Computer Graphics
Prof. Sukhendu Das
Dept. of Computer Science and Engineering
Indian Institute of Technology, Madras
Lecture 41
Advanced Topics:
Anti Aliasing, Color, Soft Objects, Animation, Visual Effects, System Architectures

Welcome you all back to the lecture series in computer graphics. We are towards the end of the lecture series and today we will talk about advanced concepts in computer graphics. It will be an introduction to the concepts, techniques and methods used in the text computer graphics literature, which could be a stepping-stone to concepts such as virtual reality visual realism and things like that.

(Refer Slide Time: 01:37)

Let us look back into the slide. When we talk about advanced concepts in computer graphics here and we are going to discuss in very brief concept such as antialiasing texture bump mapping, animation, soft object modeling and we will conclude with visual realism, what is visual realism.

(Refer Slide Time: 01:49)

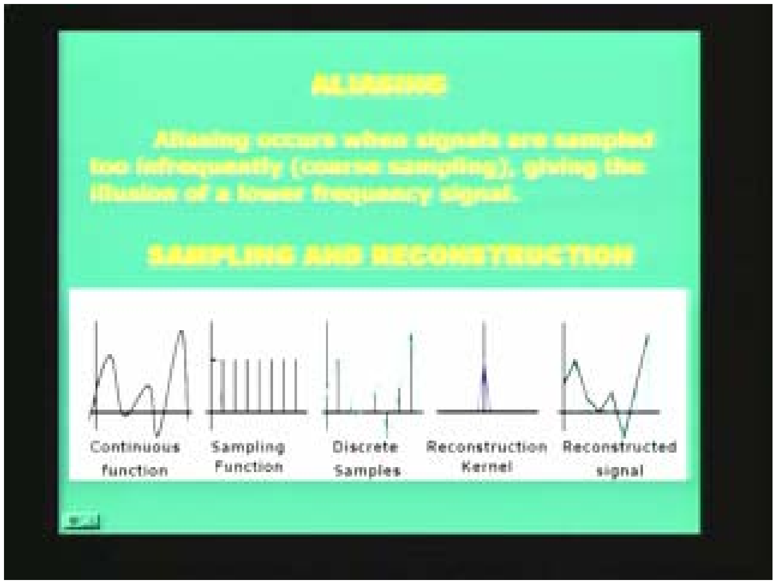
Let us start with antialiasing, what is antialiasing. Well, you remember the concept of aliasing with lines, points, circles and any other figures, which we draw.

(Refer Slide Time: 01:57)

So we need to remove the effect of the aliasing, the other terms which we were used in aliasing were jaggies or staircases. So, if we look at aliasing it occurs when signals are sampled too infrequently that is we have core sampling done that is giving the illusion of a lower frequency signal.

The concept of aliasing is a very common concept used in signal processing. We will try to see without getting into the depth of the concepts of sampling theorem and aliasing effects signal in the signal frequency domain how does it occur in computer graphics as well. That is aliasing for you.

(Refer Slide Time: 02:41)



And to just give you a brief example, a simple example for that matter if this is an analog continuous function on the left hand side for you and that is a sampling function, a sampling function is also talked of as a cone function.

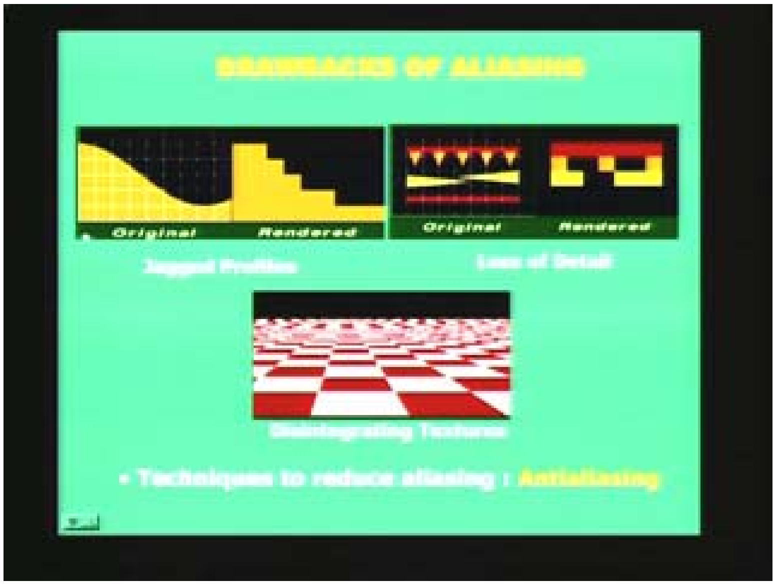
What do you do, you overlap the sampling function on the continuous function and sample it at regular intervals of time. So wherever you get a non zero value of the sampling function multiply with the continuous function value at that point in time and that is where you get the discrete samples which are the vertical lines here.

The dashed line here shows the continuous function which is analog or continuous and then you have the discrete samples. This is very common in signal processing and that is what used in computer graphics also when you have discrete samples as pixels spread over an image. But in this case we are taking a one dimensional equivalent or an example in 1D to understand the concept clearly.

I repeat again; sampling function multiplied with the continuous function gives you the discrete samples at regular intervals of time. In addition, if you take this, you can take any reconstruction kernel. I have taken a very shallow lamp function which if you convolve with this function you will get a reconstructed signal like this. This is an approximated reconstructed signal of the continuous function, which you see here and you can almost see the effect of discontinuity, the coarseness in the signal represented in the discrete digital domain corresponding to the analog continuous domain here.

What are the drawbacks caused by aliasing? Let us take an example in 2D now. The left hand side is an original function let us say two dimensional example you have discrete set of pixels. And if you have the part which should have been in color and the background being dark here, then when you try to render it using a concept which says that if the majority part of the pixel or 50 percent of the pixel is covered by this area then I will shade it completely otherwise not. Then you have this rendering effect, which is causing the aliasing effect after rendering.

(Refer Slide Time: 06:09)

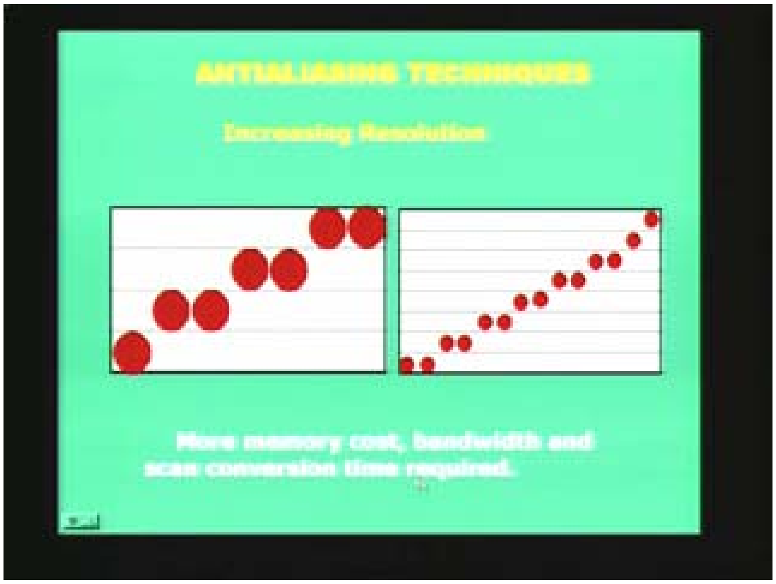


You can see this stair case effect very clearly in the dicretesized domain. That is a continuous curve, a continuous two dimensional function covering a finite 2D area. The things could be very severe in situations like this when you have the original signal here, we have obtained this example from a particular website but you can create this very easily.

As you can see here that there are various types of two dimensional shapes rectangle, triangle sphere, small and big and again a rectangle on the top and when you use concept such as area coverage in trying to fill up this you will actually obtain a rendered image which looks very different from the original image. That causes jagged profiles or loss of detail. These are the problems with aliasing. It causes disintegration in textures as well, you know.

This is a three dimensional texture gradient as you can see. Of course there is an aliasing effect as we move along the edge of the tiles. More than that as you move away towards the direction of what is called the four shortening this is a perspective projection of a planar surface painted with texture and as you can see far away, the textures do not line up very prominently as the tiles which are closer to the viewer. What are the techniques to reduce this concept of aliasing, which deals with antialiasing.

(Refer Slide Time: 07:26)



Antialiasing techniques is the one, which is not commonly used but helps quite a lot, it is trying to increase the resolution of the system. As for example, if you see, on the left hand side here, assume that your Bresenham’s line algorithm has created the set of pixels to draw a line from a point p1 here on the left bottom to a point p2 on the top right. These are the pixels you get using the Bresenham’s algorithm or the round function, as the Digital Differential Analyzer algorithm or the DDA algorithm, then you can see the jag nature or the staircasing effect.

You can probably increase the resolution to double and you will get a set of pixels like this where the line appears much smoother. You can keep on increasing the resolution of the screen and the aliasing effect will be less and less but you know that you cannot go beyond a point. Not only that, increasing the resolution of the display device from one level to another involves lot of cost in terms of money as well as in bandwidth. There are monitory issues, there are issues in terms of bandwidth which we discussed about display devices. If you remember that there is lot of bandwidth necessary between the video controller and the output display.

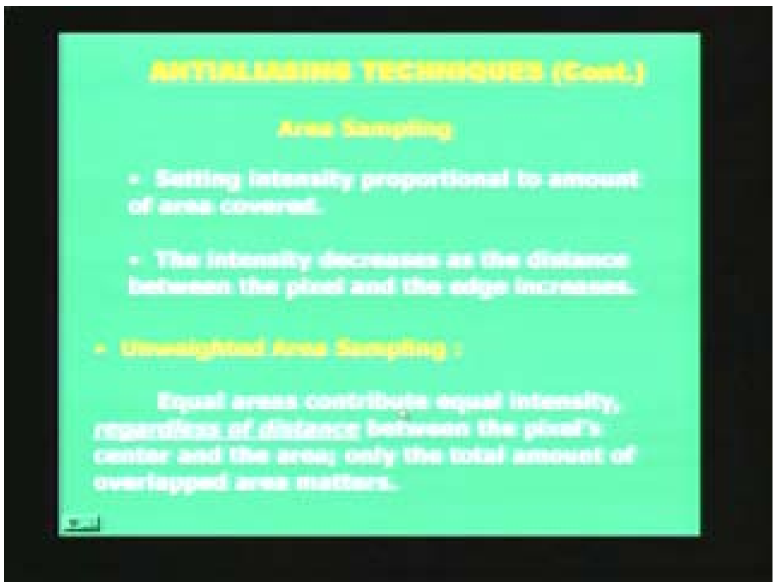
It involves more memory cost bandwidth and scan conversion time required as you need to obtain the set of pixels from left to right. The amount of time required to obtain the set of pixels is double or more than the amount of time required on left hand side. So this is not the good solution the increasing resolution. So what do you do? The next technique is called as area sampling.

In area sampling what you basically do is you try to set the intensity of the pixel value which you are going to shade and obtain using Bresenham’s algorithm proportional to the amount of area covered. We will see this with an example.

Setting the intensity proportional to the amount of area covered. Well, the intensity decreases as the distance between the pixel and the edge increases. That is also one way of trying to reduce the aliasing effect.

We will see with an example what do you mean by the intensity decreasing as the distance between the pixel and the edge increasing. Well, you have concepts such as unweighted area sampling where we say that equal areas contribute equal intensity regardless of the distance between the pixels center and the area only total amount of overlapped area matters.

(Refer Slide Time: 08:29)



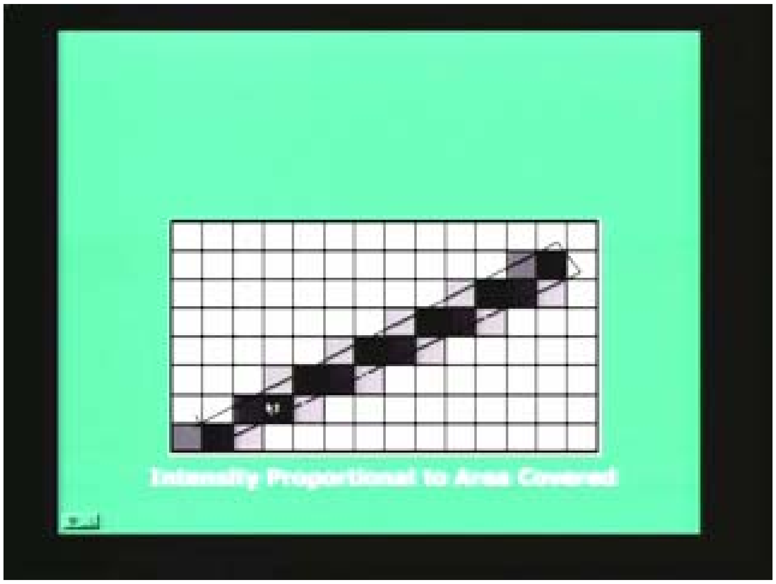
Let us understand the concept of unweighted area sampling with the help of an example.

If you see here, in this example we are talking of a line which starts from the pixel here to the point here and as you can see the amount of area covered by this rectangular strip of line is assumed to be about one pixel wide because the line is assumed to be of finite width. Whatever area it covers over a pixel is what we calculate at that particular area coverage and shade it with that particular intensity. If the pixel is almost entirely within this rectangle, let us take this pixel or even this or even this one now, you can see some of this pixels have been painted absolutely dark assume the background to be white. It could be the reverse you can have a dark background and a white foreground whatever may be the case it is a question of how much of the pixels area is covered by this rectangle.

If the entire pixel or almost the whole part is covered use the full intensity. If the area coverage is less you reduce and if it is still further less you reduce and make it close to the background.

As you can see here that the pixels which are to the left and right have lesser intensity compared to those which are in the center and the intensity is hence proportional to the area covered.

(Refer Slide Time: 10:07)



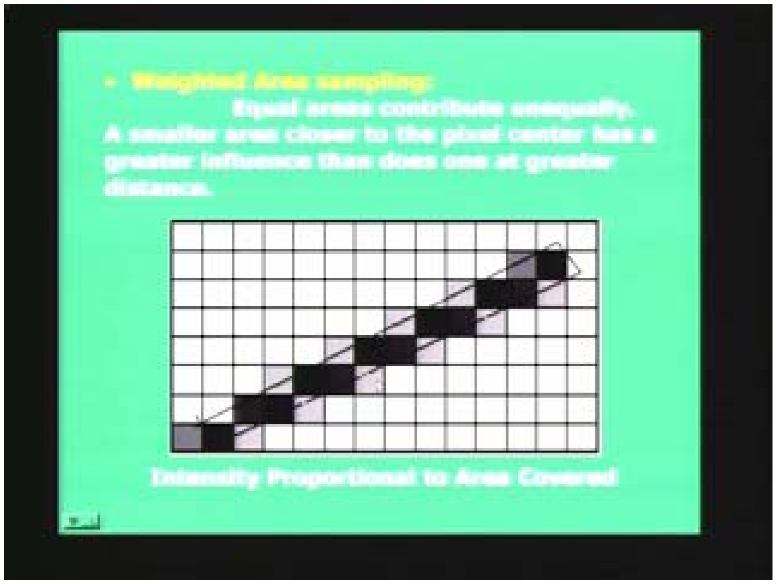
This is the concept we just read, again of the unweighted area sampling where we say that equal areas contribute to equal intensity regardless of distance between the pixels center and the area only the total amount of overlapped area matters. Since the total amount of area overlapped matters only regardless of distance hence it is called an unweighted area sampling. There exists a weighted area sampling which we will understand now.

This is the concept of weighted area sampling as given in the top of the screen. Remember, the one at the bottom is an example of unweighted area sampling because it is written clearly that the intensity is proportional to the area covered. What is weighted area sampling? We will say here that equal areas contribute unequally, equal areas in terms of over lap. A smaller area closer to the pixel center has a greater influence than thus one at a greater distance.

In case of weighted area sampling we not only have to calculate the amount of area covered by the rectangle of over a pixel but we also have to find out the distance of a center of a pixel from the center of the line. So there are two factors which one has to take in to account in terms of trying to compute weighted area sampling based antialiasing. You have to find out how much is the area coverage. That you can easily calculate in terms of the rectangular finite width line, line with a finite width which is a rectangular strip and how much that covers a particular pixel.

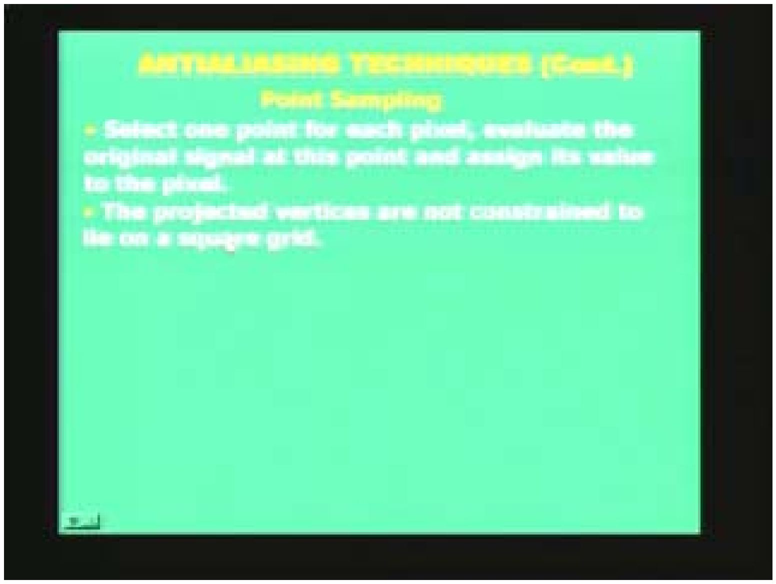
If it is full go for the full intensity if it is half use half of the intensity if the if it is nil or approximately close to 0 use a 0 negligible amount of intensity to shade that pixel and that take care of the unweighted area sampling where the area overlap only matters. In case of weighted area sampling you have to of course first calculate the area coverage number one. Number two what you have to do is your are of basically find out what is the distance of a pixel from the center of the line.

(Refer Slide Time: 12:43)



So use that 1 by d square term or 1 by d multiplied by the area coverage so there are two factors. So, if there are two pixels with similar area coverage and with same amount of distance the intensity put to that pixel will be the same but however if one of the pixels is closer to the center of the line, that will have more intensity than the one which is further away from the center of the line. This is the concept of weighted area sampling. We will wind up this concept with this statement we will read again where we say that equal areas contribute unequally mind you. It was contributing equally in the case of unweighted area sampling. But in the case of weighted area sampling equal areas contribute unequally. A smaller area closer to the pixel center has a greater influence than one does at a greater distance.

(Refer Slide Time: 13:28)

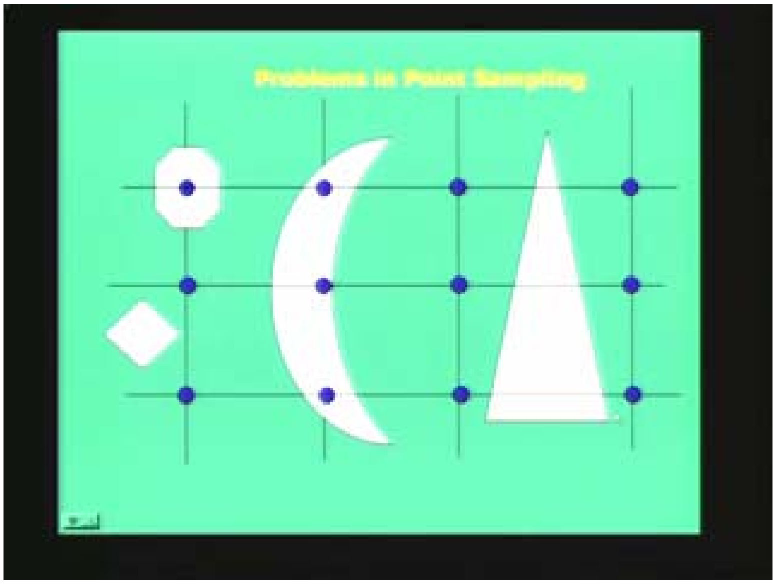


Let us look at other concepts of antialiasing which is based on point sampling where in this case we select one point for each pixel, evaluate the original signal at this point and assign its value to the pixel.

What does this mean? There are certain drawbacks. We will read this drawback and see the example. The projected vertices of a polygon let us consider the vertices of a polygon are not constrained to lie on a square grid. Let us see this particular example. As you can see the problems in point aliasing here what we are basically checking here is whether the pixel is falling within an area. If it is falling within an area you use the color of that particular area to shade it to shade the pixel. But if the object is so shallow, so small that it relies within the resolution grid, remember, let us assume that the background is the analog world, the foreground is this grid of pixels which we have overlapped on this analog area and we are sampling the continuous analog world using a 2D array of points.

You remember the cone function we talked about in aliasing which was an area of 1D vertical strip. So you can visualize that as if you have a 2D array of thongs which you are putting on the analog world and you are sampling each of those tip of the thongs the continuous analog world. That is what you have to imagine. The bottom is the analog world continuous domain, a discrete domain on the top and you are using a set of points to sample each particular point x and y in the continuous domain.

(Refer Slide Time: 13:58)

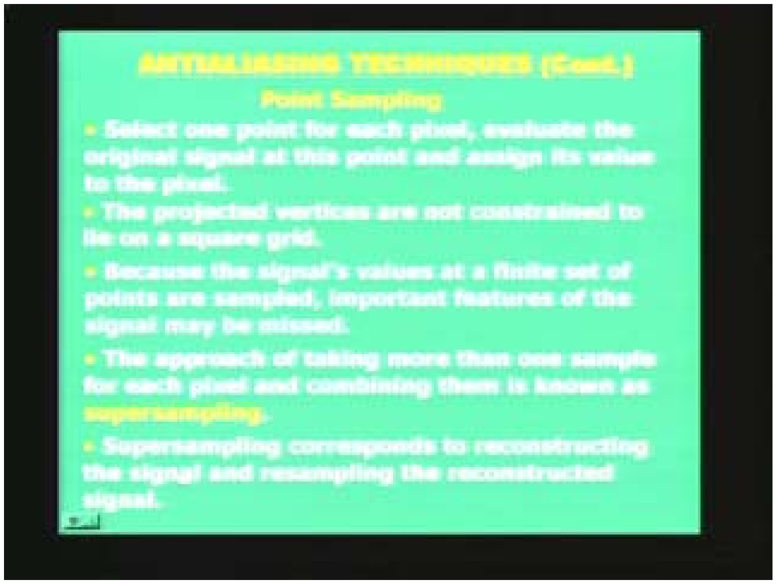


Therefore, if it is the case, there is absolutely no problem may be when areas cover a pixel. Now, look at this particular rectangle or triangle. Here none of the vertices may fall actually on the intersection point or a grid point and the entire area may actually fall outside the grid point. These are the problems, which could be associated with the points sampling where none of the grid pixels will be illuminated due to the presence of this triangle. That is the problem with point sampling where we will see that the projected vertices are not constrained to lie on a square grid.

So because of the signals values at a finite set of points are sampled the important features of the signal may be missed we have already seen that with an example. The approach of taking more than one sample for each pixel and combining them is known as the concept of super sampling. That is the method by which we used to count the aliasing problem in point sampling. The super sampling is similar to the concept of super resolution in image processing where in super sampling you take more than one example for each pixel and then combine them to obtain sample points.

Super sampling corresponds to reconstructed signal and resampling of the reconstructed signal. I repeat super sampling corresponds to reconstructing the signal and resampling again the reconstructed signal.

(Refer Slide Time: 15:58)



Continuing with another method of antialiasing which is called filtering is done by removing high frequency components from the original signal. The new signal can be reconstructed properly from a finite number of samples.

Well we did say that aliasing corresponds to as if some core sampling. Now, whenever core sampling is done without an effect of smoothing it typically results in jaggies or aliasing effect or the stair casing effect. There will be high frequency components present in the signal equivalent to saying that there are some ripples or stair casing effect which is the effect of aliasing we observe in computer graphics. The similar concept also exists in concepts of digital signal processing. So we have to remove this high frequency components by method of filtering. Let us see this example of the original image.

As you can see the textures which are very close the textures of the tiles which are very close to the viewer or closer to you, you have almost no effect of aliasing. But as you go further beyond in this part of this planar inclined surface of the texture which is inclined here as you can see through the beaker or within the beaker there is a strong effect of aliasing. You must come very close to the screen to observe the effect of aliasing here as well as there as well as here. I cannot enlarge this image much larger because I have to squeeze in the effect of antialiasing using filtering here.

(Refer Slide Time: 17:13)

So please come to the screen and observe here the effect of aliasing at this region even at this as well as in this region there is barely an effect of aliasing at this location. Let us try to find out what we mean by high frequency components. Well, if you pass it through a high frequency filter, basically it suppresses all the low frequency components in terms of brightness values. Whatever brightness it is the brightness contrast within the tiles are removed and in fact all the aliasing effect comes here.

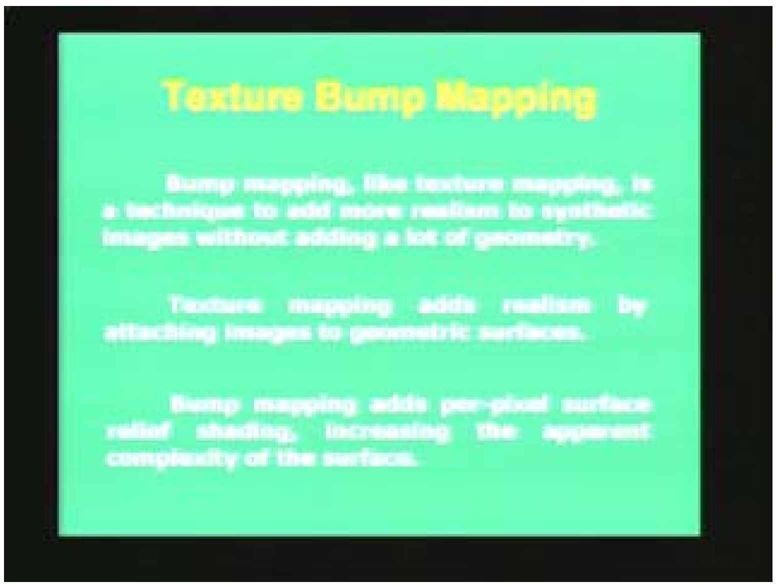
Logically, I should not algebraically subtract this high filtering signal from the original image but basically pass it through the low pass filter. A typical example of low pass filtering is to use a Gaussian function. You can see that you get this image. Yes, it appears blurred compared to the original image but if you can carefully see here, again, I would request you to come close and observe these areas, one area here is very light and you can see it is a blurring effect but the effect of aliasing has disappeared. There is a strong effect of aliasing here, which has almost disappeared in this range here. This is one way with the help of filtering. You need to use a low pass filtering to avoid the concept of aliasing. That concludes the discussion of anitaliasing.

We have just some time to discuss a few of this techniques and we have to move forward and discuss about other concepts of advances in computer graphics. Let us look at the next topic known as texture bump mapping. What is texture bump mapping? It is like a texture mapping, when you map a texture on to a surface but it is a technique, which adds more realism to synthetic images without adding a lot of geometry.

Texture mapping adds realism by attaching images to geometric surfaces whereas bump mapping adds per pixel surface relief shading increasing the apparent complexity of the surface. We will see what you mean by this later on as we go along. Bump mapping adds per pixel surface relief shading increasing the apparent complexity of the surface.

Bump mapping is done for surfaces that should have a patterned roughness. Examples include surface of oranges, surfaces of strawberries, stucco, wood etc. Wood means you can probably visualize the trunk of a tree. Stucco is the textured three dimensional ripple texture surfaces done for indoor decorations or even outdoor decorations for that matter. So, for any rough type of surface where you can visualize ripples existing on a surface relief shading is provided with the help of bump mapping. You can almost visualize the scenario when you have a road with lots of bumps in front of you. That is of course very regular in periodic in a sense that there is absolutely no randomness. You can visualize a series of bumps on a road by simple sinusoidal variations but there is lot of randomness when you talk of a texture.

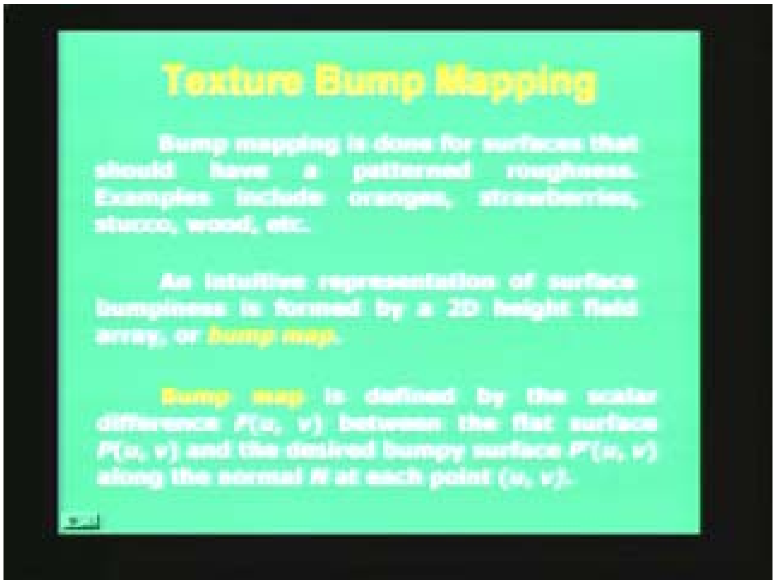
(Refer Slide Time: 19:18)



Not like a random noise where there are methods which can do that but we are talking of some sort of randomness in the shading effect when you talk of texture bump mapping. So these are examples of surfaces, oranges, strawberries, stucco and wood where the bump mapping is done for surfaces that should have or you want to provide a pattern roughness on the surface.

An intuitive representation of surface bumpiness is formed by a 2D height field array. We are talking of a 2D height field called a bump map. So if you have a planar surface z equal to constant instead of z equals to f(x, y) and then you assume that you put another layer on top of it where the z function and the second layer is sitting on the planar smooth surface and that provide the bumpiness or the roughness. That could be a random sort of a variation to provide the texture but it basically implements the bump map for you. That z equal to function of x y could be the small randomly varying function. It should not be too random like a noise but there should be some randomness in terms of creating this bumpiness on the surface. So you need a bump map which is also called as 2D height field array to create the surface bumpiness.

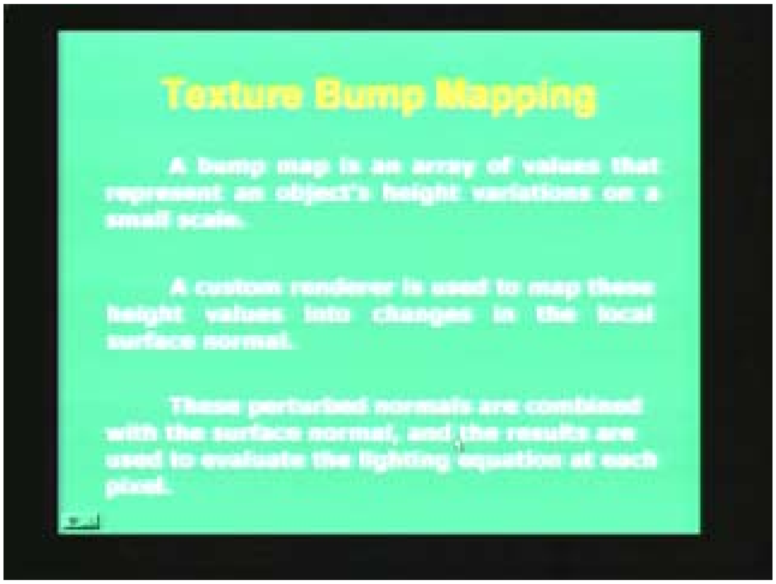
(Refer Slide Time: 21:43)



Well, the bump map is defined by the scalar difference f of u v. Well, although you have two parameters u and v it should be scalar. But at any point u and v there is a scalar difference f between the flat surface p u v and the desired bumpy surface p prime u v along the normal N at each point u, v. So this is the one which is varying. The desired bumpy surface creates p prime which is a random variation at u of u, v along the surface normal and that sits on the flat surface p u v and creates your f u v that is the final bump map. So you talk of a layer p u v another p prime sitting on p and that creates your f the bump map. We will see example as we go along but let us read a few more concepts. A bump map is hence an array of values that represents an objects height variations on a small scale.

I repeat again; a bump map is an array of values that represents an objects height variation on a small scale. A custom render is used to map these height values into changes in the local surface normal. Well, we do not have time to go through these equations. But I would like you to refer to Phongs shading models where, you remember, we had taken the normals at the vertices and use that to interpolate along the edge and from that edge we used the scanline method to compute the normals at each surface and then use shading. So a concept similar to that can be used for texture bump mapping where you recomputed the surface normals at each vertex and then use relief shading on top of that to create the effect of texture bump mapping. So a custom render is used to map these height values into changes in local surface normal. Then these perturbed normals are combined with the surface normal and the results are used to evaluate the lighting equation at each pixel. We are just mentioning this.

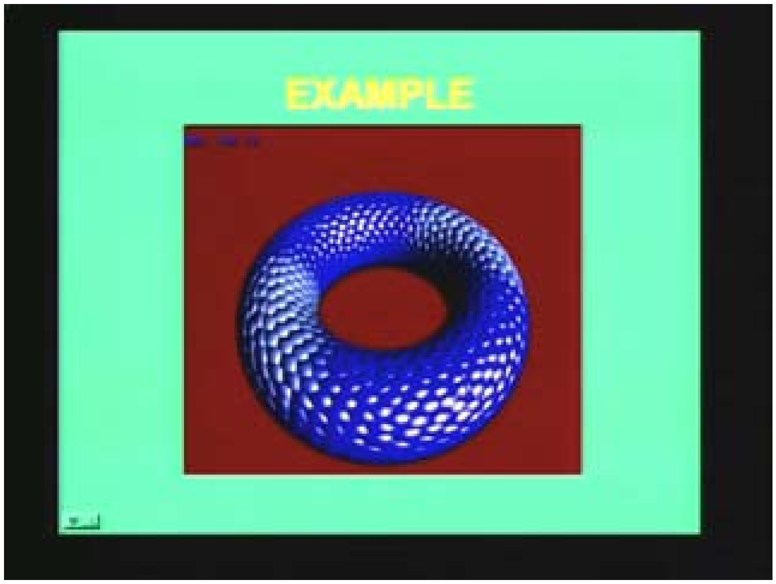
(Refer Slide Time: 23:42)



This perturbed normal remember, are obtained when the normal height values are changed and that causes a change in the local surface normal and then this perturbed normals are combined with the surface normal and the results are used to evaluate the lighting equation at each pixel.

Let us look at a few simple examples. This is the one, which is an example of texture bump map. This has also been obtained from an example created by one of the institute websites. That is an example as you can see the surface roughness and the brightness fashion. Well, texture bump map is one of the techniques which is used to create this nice visual effects. But there are other visual effects possible, I will probably list them towards the end of the lecture today and a few of them will also learn along with aliasing you need to provide texture bump mapping and a few other methods to really create realistic pictures.

(Refer Slide Time: 23:45)



Before we move onto another concept let us look at a very realistic picture as to what we are trying to make. If you look at this picture again simulated this is courtesy from this particular website. You can try this there should be the gallery of other objects as well realsoft.5. As you can see this is a realistic scene rendered using various types of concepts.

(Refer Slide Time: 24:42)

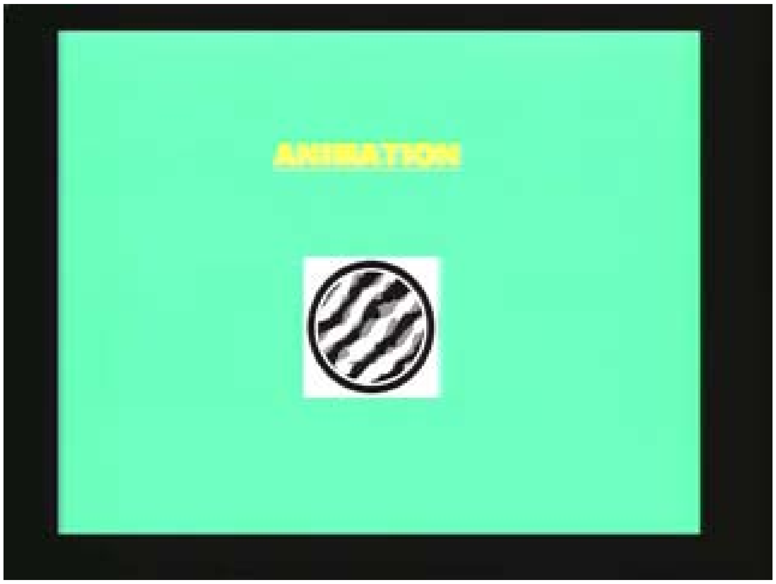
You use nurbs which are non uniform rational B-Splines. You must read about this. We discussed about B-Splines in our lecture and slightly advanced equations we will talk about nurbs which are used in all sorts of vtks and softwares, advanced toolboxes like maya, or realsoft has its own engine which will use nurbs. That is very common for creating polygons, meshes and various types of complexes such as even it could be a dinosaur another complicated structure so people will use nurbs. Nurbs have been used along with that image maps, bump maps, texture maps, procedural noise which is a very interesting concept. There are concept of procedural noise which deals with concept such as purling noise, three dimensional noise, it is not truly random like a noise but it is not rhythmic like a sinusoid or not periodic like a sinusoid, it is somewhere in between.

There are very interesting concepts if you go through a course of advances in computer graphics or virtual reality or visual realism you must study concepts and various other concepts including procedural noise and depth of field. This depth of field we have studied when we talked about shading. This is an effect where the distance object should appear lighter than objects that are closer to you that is the effect of the atmospheric catenation was an example which we studied for depth of field.

You can see that I have used this website. Among the different galleries of objects we have the surface bump map on the apple we have eggs, we have a vase which has two layers; one layer of blue another is a shiny layer on top of that and of course another texture procedural noise used for shading this particular vase itself. That is a nice example that appears almost close to reality. This is a very nice example I taught which you should see and which one can try this.

There are various concepts, some of them which we study, some of them we do not study in this basic introduction to computer graphics, the last lecture almost towards the end of the lecture about different advances in computer graphics. We will just introduce some of the basic concepts and terminologies of advances in computer graphics but may not be able to cover all topics. We will look at the next topic that should be of interest to many especially for video game designers here is the concept of animation.

(Refer Slide Time: 27:12)

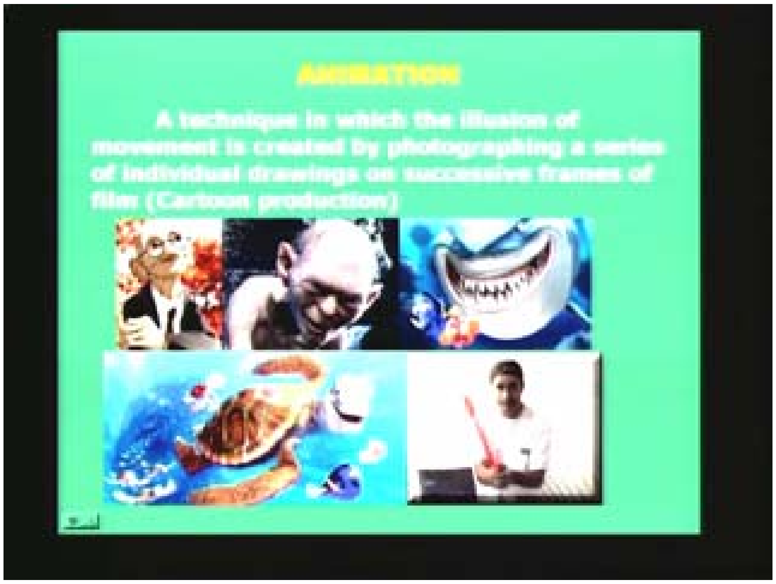


What is animation? Animation is something that moves. What are the applications of animation? There are special effects in movies and TVs that need animation; video games need animation, virtual reality and everything in front of us moves. So, when you want to create reality in virtual domain things must move. When things have to move they must move efficiently in a smooth manner not in a jerky manner it will create an aliasing effect also in motion.

(Refer Slide Time: 28:04)

So virtual reality, simulation, training and in military, medical applications, robotics and animatronics, well that is an example of robotics and movements of robots or electronics mixed with animation I should say animatronics, visualization and communication as well. What is animation? It is a technique in which the illusion of movement is created by photographing a series of individual drawings on successive frames of films. Typical example is cartoon production.

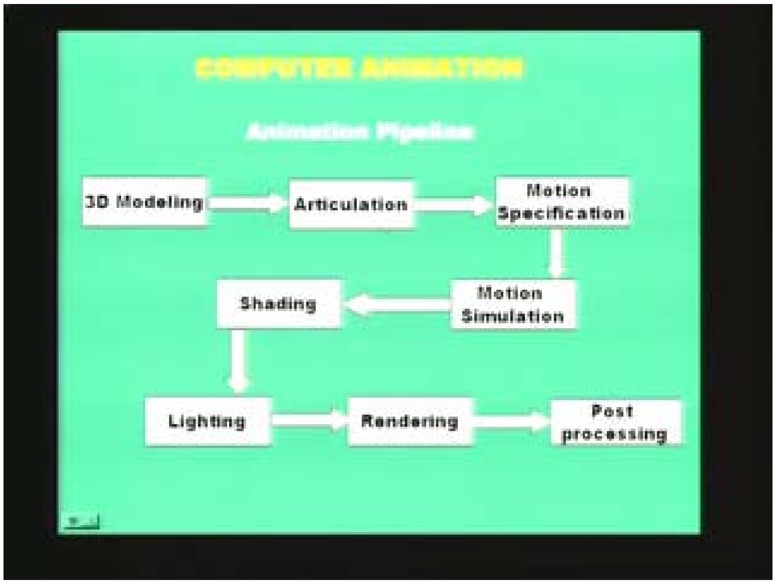
(Refer Slide Time: 28:42)



These are various examples. You can obtain some of these on your web. Of course these are static pictures but these are some examples of animations needed. I only got a static picture in some of these. Of course, this is also an example obtained from the web page of the institute that works on movies of star wars and these are of course cartoon movies.

Well, the animation pipeline can be visualized as a stage of blocks like this like a flowchart. We discussed about a 3D modeling then we discussed about articulation of the models which have to move and specify the motion in terms of motion specification, simulate the motion, shade individual pixels and regions and polygon. In addition, we talk about light source, render them and do some post processing to eliminate the effect of aliasing, jaggies and jitters if they are present in this portion. These are examples of animation. Now, there are of course various categories of animations possible. We will just touch upon a few of them. There is a difference between what we call as 2D and 3D animation.

(Refer Slide Time: 28:57)

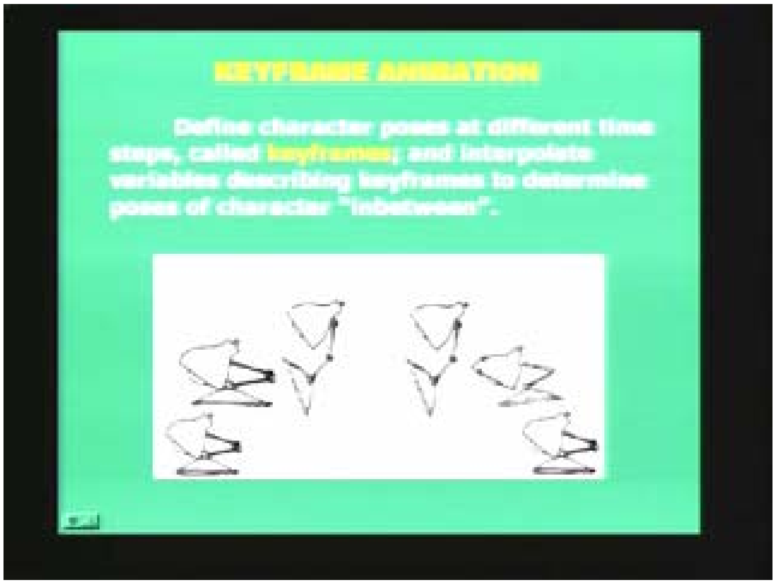


Well, we do not have an example but I will try to illustrate the difference between animation characters in 2D and 3D with a static picture. Well, you can see here that this is an example of a 2D animation character. That is because this can only move in two dimensional structure but there is 3D in formation here. This person not only can turn around but can move up and front in 3D and can revolve and rotate around. This character in the right hand side is mainly meant for two-dimensional animation. The second static picture is meant for 3D animation.

(Refer Slide Time: 30:12)

When we talk of types of computer animations, there are three types. The key frame animation, procedural and motion captured animation. We will read what key frame animation is. Well, we define characters, poses at different time steps called key frames. We will see what are these where we define character, poses at different time in steps called key frames and interpolate variables describing the key frames to determine pose of characters in between. This in between is a very nice concept with key frames which we will see here. Let us look at trying to create different poses of a particular character. I have taken this example of a table lamp that is the first pose. The second pose is where it moves a little bit up. This is the third pose where it springs up so it is a dynamic table lamp and then it moves here and it is falling down, it is coming back to its original position so it may move in this arc.

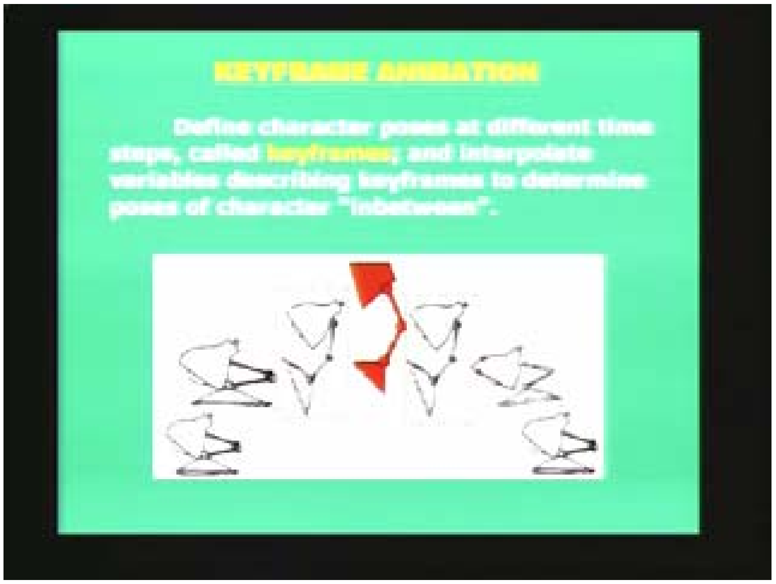
(Refer Slide Time: 31:03)



It is possible that you might specify that it may move in this arc but it could be in the same location where it could spring up and down at the same point without moving from left to right as given in this curved path here. That is possible but also in the same position if you see that it could jump up and down and these are the six different poses at the same location, so that is possible. But of course it is better to provide a lively animation by moving the entire structure as well at different time instances as when the shape of the structure changes.

So what is key frames and in-between. Let us say there is a big amount of discontinuity here between and I select two key frames here; this is key frame number one, key frame number two and I am interested to create another frame simulate another frame in between. This is the concept of inbetweening which could create animation frame like this in-between.

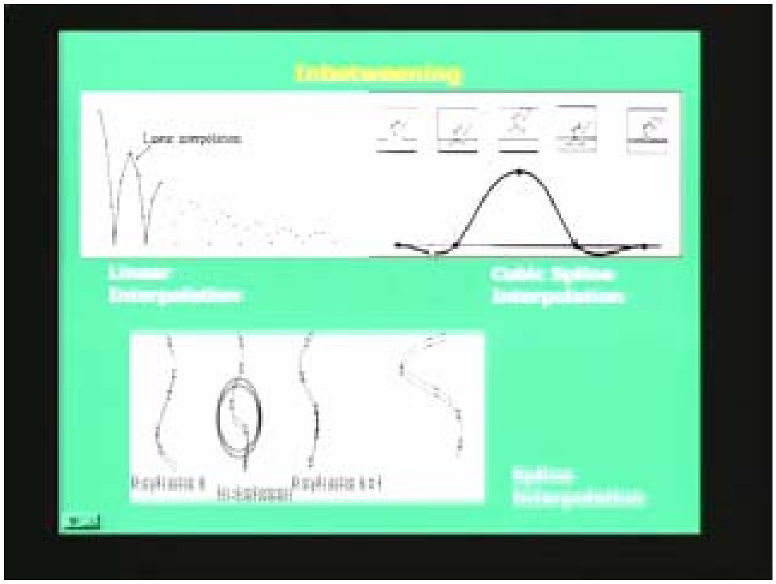
(Refer Slide Time: 32:01)



I must admit here that you must carefully observe that if you only use these two frames, let us go back. If you use only these two frames to create and interpolate in between you might get the same frame back, you will not get the one in red I just showed. What you must do is try to interpolate back from the previous frames and the next frame and then only it is possible for you to get a frame like this in between. So you need little bit of mathematical skills not simple interpolation to create an in between frame from key frames. Let us look at the different topics of interpolation.

Inbetweening concepts of linear interpolation: It is not a very sophisticated method if we are talking of an analog trajectory like this, a modulated sink curve or a dam that is sinusoid rectified, sinusoid you can say. Now, if we do an interpolation in between you will get frames that are not accurate as you can see here, so typical linear interpolation is not good. What is done is a cubic spline interpolation. As you can see here these could be a set of key frames which I could use to create any frame inbetween. Now I will use the set of all these five frames and find out the paths of this set of points within the structure from one point to another. Remember, this key frames does not show that the movement of patterns like in the previous key frames of the animation I showed in the previous slide but I am just showing you the position of the structures, at what time frame, t is on the right hand axis.

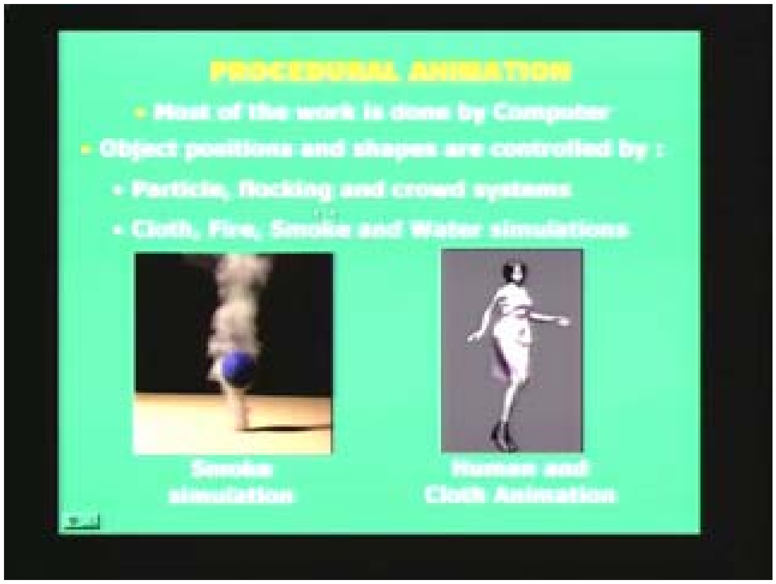
(Refer Slide Time: 33:56)



Vertical axis typically shows the curve which you will fit across the time frames and you can use as cubic spline interpolation to come up with a frame inbetween at any instant of time. But you must use all of them. If you use just two of these you will get a linear interpolation and you will may not get the actual change of the animation carried out. This is another example of spline interpolation, which gives you a very good idea. I hope you can see the lines in between. So you take a key frame k plus 1 and what you do basically is use a set of key frames K may be k minus 1 as well k plus 1 k plus 2 to fit a spline and get a frame in between. So these are examples of inbetweening between frames. Very really if there is a smooth amount of variation very less amount of movement for simplicity for fast implementation you can use linear interpolation. But typically it is better to use cubic spline or some sort of spline interpolation to obtain key frame animation.

Let us move to the next part of the animation, which is procedural animation. We will go through the key points in a procedural animation most of the work is done by a computer. Object positions and shapes are controlled by particles flocking and crowd systems. Typical examples are cloth, fire, smoke and water simulations.

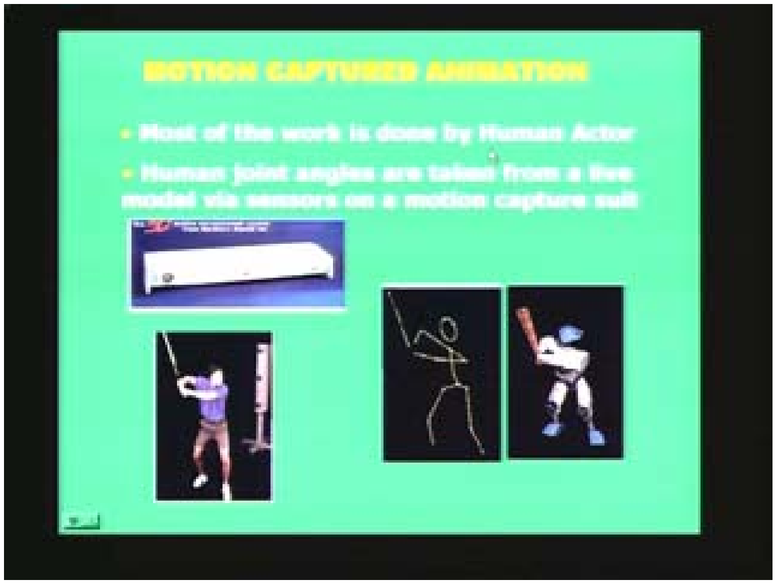
(Refer Slide Time: 35:18)



These are certain examples of static pictures about smokes simulation where the concept of particle systems are used for smoke flocking and curved systems and soft object modeling is done for human and cloth animation. You can make them move only if the modeling is very correct. There are of course these are advanced topics which I am not covering about particle flocking and curved system to generate flop, fire and even water simulations.

What is motion capture animation. This will be of interest to some people. The human actor does most of the work. And the human joint angles are taken from a live model via sensors on a motion capture suit. This is an example. People who are used to outdoor games such as baseball, cricket etc this concept should not be new to them because we know in the past that there are this laboratories which not only try to simulate but experiment and also try to obtain to correct the motions and the pose of certain sports persons.

(Refer Slide Time: 35:48)

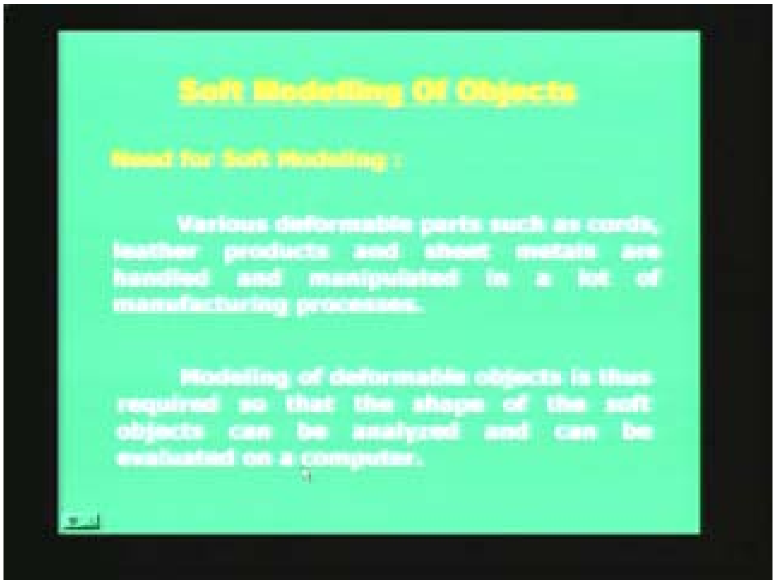


If you take the example of cricket , the bowling action of several bowlers from within the country and abroad has been tested using simulation labs in Australia. This does the following, where they will put different light emitting devices at the various joints of the human body and there will be sensors that will capture this motion. After capturing the motion it will try to find out and simulate this interactivity. Then you can put it on a robot to create this animatronics which we discussed about. Then you can create the robot movement as smooth and as real like a human being provided you know the motion parameters the motion path of the way human beings move when they are either walking or playing a certain game or doing some other activities.

This is a motion capture animation where you capture the path and the parameters of a human being when a certain person moves or in a certain activity. You capture the motion of all the parts of the human body and then put those parameters inside the simulation of the character at the respective points and move it in a similar path. That is the motion capture animation. We have seen key frame animation procedural one and examples of motion captured animation as well.

We move to the next topic which is also of interest called soft modeling of objects. What is soft modeling of objects or what is the need of soft modeling. Various deformable parts such as cords, leather products and sheet metals are handled and manipulated in many manufacturing processes. You need to simulate them so modeling of deformable objects thus requires that the shape of the soft objects can be analyzed and can be evaluated on a computer.

(Refer Slide Time: 38:12)



You need to simulate them not only for creating movies, creating video games but also it is necessary to give a simulation for a designer, for an engineer, for an evaluating team. You may not need all objects that we deal within the factor and real life is hard rigid objects. Some of them are soft. Typical examples of it is a cloth, some fruits, the tennis ball is also a soft object, but cloth here, muscle tissues which we have in our body are all examples of soft objects and it is often necessary to model them as well to create a real effect. So that is the need of soft modeling where we talk of modeling of deformable structures or objects and it is required so that the shape of the soft object can be analyzed and can be evaluated on a computer.

What are the applications of soft modeling. Computer aided design and computer drawing applications we talk about these as a class of applications of soft modeling where deformable models are used to create and edit complex curved surfaces and soft solids. Computer aided apparel design is used to simulate fabric draping and folding as we have already seen that example. One such example in the case of an animation where we saw a structure of a human being and with a cloth draped on top of her so that was an example where you needed soft object modeling to simulate the apparel or the fabric.

(Refer Slide Time: 39:56)

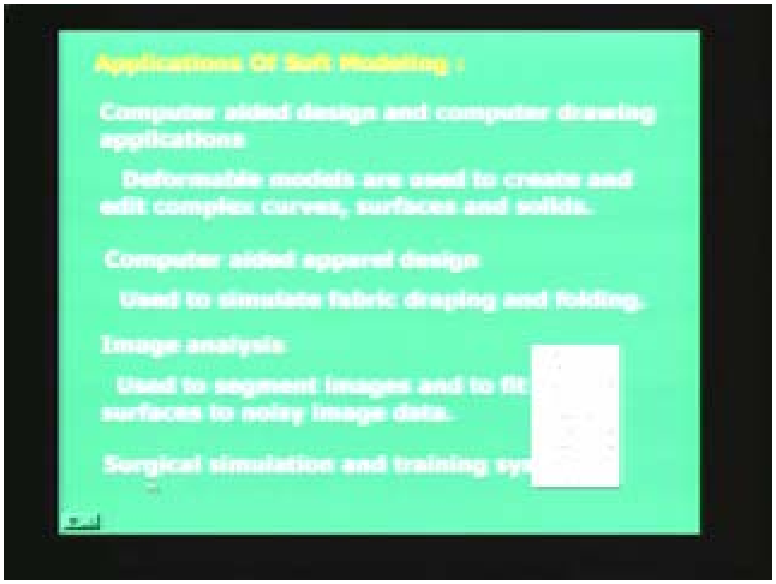
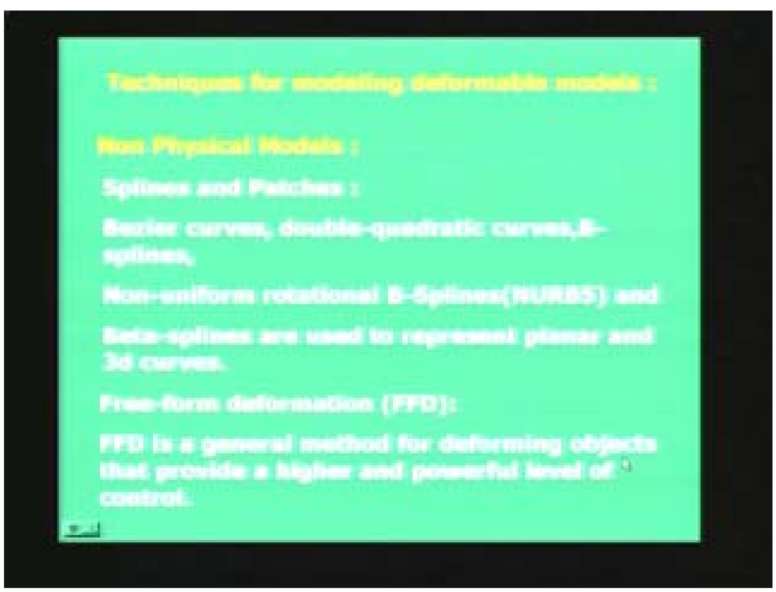


Image analysis is used to segment images and to fit curved surfaces to noisy image data. That is not a best example of computer graphics but it is also used and surgical simulations and training systems. That is another good example of soft modeling.

Techniques for modeling deformable models:
What are the techniques used? There are lots of them. We will just go through a few of them and will take one example to study in detail with the time available to us.

(Refer Slide Time: 40:56)



Non physical models involves splines and patches. We talk about Bezier curves, double quadratic curves and B-Splines which are used to model non physical models.

Non uniform rational B-Splines we discussed about the nurbs already. We saw a realistic example after we discussed about texture bump mapping. Nurbs are very common and also people use beta splines used to represent planar and 3D curves.

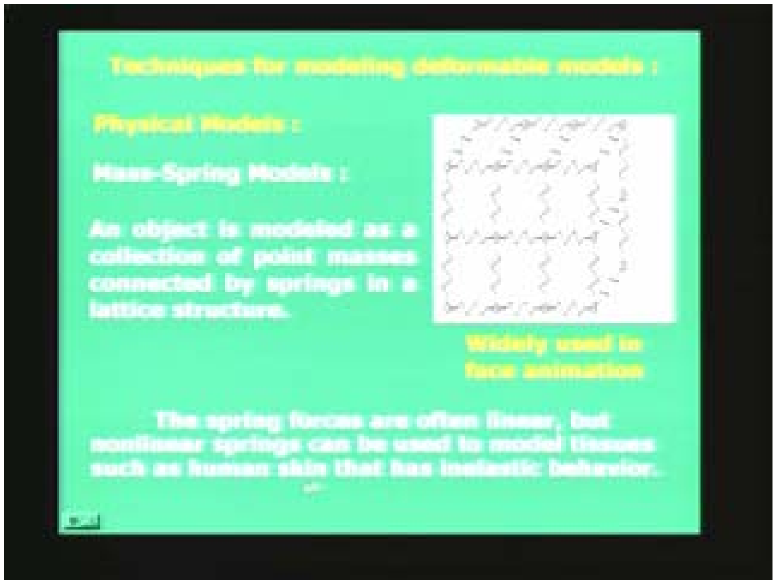
Well Free Form Deformation FFD is a general method for deforming objects that provide a high and powerful level of control.

Free Form Deformation is a concept which is a general method for deforming objects that provide very high and powerful level of control.

Continuing with the discussion, with physical models, well this is the most common example which is used called the mass spring models, what is done here. An object is modeled as a collection of point masses connected by springs in a lattice structure. I repeat it will come with this example in a figure very soon. We are talking of an object which is modeled as a collection point masses. So each point structure will have a mass. It is a small object with a mass and they are connected by springs in the form of lattice structure.

Let us see this example and this is widely used in face animation as well as for modeling tissues, springs, soft objects such as sponge structures. These are the points that will have certain mass and there are springs in between. All of them could be of the same value. This is the spring constant k the mass m all the structures may have the same mass or it may have a different mass struck to each point. The spring constants could be different or they may be same all over. Therefore, this is typically a cubic structure as you can see here.

(Refer Slide Time: 41:41)

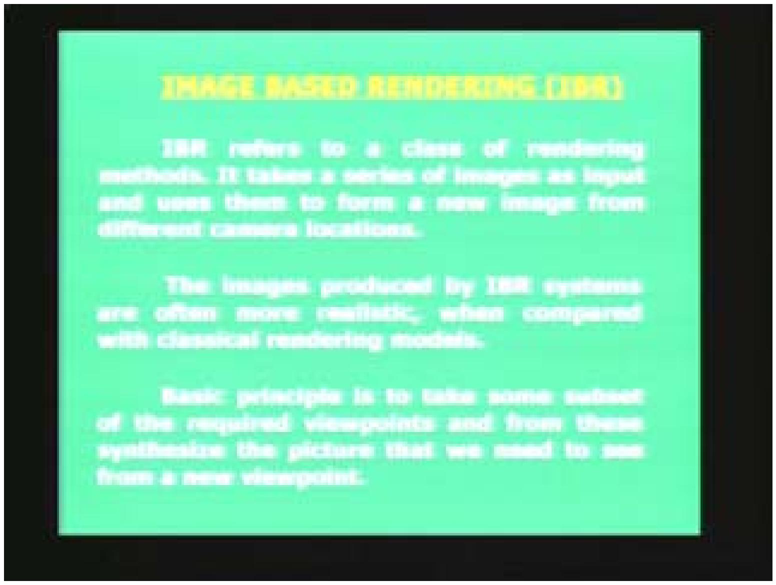


But you can form any structure, a 3D array of points put mass on each of these points and springs connected instead of a rigid structure. Where in the case of a wireframe if you remember, there was just a line joining between two vertices. It was very rigid under any transformation like rotation, translational and scale except may be shear it is invariant. But you can apply a force on this soft object this mass spring structure and you can try to visualize what happens when you apply force on a spring. You know the springs theory from the concept of basic physics. Now a set of springs put together and you apply pressure some of these springs will contract some of them might expand and that will allow the formation of the structure. That is the concept which is used in the mass spring model here. I repeat an object is modeled as a collection of point masses connected by springs in a lattice structure like this. The spring forces are often linear but nonlinear springs can be used to model tissues such as human skin that has inelastic behavior.

I repeat; spring forces are often linear we can have a spring constant k and the equation could be linear. But you can have nonlinear springs as well which can be used to model tissues such as human skin which has not elastic but inelastic behavior. This brings us to the conclusion on soft object modeling. We will move on to the next concept based on the time limits which will bring reality into the picture which is called Image Based Rendering. Let us look at this example of Image Based Rendering or IBR as it is called. IBR or Image Based Rendering refers to a class of rendering methods. It takes a series of images as input so it might be a video.

Often video based rendering and Image Based Rendering are used interchangeably. If you use a single image or a couple of images may be at the most you can talk of image base rendering. But if you talk of a sequence of series of images you can say you are using a video based rendering as well. What does it do? It takes a series of images as input and uses them to form a new image from different camera locations. I think people have seen this concept of image based rendering in movies and even in sports. It is a concept such as let us say you observe my face from the front. Now you are observing me with the camera that is shooting this lecture.

(Refer Slide Time: 47:10)



It is possible that there could be another camera on the right side of me which could also be taking this picture. That is also possible. What could happen that means I am using two cameras one on the front side and another on the side and the side camera will capture my side view. It will capture my side view as I speak. This is an example of the camera if I had used a side camera with a side angle and if you have two such cameras it will take two views.

Now what is the problem that I am trying to solve here. With these two views, as I am talking to you, the front and the side camera is taking my side view, let us say I would like to simulate a view of an image or a video of a camera. Imagine it does not exist but I want to simulate those shots for a camera that exists at an arbitrary oblique angle say at 45 degrees with respect to my face. So probably, I want to simulate the video of the speaker talking at a particular angle as you can see me here. You only have the front and may be the side view. That is an example of Image Based Rendering trying to simulate views from a different camera location. It could be at any arbitrary angle of course all camera angles may not be possible.

You need lot of camera views to really create different types of views. At least from two camera locations you can obviously simulate a third and the third is the correct you can also generate a fourth so that is the concept based on Image Based Rendering. Of course, there are other applications of Image Based Rendering where you try to provide a texture on a surface whether the texture is borrowed from a realistic image that is also an example of Image Based Rendering. Moreover, it gives you a good simulation of reality because you are using real images not artificial ones real images, shadings, shadows, intensities obtained from real images to shade it on a picture that is a simulation. That simulation model itself could be computer graphics simulation but you are not going to calculate the intensities, brightness and reflection patterns on any surface whether it is human beings speaking or any other event sports or whatever is happening.

You are using images, pixels, intensities, textures and all other artifacts such as reflection to obtain those pixel values and paint it on a scene for which the camera angle is not available or the camera shots are not available. That is a very interesting Image Based Rendering for you. The images produced by IBR systems are often more realistic when compared with classical rendering models.

The basic principle is to take some subset of the required viewpoints. What are the concepts of these viewpoints that I just discussed about these viewpoints right now. As you see right now in front of me, there is a camera recording my lecture so that is a viewpoint. And of course you can have another viewpoint were the camera is shooting from this side. Therefore, these are two viewpoints and you would like to generate a third viewpoint and simulate it. So the basic principle is to take some subset of the required viewpoints and from these synthesize the picture that we need to see from a new viewpoint. That is image based Rendering for you. Well there are many methods of Image Based Rendering available in literature. You can have hours of lectures of these concepts mixed with digital image processing which should be covered in next two slides.

There are concepts based on image simulation, there are concepts based on the method of animation in which we talk about the motion capture are used for Image based Rendering. We will discuss just two of the common methods on Image Based Rendering.
And the two methods are; one is called mosaicing which is also basically an area covering under image processing where we match multiple images by aligning and pasting images to have a wider field of view images. There is more of image processing than Image Based Rendering but that also could be used and that concept will also introduce this terminology of light field rendering or what is called as lumigraph where the light field rendering is the process of rendering novel views of a scene captured by the light field function.

(Refer Slide Time: 48:53)

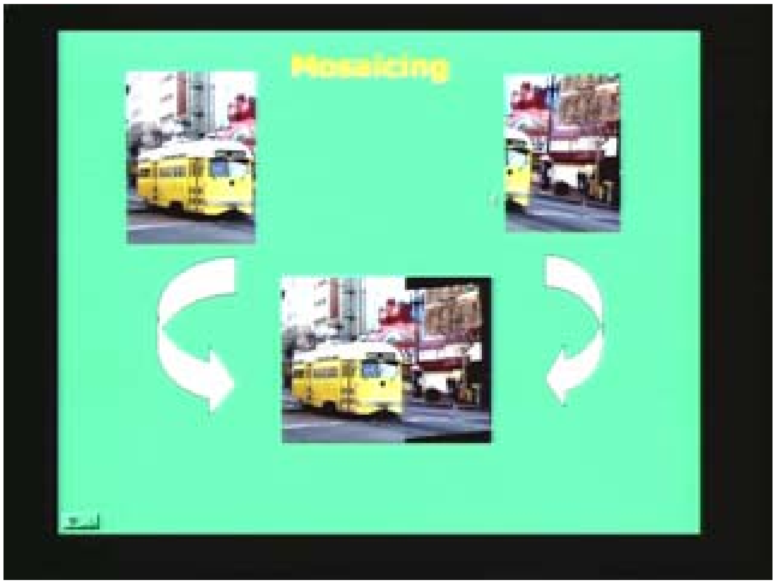
Well, this light field rendering lumigraph is the concept where you define the light rays. It is an extension of light ray tracing. Remember, light ray tracing in shading my side where light rays are coming out from the centre of the COP through the image pixels and moving towards the objects. Well, you can define light rays like that in the case of lumigraph we discussed about a 4D vector defining my light source.

You can visualize as if you have a point u1 v1 at the near plane and a point u2 v2 at the far plane and a vector that connects these two to simulate your light ray. And that is what you vary to simulate the effect of light field function. So these are the two methods we commonly use but there are other methods used in Image Based Rendering also.

Let us look at mosaicing for an example. If you have these two images you can almost see for yourself that there are images taken at the same location but at two different viewpoints while the camera has moved itself or it has rotated by itself. There are certain sections on the left hand side of the picture here which is absent and there are of course certain parts on the right hand side picture which are absent on the left hand side. But you can see that there is a small section here that is common. The right part of this image on the left image is common to the left part of the right image here.

As you can see the front end of the bus with its right window here and the corresponding temple like structure here is common. So you can use them to match and simulate and create a larger panoramic mosaic image like this.

(Refer Slide Time: 50:36)

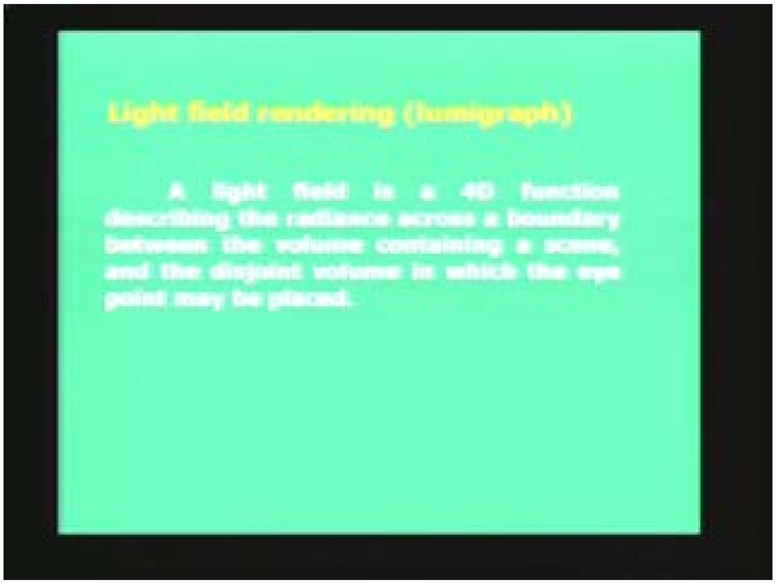


That is possible as you can see, there will be some parts of this image which will not be available. But as you can see now it shows that not only the camera has moved and shot the right frame from left to right as it moved but it also rotated and turned by itself. We have a shift in the pan and the azimuth angle as well. It has it has probably gone through a pan azimuth scroll and what has been done is you find out the part of the image which is appearing as extra on the right hand side of the left picture and merge it with that. So you need to match some parts of this picture and create it. So this is an example of two frames to create a larger view.

People have experimented software toolkits available where you can take snapshots around you at full 360 degrees rotation now. You can take snapshot about say 15 degrees or 30 degree intervals provided there is of course a little bit of overlap between two successive frames. As you say if you are moving to right the right part of the first frame should have some common part with the left portion of the second frame and so on because you need to match and find out where you have moved and how much you have moved. There are many good matching techniques used in image based rendering or mosaicing in this particular case to find out the alignment. It is called a concept of image registration.

Use concept of image registration to find out the parts of the pictures, match with certain parts there, then align it, and then create the larger mosaic. And you can actually create a full 360 degree mosaic around you either in an outdoor or an indoor of a scene but of course it takes lot of computational time. That is mosaicing for you. In addition, there are two top images on mosaic together to form a panoramic view of the scene. Lumigraph we talk of a light field that is a 4D function, which I described sometime back. It describes the radiation across a boundary between the volume containing a scene and the disjoint volume in which the eye point may be placed.

(Refer Slide Time: 52:25)



So, we come to the last concluding slide of this lecture that talks about visual realism. Why are we discussing about all these advanced concepts in computer graphics?

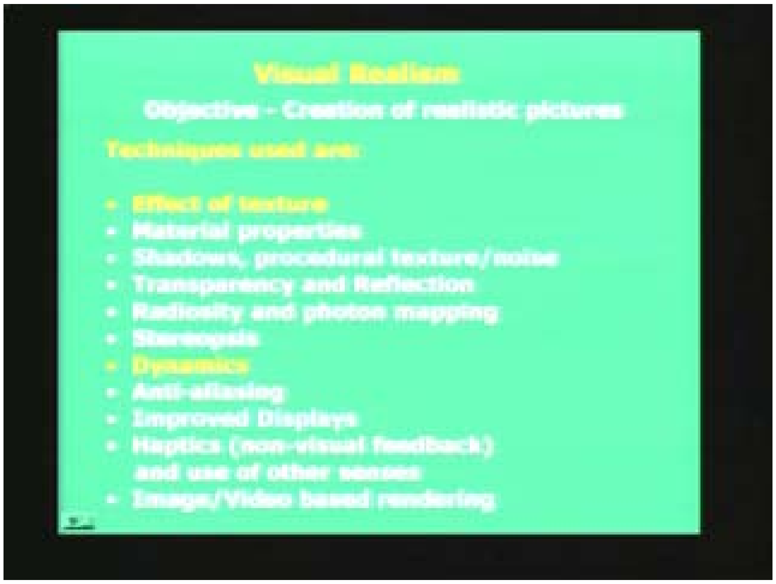
Well, you can have a full-fledged course on advances on computer graphics. In that we just introduce some of these materials and there can be extensive discussions on each of these topics. In addition, you would have not discussed some topics like procedural mapping, procedural noise creation, purling noise models, more on soft modeling, video image and video-based rendering, creating shadows, reflections and all that. So let us list some of these topics which comes under the falls of advances in computer graphics or virtual reality and why all these are necessary? It is because you need to create visual realism.

Why do you need visual realism? Where you need to create pictures, we discussed about this beginning of the course somewhere where picture should be made almost real life or real life like as close as possible so that the better algorithm designer you are somebody should not be able to distinguish between the real life picture and a computer graphics simulated picture. You need to bring in concepts of advances in computer graphics, virtual reality to simulate or to emulate visual realism in your pictures and that is the objective of visual realism.

The objective is to creation of realistic pictures. We will just go through a list of few concepts that are used, techniques which are used to create visual realism is a effect of texture. We have seen a small example today. We talk about material properties on object surfaces. There are concepts of using shadows, procedural texture and noise. You need to create transparency in objects because not all objects are opaque, reflection on object parts, a shiny surface is necessary. It probably goes with material properties and object properties. The reflection goes with material surface properties and transparency will go with object material properties inside an object. Also, we need concepts such as radiosity and photo mapping, stereopsis to simulate the view of the human beings or animals where we use to eyes to see a view and observe depth rather than single eye.

Dynamics for animation, real life dynamics, antialiasing we covered today, improved displays methods by which improve displays not only CRT monitors but improve projection screens and projectors are necessary.

(Refer Slide Time: 55:03)



Haptics non-visual feedback and use of other senses where not only visual feedback but you need to transmit a pressure. Well, you can use audio and video together to make it more of real life but in some cases for training use haptic devices were people can interact with the system in like a physical world. Suppose you are playing a game, tennis and when you hit a ball by moving your hand, well this hand is covered by gloves which will give the effect of the impact of the ball. And you could be playing with a computer audio and your partner could be also playing on the other side of of the computer and both will have the haptic devices on.

Haptic devices in virtual reality are in great demand. There is lot of research which is going on in this field and people use this for training pilots or training in the army also in defense for various civil and commercial applications haptic devices are used. We also discussed a little about image and video based animation. So, a combination of few of these concepts or all of these will definitely help to make your picture appear as real as possible and that is the main goal of visual realism in computer graphics or virtual reality where you want to create as real life simulation pictures as possible. So, with that we come to the end of the lectures on advanced concepts in computer graphics, thank you very much.