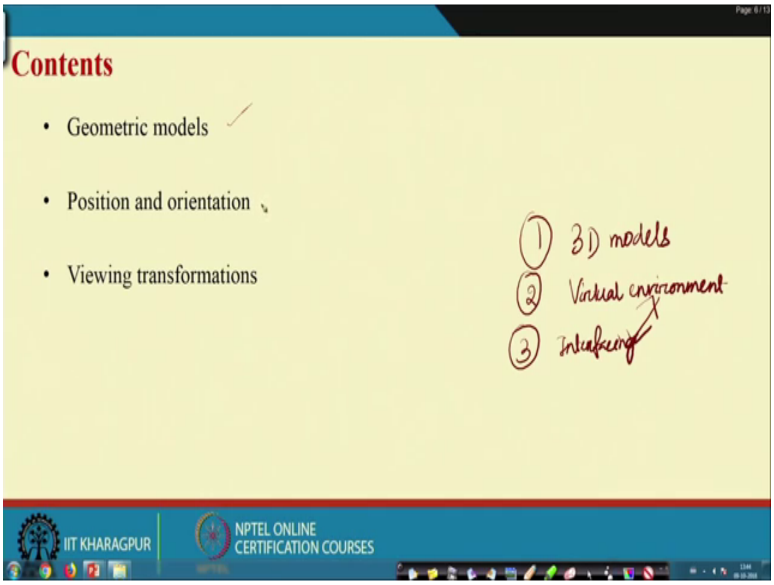
Industrial Safety Engineering
Prof. Jhareswar Maiti
Department of Industrial and Systems Engineering
Indian Institute of Technology, Kharagpur

Lecture - 58
Geometry of virtual world

So hello everyone, welcome to this lecture of Industrial and Safety Engineering on the topic of Geometry of virtual world, myself (Refer Time: 00:28) Dharmahapatra. I am the TA of this particular course, I am currently pursuing PhD under the guidance of professor J. Maiti and my thesis will be on the topic of virtual reality based accident modelling and simulation.

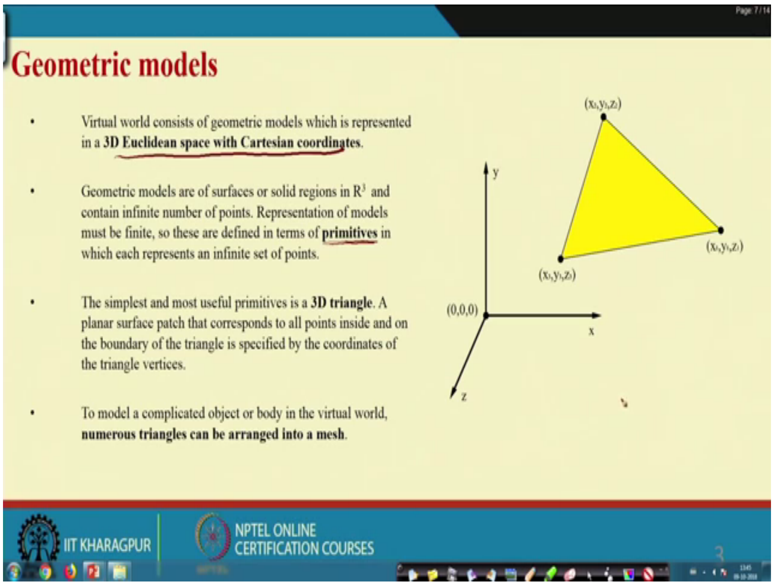
(Refer Slide Time: 00:46)



So, in this particular lecture we will cover geometric models position and orientation along with viewing transformation. So, in the previous lectures so what we observed is in the beginning; you are taught about three particular subjects that are first is first is your 3D models which is very much essential for the development of a virtual reality environment.

Second is the incorporating this 3D models into a virtual environment means development of a virtual environment and final one is the; final one is the interfacing between the user and the developed virtual environment. So, for knowing the 3D model, from knowing the three models virtual environment and interfacing, we should know how the mathematical background is taken care of in different models. So, for that we should be knowing the geometric models along with position and orientation along with the various types of transformation.

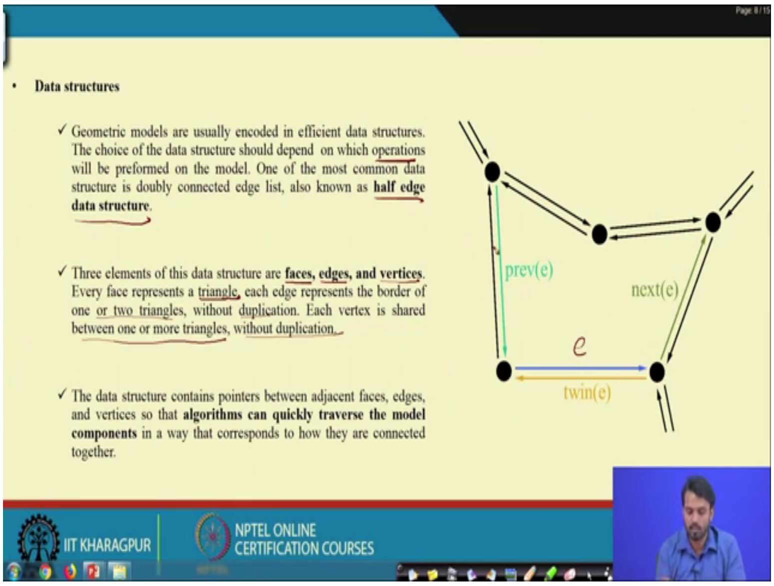
(Refer Slide Time: 02:06)



So, in this geometric models first we should know that virtual world consist of geometric models which is represented in a 3D Euclidean space with Cartesian coordinate coordinates. So, we should be focusing on this particular line that is 3D Euclidean space with Cartesian coordinates. Then the geometric models are the surfaces of solid regions in R cube real world, then which containing infinite numbers of points. This representation of models are done in terms of primitives, in which each represent infinite number of points. So, this primitive means in the small computation program which will be very much essential to develop any 3D models or any 3D modelling software that the primitives is very essential.

So, the simplest and most useful primitives is a 3D triangle a planar surface patch that corresponds to all points inside and on the boundary of the triangle is specified by the coordinates of the triangle vertices. Why we are considering triangle and not others because, if you are considering any quadrilateral form, then it will have many more computational complexity and that will consume a lot of cost and lot of time. For the simplification purpose we are considering 3D triangle is the most basic primitives used in the development of 3D models. So, to model a complicated object or body in the virtual world, numerous triangles can be arranged into a meshes. So, if we are developing a particular object or 3D model then, number of 3D triangles will be merged and meshed into a particular mesh.

(Refer Slide Time: 04:02)



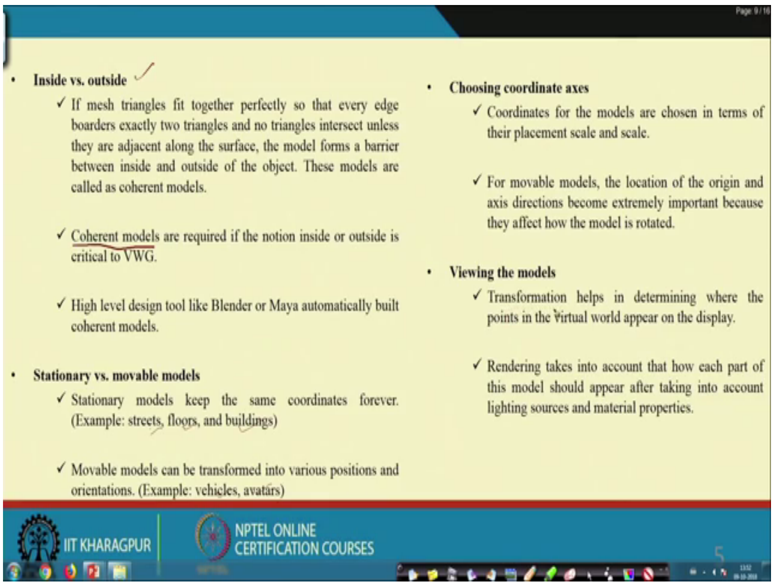
So, for the particular development of 3D model; so, what is the data structure that is needed for needed to needed to be known for the development of this 3D models. So, first is the geometric models are usually encoded in a efficient data structure; that means, the choice of data structure should be depend should depend on which operation to be performed on the model.

This operation stands for; operation stand for maybe the movement of movement of 3D models or the scaling of 3D models or the rotation of 3D models. As because we are mimicking a complete real environment so, in the real environment there are certain chances of static components and dynamic components. The static components does not have any movement, but the dynamic components are having movement in a linear axis or that can be in a rotation manner also. So, for that purpose we are creating a common data structure in which we can develop all the 3D models.

For the most common data structure which is used in the modelling of 3D models is the half edge data structure. So, in particular data structure there are three elements first is faces, then your edges and vertices. So, in this particular data structure each face represents a triangle each face represents a triangle and each edge represents the border of one or two triangles without duplication right. So, each vertex is shared by shared between one or more triangles without duplication. So, we have to know the; we have to first make a trade off between this creation of triangles in the terms of faces edges and vertices.

So, the data structure is having pointers between adjacent faces edges and vertices so that the algorithm can quickly traverse the model components in a way that corresponds to how they are connected with each other. So, if in this particular diagram you can see if this half face a structure is represented by e, then the pointer should store the value of the previous edge and the next edge along with the twin edge. The twin edge is the adjacent face of the another triangle. So, this complete 3D model will be developed with the help of this fundamentals that the previous the pointer should store the previous value of the edges, the pointer should store the next value of the edges and pointer should store the adjacent value of the edges of a other triangle, another triangle.

(Refer Slide Time: 06:35)



So, along with that we have to consider another properties of this triangle that inside versus outside. So, if the mesh triangles fit together perfectly so, that every edge borders are exactly two triangles and no triangles intersect unless they are adjacent along the surface the model forms a barrier between inside and outside of the object. So, this property of models are called as coherent properties or this model for called as coherent models. So, we have to consider the fact that there should be a barrier between inside and outside of the object otherwise there will be confusion while creating the 3D models. It will be floating in the created environment. So, 3D models will be diverged and floating in the created environment.

So, coherent models coherent models are very much necessary; coherent models are very much necessary to have a notion between inside or outside in a virtual world generator which you are taught in the previous lectures. So, high level design tool like Blender or Maya automatically build the coherent models. When we are going for development of any particular 3D models; so, now, a days the software's we are having to develop 3D models those are having this property in built. So, you do not have to know the; you do not have to go for this creation of inside and outside creation of barrier between inside and outside of the object. But we have to know the basic fundamentals between the behind the creation of this barrier of barrier between inside and outside. So, this is another property of inside and outside of the 3D models.

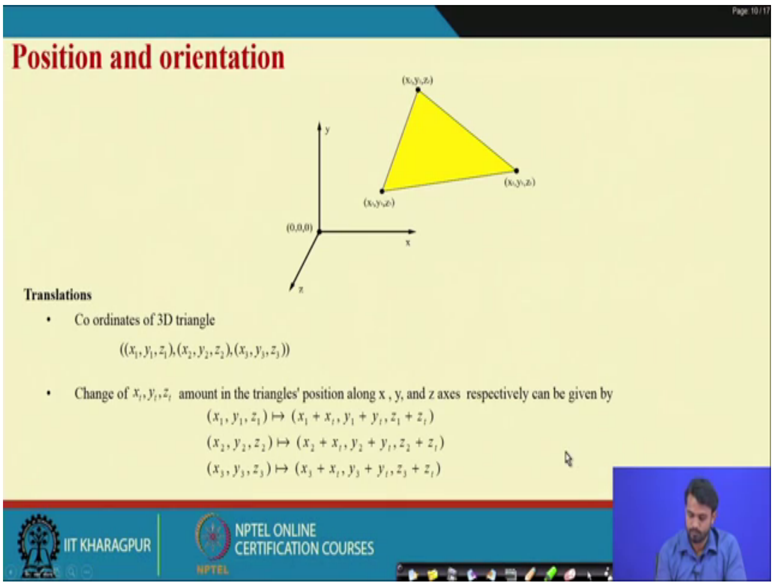
Then if you are considering stationary or movable models: so, stationary models keep the same coordinates the coordinates will not vary. It will keep the coordinates intact. Similarly if you are considering movable models, then it can be transferred into various position and orientation. So, some of the examples of stationary models are streets, floors and buildings. Similarly some of the movable models are vehicles and avatars. So, this stationary and movable models can be achieved with the help of three kinds of treatments. So, first is tracking devices if you are using tracking devices, so it can track the movement of your different 3D models.

Similarly, if you are giving any user input that user input; that means, any controller devices you are using to move for the movement of this particular 3D models, then it can be, the movability can be achieved also. Parallely there is another property of this movable models that is if you are providing physics; physics to that particular movable model. So, it will movement of that model can be taken care of with the laws of physics the rules and principle of physics. Similarly the choosing of coordinate axis; so, coordinate axis for the models are chosen in terms of their placement and placement scale and the scaling. For movable models the location of the origin and axis direction becomes extremely important because they affect how the model is rotated.

So, if you are considering the model static models then their placements and their scaling is very much important because for a replication of a real world, the static models and the movement models to look like the exact models that are present in exact components that are present in the real world so, that the user can feel he is immersed in that particular virtual environment. So, for the movable models the location of the origin and axis direction are very important because it will decide how the model is translated, how the model is rotated.

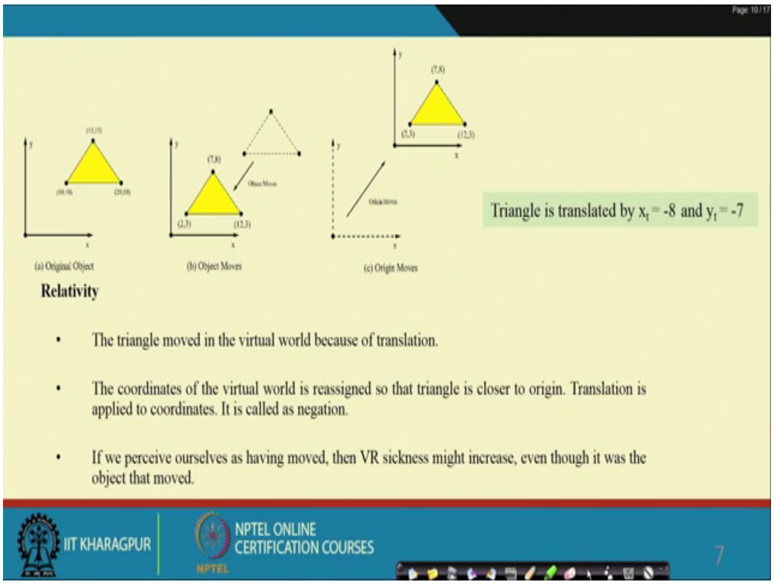
So, for the viewing of this models, so transformation properties is done so, transformation property is taken is taken care of. In the transformation property it will help in determining where the points in the virtual world are appearing on the display. That means, points inside the three models how it is appearing on the particular display. The display methods are already taught you in the previous lectures. So, the rendering takes next phase is the rendering. The rendering takes into account how each part of this model should appear after taking into account lighting sources and the material properties. So, whenever the lighting is done to a particular 3D models; so, how the rendering look like means after giving the lighting and material properties to that particular object, how it is looking like in the display medium that is the concept of this rendering.

(Refer Slide Time: 11:14)



Next we will go for this position and orientation. So, let us consider a 3D triangle where the coordinates are denoted as x 1 y 1 z 1 x 2 y 2 z 2 and x 3 y 3 z 3. So, the if the translation property will be taken care of in this particular topic in this particular model, then change of x t amount y t amount and z t amount in the triangles position will affect. How it will affect the position of x y z axis, we will look. So, the position will be changed like x 1 y 1 z 1. So, x 1 plus x t because it is translated in x t amount in the x axis, similarly y t amount in the y axis and z t amount in the z axis so, each of the coordinates will be translated in the amount of x t, y t and z t; for similarly for x 2 y 2 and z 2 this property you, we can observe.

(Refer Slide Time: 12:20)

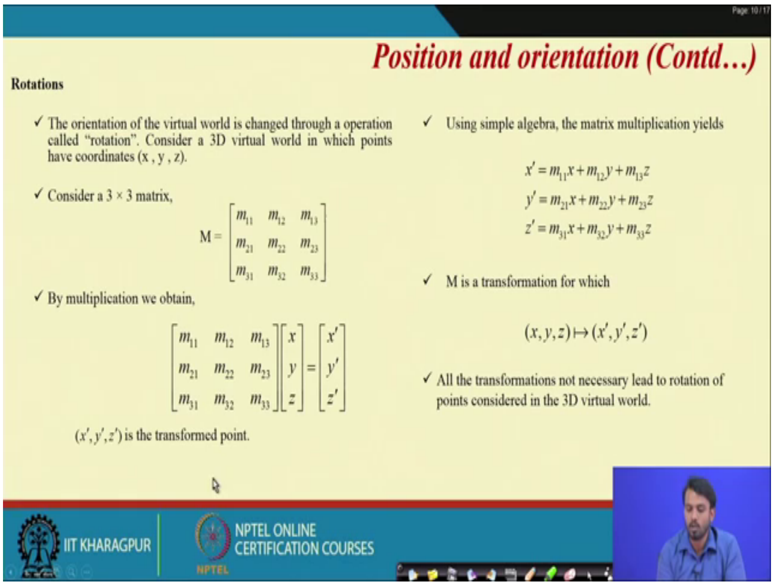


Then another concept is relativity. So, if you will consider this triangle, its original position was 10 10 20 10 and 15 15 15. We are considering x and y axis not the z axis. So, if you are moving this triangle in the virtual world because of translation how the coordinates are changing we will look. So, the coordinates of the virtual world is reassigned. So, the triangle is going closer to origin then translation is applied to the coordinates; it is called as the negation. So, if the original coordinates were 10 10, 20 10, 15 10 and the triangle is translated by the amount of x t is equal to minus 8 y t is equal to minus 7. So, giving x t is equal to minus 8, then 10 minus 8 is your 2.

Then similarly in y t axis 10 minus 7 is your 3. So, the transform coordinates in the triangle is 2 3, 12 3 and 7 8. In this case what we did is the we move the object by giving the translation property to a particular triangle. Similarly in other fashion keeping the object constant keeping the object static, we can translate the origin also to nearer to the particular triangle; that means, 3D model.

So, in this property if you are putting an object at the origin, then he will look at the; he will look at the object in a same fashion in both the case that mean you say in this case; that means, in the previous case how the object is moved, then in the second case how the origin is moved. If the user will be at the this at this position, he will feel no difference between 2 cases. But he will have the he will have the perception error; that means, if the user will move along with the move along with a axis and the object is static, then it might increase the VR sickness VR sickness even though it was it was the object that is moved right.

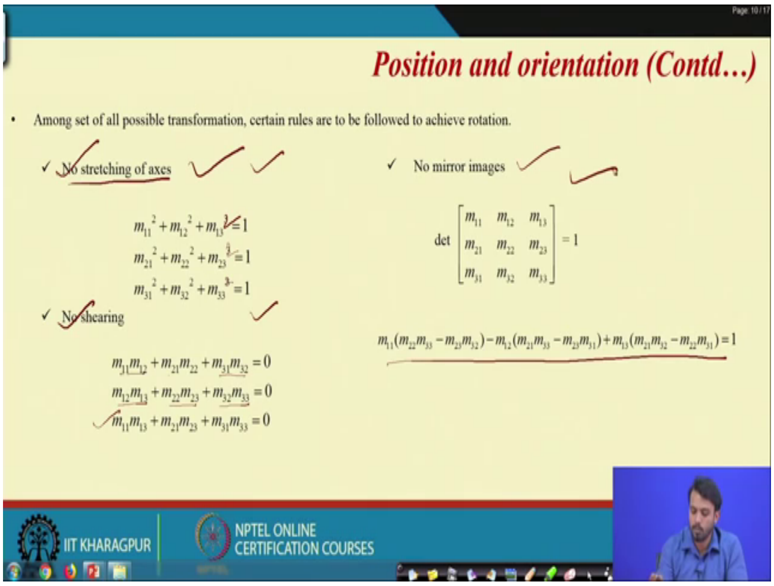
(Refer Slide Time: 14:34)



Next we will go for rotation. So, we first saw the translation and relatability, then we will go for the rotation property. So, the orientation of the virtual world is changed through a operation called rotation. So, consider a 3D virtual world in which the coordinates are x y z. So, consider a 3 into 3 matrix that is a transformation matrix m in which the different coordinates are mentioned. So, by multiplying by multiplying we obtain m 11 into x m 11 matrix then x y z matrix which is the original matrix we have considered; then this is the complete transform matrix that is x dash y dash and z dash.

So, x dash y dash and z dash is the transform point in the 3D virtual world. So, using simple algebra we can obtain that x dash is m 11 into x plus m 12 into y plus m 13 into z. Similarly we can obtain the coordinates for the y dash and z dash. So, m is the transformation for which the original coordinates of this 3D component x y z means points in a virtual world transformed to x dash y dash and z dash. So, all the transformation not necessarily lead to rotation of points considering the 3D virtual world. So, we have to consider certain properties which is to be applied to that particular components so, that the rotation can be achieved.

(Refer Slide Time: 16:10)

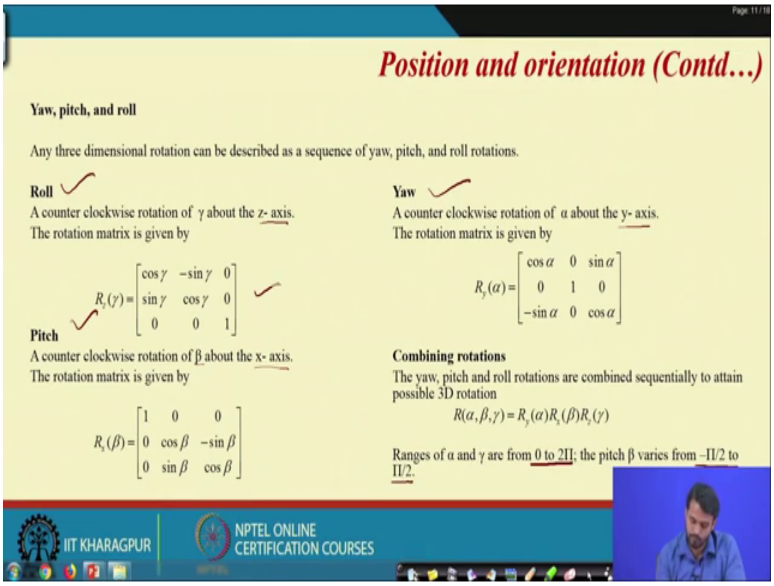


So, first is the no stretching of axis; that means, in the previous slide you saw that m 11 m 12 and m 13 represent the transformation matrix along with this m 21 m 22 and m 3 m 23. So, what for no stretching of axis, you should be having this you should be following this property that is m 11 square plus m 12 square plus m 13 cube is equal to 1, square is equal to 1. Similarly m 21 square m 22 square and m 23 square is equal to 1 and m 31 square and m 32 square and m 33 square is equal to 1. So, this property is for the no stretching of axis.

Similarly, the model should not be sheared that; that means, the model should not be distorted. So, for the shearing properties m 11 into m 12 plus m 21 into m 22 plus m 31 into m 32 is equal to 0. Similarly for m 12 into m 13 m 22 into m 23 m 23 2 into m 33 is equal to 0. Similarly we can observe for the third condition also. So, first one is the no stretching of axis, second one is the no distortion and third one is the no mirror images. So, for the mirror images; so, determinant of the transformation matrix should be 1. So, if we will consider the; if we will calculate the determinant so this will come like this.

So, m 11 into whole into m 22 into m 33 minus m 23 minus m 32 similarly for m 12 and m 13. So, these 3 properties should be maintained for the rotation of particular 3D components; that means, no stretching of axis, no shearing and no mirror images.

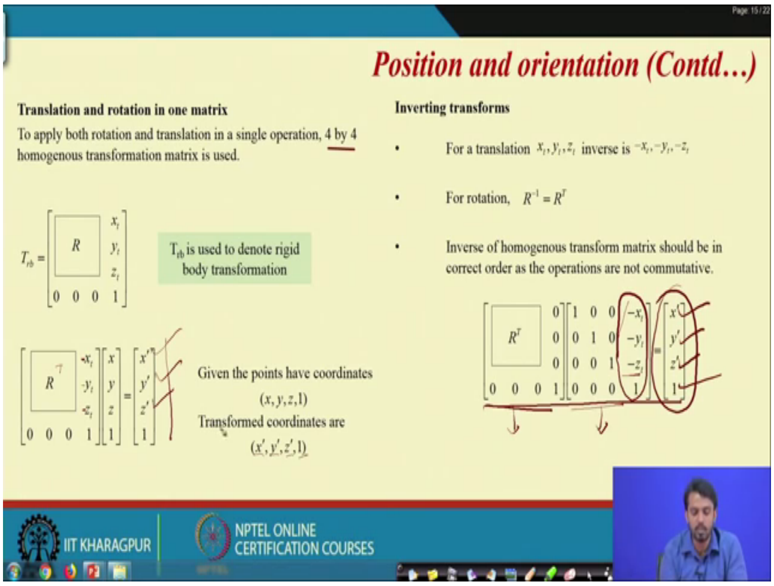
(Refer Slide Time: 18:04)



So, yaw, pitch and roll. So, any 3 Dimensional rotation can be described as a sequence of yaw, pitch and roll rotations. So, what is roll? A counter clockwise rotation of gamma, we are taking the gamma as a angle rotation angle about the z axis in which the rotation matrix can be given by r z gamma is equal to r z gamma is equal to cos gamma minus sin gamma 0 sin gamma cos gamma 0 then 0 0 1.

Similarly, for the pitch rotation, a counter clockwise rotation of angle beta about the x axis is considered in which the rotation matrix can be given by 1 0 0 0 cos beta minus sin beta, then 0 sin beta and cos beta and similarly for the yaw motion a counter clockwise rotation of angle alpha about the y axis is considered. So, the rotation matrix is given by cos alpha 0 sin alpha 0 1 0 and minus sin alpha 0 and cos alpha. So, in 3 of the different kinds of rotation; that means, roll pitch and yaw, you have to you have to consider first thing is z axis in roll pitch x axis in pitch motion and the yaw y axis in the yaw motion. So, if you are combining the rotation, then your pitch roll rotation are combined sequentially to attain a possible 3D rotation. So, R alpha, gamma and beta can be represented as R y alpha R x beta and R z gamma. So, the ranges of this alpha and gamma are 0 to 2 pi and the pitch varies from minus pi by 2 to plus pi by 2.

(Refer Slide Time: 20:07)



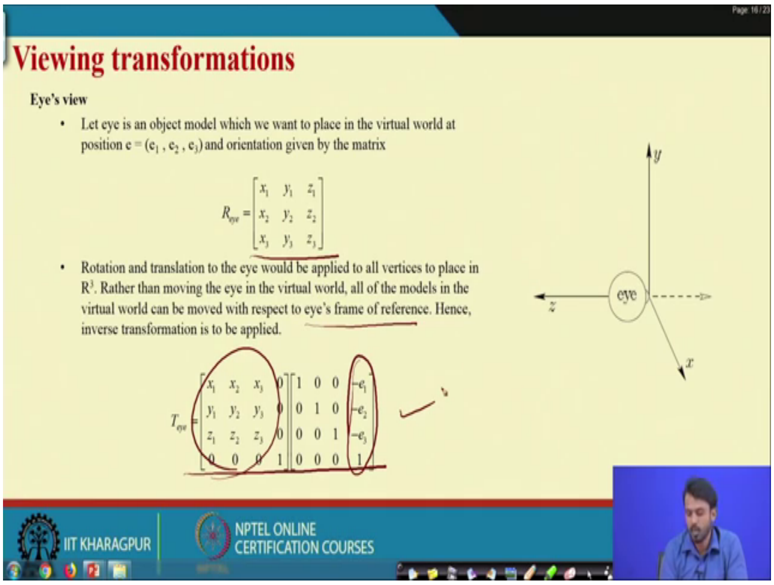
So the, if you are considering means translation and rotation in 1 matrix, to apply both rotation and translation in a single operation 4 by 4 homogeneous transformation matrix is used. Why? Because if you are transforming means with the help of 3 into 3 matrix, the operation will be not possible means for rotation and the rotation and the translation in a 3 into 3 matrix is not possible. For that particular property, what we are considering? We are considering 4 into 4 homogeneous; 4 into 4 homogeneous matrix. So, for that homogeneous matrix a particular matrix is used that is homogeneous transformation a particular matrix is use that is T r b that is the rigid body transformation. In this T r b, first is R is R stands for the rotation matrix along with x t, y t, z t.

So, the matrix will be represented like this 4 into 4 matrix; if the rotation is having 3 components, then x t, y t, z t is for translation matrix. So, if you are considering a point as x y z, then the rotation the rotation and scaling; the rotation and translation will be if you will given to the particular component, then the transform matrix you can obtained as x dash y dash and z dash. So, given the points have coordinates x y z 1 as we are considering 4 into 4 matrix, 4 into 4 homogeneous matrix. So, their transformed coordinates will be x dash, y dash, z dash and 1 which we can obtain from here. This is the operation for the translation and rotation if we are considering inverting transformation for the transformation inversion of transformation; so, x t y t and z t the inverse will be simpler minus x t minus y t and minus z t.

But for the rotation R inverse is equal to R transpose. So, inverse of homogeneous transform matrix should be in the correct order as the operations are not commutative. So, if we will consider this matrix; this particular matrix as the homogeneous transformation matrix, then we can put minus x t minus y t and minus z t in the inverse matrix and this R should be represented as R T. But in this particular matrix, the order should be should not be maintained; that means, in the in the operation the translation and the rotation this should this two properties are not commutative; that means, the rotation should be followed by the translation. For this particular matrix the rotation is the rotation and the translation are not following the particular properties. For that purpose, we are denoting and particular denoting with the help of a particular transformation that transformation matrix.

In the first page, you will be going for the translation part having minus x t minus y t and minus z t and secondly, you will be going for the rotation. In this part, there will be no rotation and in this part there will be no translation. So, finally, we will obtain the x dash, y dash, z dash and one this is our final objective; that means the transform coordinates for a particular 3D model.

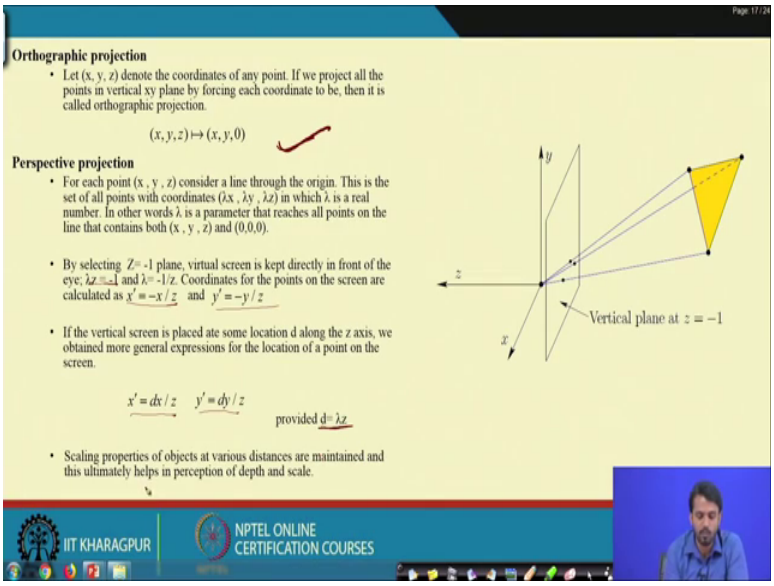
(Refer Slide Time: 23:36)



Next the viewing transformation; so, in this viewing transformation we will see a eyes view; that means, the eye is at the origin of a model. So, how the this 3D model for a eye can be 3D model can be developed for a view of eye that is represented here. That means, the let eye is an object model in which we want to place the virtual world at a position e is equal to e 1, e 2 and e 3.

So, the orientation is given by the matrix the orientation is given by the matrix x 1 y 1 z 1, x 2 y 2 z 2 and x 3 y 3 z 3. So, the rotation and translation to the eye would be applied to all the vertices placed in the particular matrix. Rather than moving the eye in the virtual world all of the models in the virtual world can be moved with respect to the eyes frame of reference hence inverse transformation is to be applied. So, what we read in the previous slide the inverse transformation is to be applied for the particular eyes frame of reference. If you want to move all the models, all the models created models in the virtual environment virtual world in the terms of eyes frame reference, then you have to apply the inverse transformation to the developed eye model; that means, the model object model placed in the eye view. So, in this we will go for this translation along with the rotation. So, 2 kinds of matrices will be considered here.

(Refer Slide Time: 25:18)

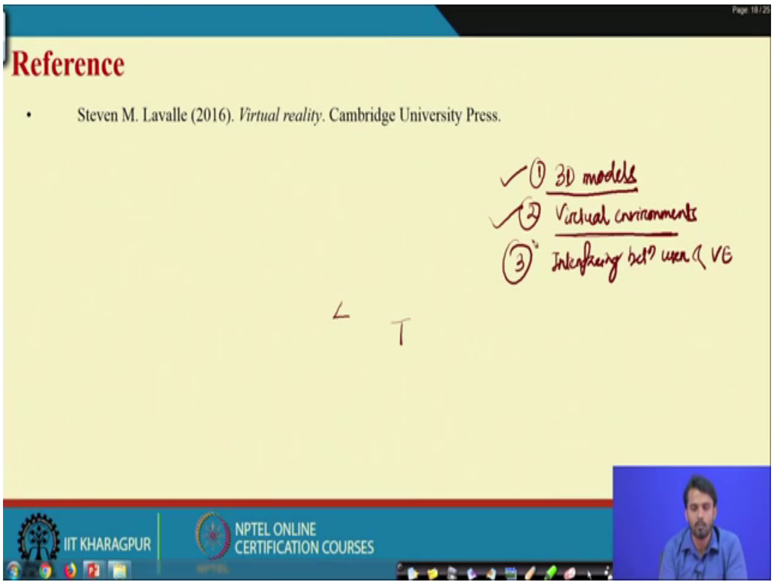


Next we will see orthographic projection and the perspective projection. So, the orthographic projection stands for the having the z axis as; having the z axis as 0; that means, if x y z denotes the coordinates of any point if we project all the points in a vertical x y plane by forcing each coordinate to be in that projected in that particular plane then it is called as your orthographic projection. In which the projection will be on x and y x and y axis means x sorry x and y plane; that means, the z axis stands for 0. So, that will be considered as the orthographic projection.

Similarly, for the perspective projection, for each point x y z consider a line through the origin you can look at this figure in which we have considered a line through origin in which the all coordinates are mentioned as lambda x lambda y and lambda j lambda z in which lambda is a real number. So, in other words, you can say lambda is a parameter that reaches all points on the line that contains both coordinates 0 0 0 ad our point that is x y z. So, by selecting z is equal to minus 1 that is the plane the planes equation is z is equal to minus 1 when the virtual screen is kept directly in front of the eye, then the lambda z is lambda z will be minus 1 then lambda is equal to minus 1 by z.

Similarly, you can calculate the coordinates for the other points on the screen that is x dash is equal to your minus x by z and minus y dash is equal to minus y by z. So, if the vertical screen is placed at the same location d along the z axis because we have considered minus 1 here. But in case along a distance d from the z axis we obtain more general expression for the location of a point on the screen; that means, x dash is equal to dx by z y dash is equal to dy by z provided d is equal to your lambda z. So, scaling properties of objects at various distances are maintained and this will ultimately help in perception of depth and scale.

(Refer Slide Time: 27:38)



So, in this lecture we saw that first we will go for, we will see that how the 3D models are created because our main focus will be 3D models, then our second focus will be on the development of virtual environment in which all the 3D models will be incorporated and the virtual environment will be created and the third one is interfacing between the user and the virtual environment. So, for the creation of this 3D models we saw that different properties are there how we are considering and Cartesian coordinates for a triangle, then we are seeing how the translation property is taken care off then we are seeing how the rotation property is taken care off for a 3D models. Then we went for the transformation.

So, different view in eyes view how the transformation can be done then finally, the virtual environment in the virtual environment how we will look at the different models that are developed with the help of 3D modelling softwares. That means, the different operation you will perform in the perform to that particular 3D models; that means, static and dynamic components, how the scaling properties of the static and dynamic components are taken care of and how the movement properties and movements translation and the rotation properties of the 3D models are taken care off. We can know with the help of this particular lecture.

Thank you.