersect and them with a unit sphere. And I get two points

OK, well there could be bad things happening. If your measurements are wrong, these two cones may not even intersect. But we'll ignore that for the moment.

And we could write out the algebra for that. It involves solving a quadratic. We know how to solve quadratics. But we can actually turn this into a linear equation problem and make it somewhat more interesting.

I mentioned that, amongst other things, one of the things that will affect how bright something is in the image plane is the reflectance of the surface. And reflectance, unfortunately, is a very fuzzy term that means something else for everyone. So I'm going to call talk about something called albedo.

And so that's a quantity that would be between 0 and 1. And it's simply telling you how reflective the surface is, how much of the energy going in comes out again versus how much is absorbed and lost. And I put albedo in quotation marks because it has a very well-defined meaning in some technical areas, such as astronomy.

It means something slightly different in the case of astronomy. It means you've got a spherical planet. And what's the ratio of power in over power out? And so that means it's some average of lots of different directions.

Here, I'm talking about just a particular orientation. But it's a very simple concept, just a piece of white paper presumably has an albedo that's close to 1. And black coal has an albedo that's, like, 0.1.

And can you have a albedo greater than 1? No, right? Otherwise you'd be violating some law of physics. But you can get super luminous surfaces by cheating.

So if you have fluorescent spray paint, for example, or if you starched the color in your white shirt, which I know you all do, then when you illuminate it with sunlight, they are brighter than bright. They shouldn't be as bright as they are. And that's because they're converting ultraviolet into visible.

So they're not violating second law of thermodynamics. They're converting energy outside the spectrum into energy in the spectrum. And in that case, the power total visible power out can be larger than the total visible power in. And so a surface with that kind of property-- if you put starch in your white shirts, they will appear brighter than a 99.99% reflective magnesium sulfate powder, which is one of the standards that people use.

But generally speaking, 0 less than rho less than 1. And so what does that do? Well, now we have a slightly different situation, where E1 is rho n dot s1. And E2 is rho n dot s2.

And so now, actually, I also have 3 degrees of freedom in terms of unknowns because I've got the unit vector-- that's 2. And I want to recover the albedo. That makes it 3.

Well, that means I can't do it with two equations. So let me add a third one. OK, so that seems to be a match. I have three degrees, three unknowns. And I have three measurements.

And now I'm going to define n to be this quantity. So I'm going to define the three vector where all three components are actually independent to the variable. And that vector will encapsulate the things that I want to know-- rho and unit vector n.

And, obviously, I can recover very easily-- if I can find this n, I can easily find rho because it's just a magnitude of n. And I can easily find the unit vector by just dividing. And the reason I do it is because this is going to be more convenient. So then I have transpose.

Very similar to what we had earlier today, except this time it's going to work. So what's going on here? Well, the first result of multiplying this matrix by this vector is the product of the first row with this column vector.

And so that's this. That's E1 and so on. So this is just a compact way of writing those three equations. And I can then write-- I can write the solution that way. And I'm done.

And from that end, I can then recover rho and the unit vector surface [INAUDIBLE]. So a number of things to discuss about that. One of them is that this is assuming that as is invertible.

I can easily construct cases where that's not the case. For example, if s3 is actually just the same as s2, then the determinant of this matrix-- two of the rows of the matrix are the same. The determinant would be 0. I can't invert it.

And it makes sense. I'm not getting new information. If s3 is the-- if my third light source position is the same as the second light source position, I measure the same brightness. So it's intuitively clear that that's not going to work.

And then you can think of other things. Suppose that s3 is half of s1 plus half of s2. That's a little different because it's saying, OK, there's one light source here. There's one light source here. And now I'm going to put one right in the middle between them. That gives me a third measurement in the image.

Well, it turns out that you can predict it from the other two because if E3 is n dot s3 and s3 is 1/2 s1 plus 1/2 of s2, you can calculate what this is. And so they can't be coplanas. So the light sources have to be spread out in your firmament. You can't put them in a plane.

And this has important implications for astronomers because the orbits of the planets and our moon are pretty much in the same plane. And therefore, as the sun orbits around the Earth, you don't get different pictures. So that's one thing.

This isn't going to work unless we pick three independent light source directions. Another thing is that we can pre-compute this. If we know where the light sources are-- say you're doing some industrial inspection, you control with the light sources are.

You just compute s, take the inverse, and you store it. And then at every pixel, you have three frames, taken three exposures. And at every pixel, you simply form this vector and multiply the pre-computed s minus 1.

And there's your answer. I means, it's incredibly simple and very little computation, very efficient. Then another thing is that it's sort of fortuitous that we need three to do this. We need three light sources to do this.

And one reason that's interesting is because cameras typically have three sets of sensors, RGB. And so we might be able to exploit that. So one thing we could do is instead of having three light sources that come on sequentially, we could use three colored light sources and then separate out from R, G, and B.

And there's a little bit more work to do because a particular color light source is going to not just excite R or G or B but some linear combination. But the important thing is we'll have three different linear combinations. And then we can use our magic matrix algebra to deal with that.

So that's a possibility. And that is in a way more convenient. And it's faster because we don't need to turn light sources on and off, although turning LEDs on and off is pretty fast. But we can just illuminate them with colored lights.

The only problem with that is if the object is colored because then the objects-- unless it's uniformly colored, if different parts of it have different colors, that's going to confuse this algorithm and make it believe that the surface orientation is something that it's not. So that's one quick look at what we're going to do in terms of measuring brightness. This is a particularly simple case. And it's a case that's, in a way, contrived because we're saying that we controlling the light sources, which has application to industrial use. In fact, if you look at some recent patents from Cognex, they've decided to use something like this.

But it doesn't really mesh with understanding biological vision systems, except in the very, very deep ocean where anglerfish have light sources that they dangle in front of them to attract prey. We don't typically find animals illuminating the environment with different colored light sources in order to figure out the shapes of their prey or something like that. But from the point of view of recovering information from monocular images in an industrial setting, this is an interesting approach.

Now, do real surfaces follow this simple rule, cosine theta i? No. So you can't use this directly. But you can build a look-up table.

So, in fact, you don't even have to model mathematically how the surface reflects light. You just calibrate it using a shape that you know. So for every-- suppose you take a sphere. For every point on the image of the sphere, you can calculate what the surface orientation is, what n is there. And then you measure brightness in three images.

And you can build a table. Of course, the table is going the wrong way. The table is going from surface orientation to E1, E2, and E3. What you really want is the inverse. Measuring E1, E2, E3, you want to know what the orientation is. But that's just numerical inversion of a table.

So that's something we can do. But we'll go in different directions next time, starting off with different projection. So we know that real cameras perform perspective projection. And we spent a lot of time dealing with that.

But in some cases, we can approximate it by a projection that's much easier to handle. Call that orthographic projection. And the condition for that is that the range in depth is very small compared to the depth itself. And in that case, you can kind of assume that the depth is constant.

And if z is constant, then f/z is constant. And there's a constant magnification. And in that case, we can use a simplification called orthographic projection, which will exploit in our efforts to reconstruct surfaces from images. Any questions? OK.