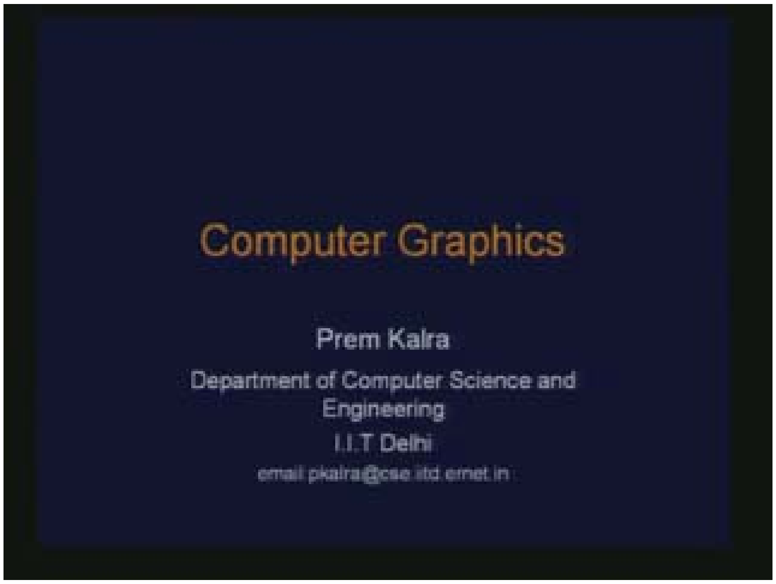
Introduction to Computer Graphics
Dr. Prem Kalra
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
Lecture - 1
Introduction

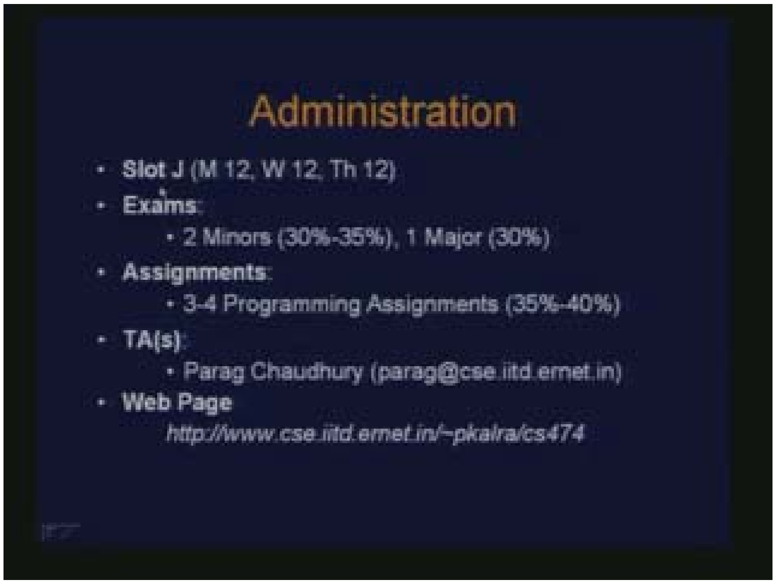
This is an introductory course on Computer Graphics where we will deal with some basic and fundamental principles of Computer Graphics.

(Refer Slide Time 1:04)



My name is Prem Kalra. The best way to contact me is through the email address which is there and you can contact me from anywhere. So let us try to see the administration of this course.

(Refer Slide Time 1:40)



As we know this is in slot J: Monday Wednesday Thursday at 12. As far as the evaluations are concerned it is going to be based on exams where we have two minor exams between 30 to 35 percent of the grade and major exam which may be of 35 percent grade and then we have projects or programming assignments 3 to 4 you will have 35 to 40 percent of the weightage.

(Refer Slide Time 3:04)

The books and the material that is going to be used are as follows:

Most of these are sort of reference materials so I will be referring to collection of these books and material. This book by Foley Van Dam and others and the Computer Graphics book by Hearn and Baker are in international edition. These are other books from where we will pick some materials.

(Refer Slide Time 4:00)

Overview of the contents of this course: first of all we are going to look at some introduction and preliminaries of Computer Graphics, Why one should study Computer Graphics, what are the applications of Computer Graphics then we will talk about what we referred to as raster graphics.

Raster graphics basically is required because when you want to look at the most of the displayed devices these days they are nothing but a collection of raster points which are pixels. So, one needs to have the display in these devices. So what does it take to have the display of such devices involves certain techniques to be followed which we studied in raster graphics. Then there is concept of clipping. So clipping is basically something which is to identify what is of interest and what is contained in a zone which is of interest for display.

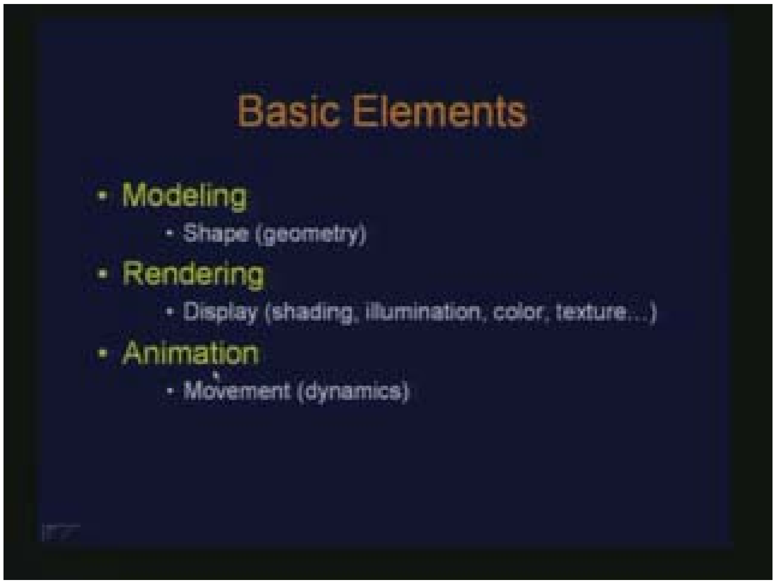
So you clip entities or primitives according to the reason of interest you have for the display. Then we are going to look at transformations. You must have seen transformations like rotations, scaling etc are known transformations. So how these transformations are used in the context of Computer Graphics and these are going to be both 2D and 3D transformations. Then we are going to talk about some of the representation schemes and the way to do modeling.

Thus, curves and surfaces are basically methods or representations which give us the indication of how can we model objects. Then we have rendering which is how do we display the models and then possible looking at the movement of those models and that is what gets captured in animation. So this is the broad content we have. By looking at many examples of Computer Graphics you would understand what it could be possibly but if you just want to see as a definition then it could be a basically a way to use computers to define, to store, to manipulate, to interrogate and present or display pictorial output.

(Refer Slide Time 6:54)

These are some of the essential functionalities we have which we addressed in Computer Graphics and the motivation to have a pictorial output is as we all know a picture is worth a thousand words. If you want to narrate a story or a scene or a situation then if you give a picture you do not have to write so many things about it. And the scope of applications is in all different disciplines we have. We have application in industry, we have application in art and entertainment, and we also have application in education, medicine and so on. Therefore the scope is very wide.

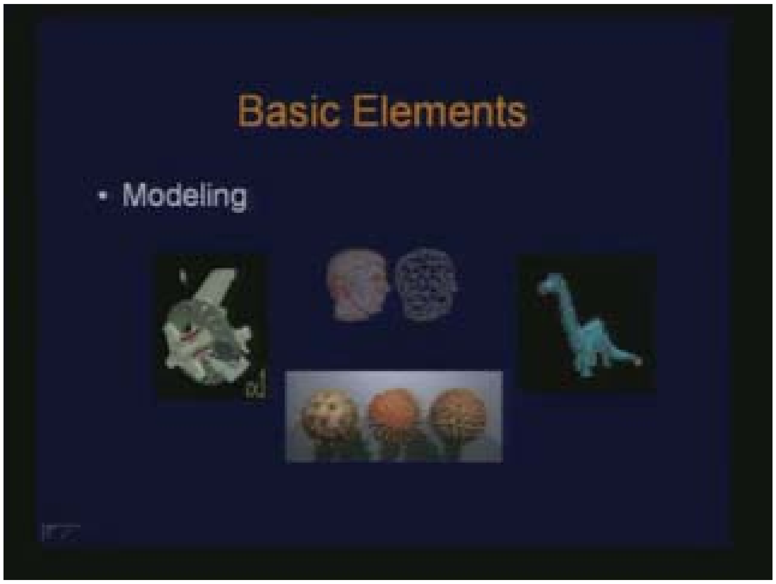
(Refer Slide Time 8:19)



So, if you look at the basic elements we had, the way we defined the contents of the course may be we can again look at the broad three elements in Computer Graphics. What are these? These are modeling where we try to define the shape of the object and what do I mean by shape is basically the geometry of the object or the scene. Now, what are the different representations we require in which we can have the shape or the geometry of an object and that is what modeling is all about. Then we are looking at rendering. Rendering is a way to display objects which in turn include not only deciding what is to display but also the attributes of display.

For instance, we are going to talk about shading of an object, illumination on the object, what is the color of the object or the texture of the object so all these attributes also have to be incorporated when we are talking about rendering because one of the motivation we have is to be able to do the rendering as realistic as possible. And then the third element or the component is animation where we are addressing the issues of movement and dynamics. This could be movement and dynamics of the objects of the same; this could be movement or dynamics of other parts of the scene for instance camera movement that also gives a notion of movement of the scene. Or may be change of the attributes in the rendering will also give you some notion of the animation. It is basically evolution of time of the scene. These are some examples of modeling for instance.

(Refer Slide Time 10:43)



So we have some examples, this is some engineering part where we have collection of primitives there is a disk, there is some cylinder and there are other simple primitives. So it is an assembly of primitives together and then there are operations which could have union of them, subtraction of them and so on. This is one way of dealing with modeling, the assembling of primitives and operations of these primitives.

Now there could be situations where we want to have models of structures which are not so regular as we find in engineering applications for instance a face, a fairly a regular structure. Then the question comes as how do we represent? Therefore one way to have a representation could be have a collection of points and the connectivity of these points which we call it as mesh so it is through mesh and this is showing that you can have a notion of sampling of these points from the resolution in which you specify the mesh.

Here you see a dense mesh and here you see a coarse mesh and that governs the faithful reconstruction which you have for the face. So there could be instances where this is better suited and there could be instances where this is better suited. Then there are also structures where we need to have a continuous representation of the geometry. Here when you see a structure like this what we have is the continuous surface more or less. So there are ways in which we can have definition of surfaces which would adhere to the continuity criteria we prescribe.

For instance, the parametric surfaces namely the b spline surfaces fall into this category. And there are structures which we have of this kind; here what we have is there is something like spherical in nature and then there are some artifacts to that structure. So we can also look at this as a definition of a function defined on a spherical domain. So we have implicit representation of a surface through these functions, these are called implicit surfaces. There is a range in which we can have a representation for the geometry of object. And here we are concerned about the envelope of the object. We are not talking about the inside volumetric representation, this is just the envelope.

(Refer Slide Time 14:35)

So this is an example where we have a rendering of the virtual scene [v….] and so on. Now this is just to illustrate the issues involved. Therefore within this scene we have defined a camera and this is the plane on which we project whatever is seen on this camera. So we are saying that there is an issue of viewing what we have in this screen through a camera which is defined for the seen.

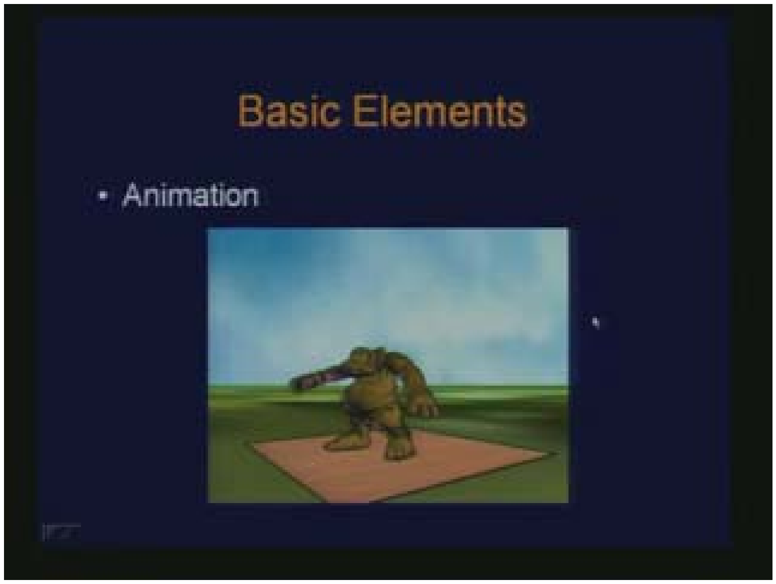
Therefore there are issues to what gets clipped within the custom of this view which is defined through the camera. And then there also issues as how do I render these objects, what is the shading I have, that is the interaction of the light source which I have in the scene and things like shadows are all the rendering issues we have. So here we see that whatever is there of the scene is with respect to this camera this sort of stimulates the process of rendering. And then you see on the walls we have this pattern which is some sort of texture to the wall, it is the texture mapping on the wall.

(Refer Slide Time 16:27)

Here is a favorite example: look at these three images, we always try to attempt when we are doing rendering to have a high degree of realism. Now if you look at these three images the question is which one of them is the real image? This one here is the real image but the other images are not too bad. If I had probably not shown you this you would not have made the difference. So, all it is saying is that the rendering using various attributes in aspects we have one can achieve the quality of rendering as close as to the realistic rendering we have.

Of course there are differences, there are light differences, there are texture differences etc so this is an inspiration that we have a real image here and I want to try to simulate this through Computer Graphics technology. This is an example of animation and here which what we are trying to show is that this cartoon shape what you see you see will do some movements.

(Refer Slide Time 18:00)



So here in fact there a several things which you might have observed that there was a motion of the camera, there was a motion of the character moving hands, arms and so on and all those things were put together. So when I need to do this animation of this character I need to have a representation in such a way that I can perform the animation.

Therefore when you look at the structure it is an articulated structure meaning that there are joints, there are segments which I can move and in turn I can move the envelope of this character which I see it has outside skin in association with my articulated structure which I have inside. So there are several ways in which we can do this.

(Refer Slide Time 19:37)

Historical perspective of Computer Graphics:

In 1963 we got the first graphics workstation. So it enabled to have the graphical output on the monitor. Then the first SIGGRAPH which is a very popular event in Computer Graphics community is a special interest group in graphics, it is an ACM conference which was held first in 1969 since then it has been held every year. And then if you look at the activities which went in 1970s they were primarily concerning raster graphics, shading illumination so rendering was quite emphasized. The trends continued even in late 70s and in early 80s. So the emphasize has been in the realism of the rendering. And then in particularly late 1980s there was this notion of physically based animation.

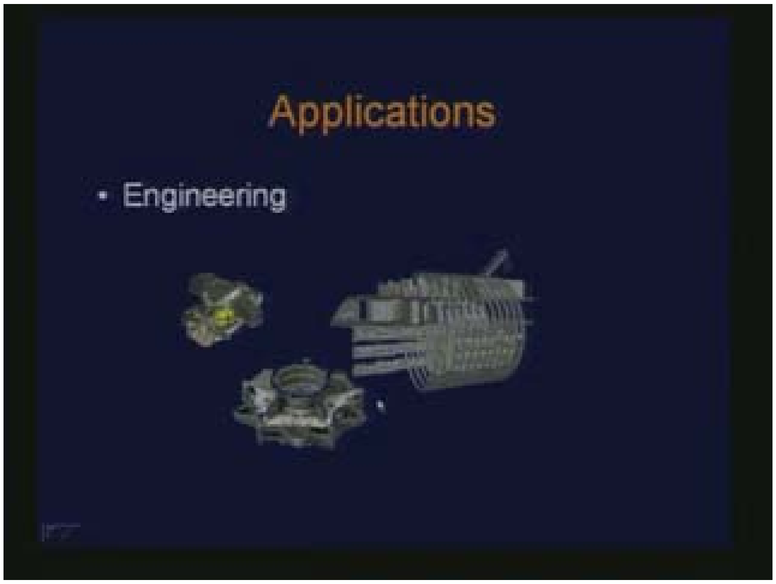
What it means is, basically how do I eject the laws of physics to the process of movement? There are several situations where I observe if physical law governing the movement. If I throw a ball it is a projectile motion and if know how to do projectile motion. So there are many situations where we can incorporate laws of physics and do the animation that was the motivation.

In 1989 what we had is the first academy award given to a completely computer generated animation film which was tin toy by Pixar. Pixar is an animation company which does the computer generated animation films. And in 1990s the emphasis was given to the interaction, scientific visualization, virtual reality, augmented reality, multimedia and so on. Therefore here in fact what we you see is the broadening of Computer Graphics many aspects started coming in. By the nature of Computer Graphics it is interdisciplinary in nature but this was more sort of emphasized in 1990s. And in 2000 onwards we have real time visualization of large data sets. So what had happened is the concept of modeling which was in the form of primitives and the representations for surfaces and so on in fact started depending largely on the acquisition methods of data and geometry. So they just put laser scans for acquiring the data, what it gave in turn is huge amount of data.

Then the major issue was on how to cope up with this data. Now we have the way or we have the data which can capture the geometry but the data was so dense then you have to have processes by which you can prune out the data and use it in a meaningful manner and these relate to issues like data compression and also some combination of computer vision and Computer Graphics.

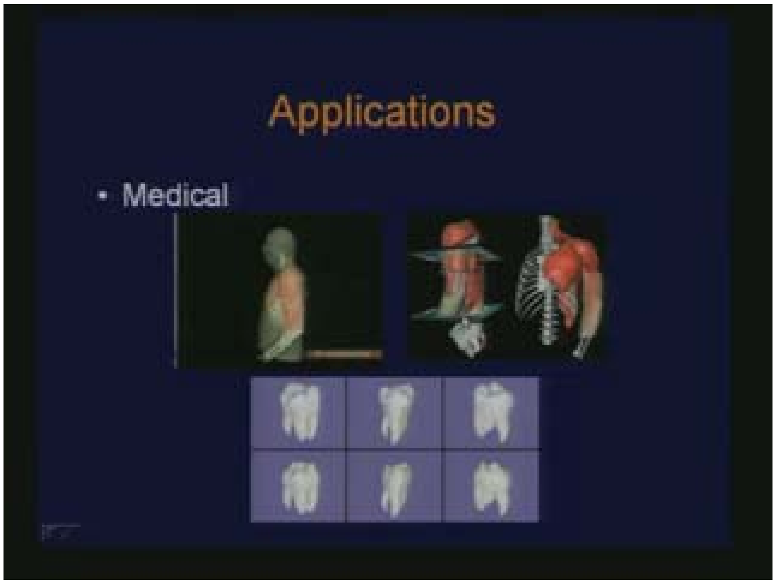
So, if you look at the process of Computer Graphics what do we have? We have input coming as a specification of the scene where we have the geometry of the models and so on, output which is on a device is an image whereas computer vision classically is seen as I have an image or may be more than a single image I have multiple images and I try to do the reconstruction of the scene or the model through those images. It is some what an inverse problem what we do in Computer Graphics. So there could be several techniques which one can use from vision and from graphics and do some meaningful thing. That is how the activity started in this area. Here are certain applications which are specific to studying engineering so here are various engineering parts.

(Refer Slide Time 25:25)



Now these are helpful in a way that you can do lots of simulations like instead of doing these experiments live with the actual parts you can do simulations using these virtual parts. So that is the way we can have the application of Computer Graphics in general.

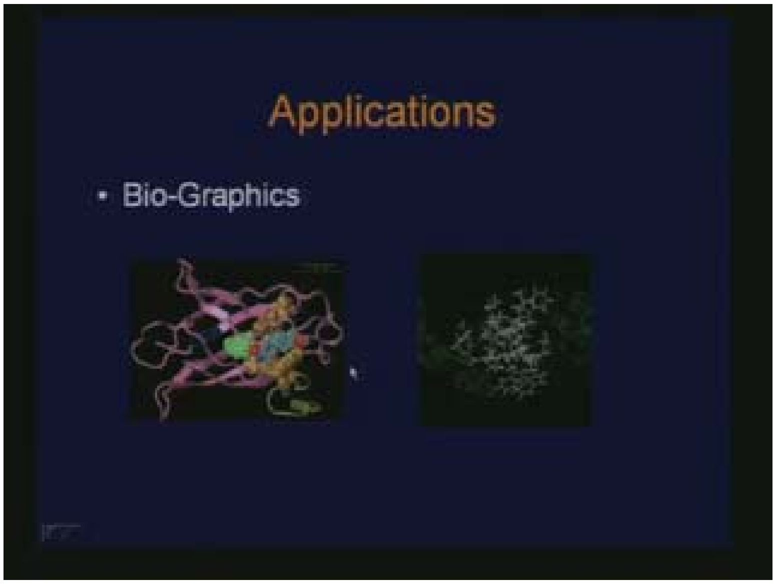
(Refer Slide Time 25:54)



Similarly we also have applications in medical things. Here it is shown that what we can use in Computer Graphics as building tools which allow you the visualization of various parts or organs in a human body. So there is a project called visible human project where you have enormous amount of data in digital form, in slices and you can use that data to do 3D reconstruction of parts.

These are the slices and this is the 3D reconstruction from this part and what is sort of shown here is you have this skeleton and some muscles which are on the top of that skeleton and this is the envelope which is like the skin of this arm. then you can add functional attributes to this in addition to the geometry which we have reconstructed and again perform the biomechanical simulations; as what happens if I move this skeleton, how does this muscle deform, how does the skin change and so on. So we can use Computer Graphics in the domain of medicine. This is a gross level of body structure we have and these are the minute structures we have for instance tooth and we can look at the reconstruction of those. Hence they can help us in the process of diagnosis of any abnormality, visualize them so this can be a good aid to the medical [cle…]. Similarly one can see applications in bio-graphics.

(Refer Slide Time 27:55)



Bio-graphics meaning we are dealing with molecules and molecular structures so we can have representations of various parts or atoms of a molecule and their connections or other functional aspects. For instance, if protein is represented, what are the linkages of protein to the rest of the structure and so on? Hence one can also apply graphics in biology.

(Refer Slide Time 28:40)

Entertainment: of course this is a sort of an obvious application and we all see application of Computer Graphics and entertainment. So you might have seen several of these special effects which are done through Computer Graphics. You might have not noticed that it was done through Computer Graphics but you must have seen those special effects. This example is from final fantasy and you can see the amount of realism which has gone in.

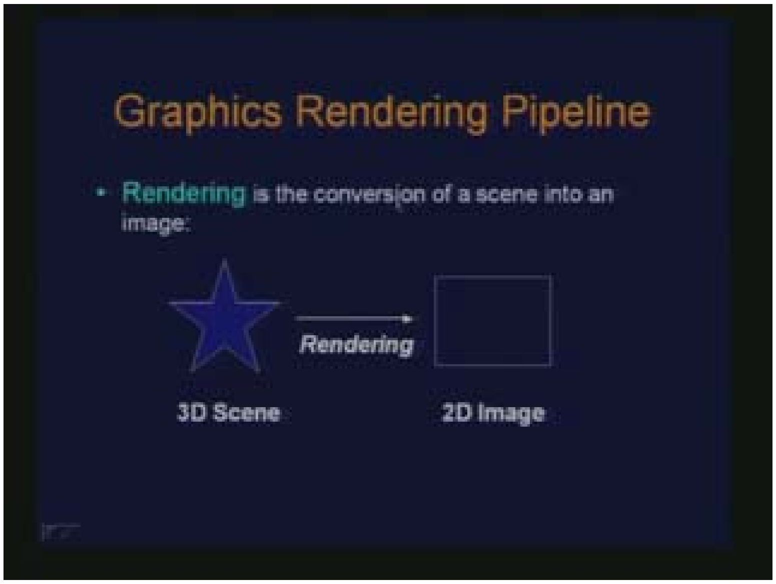
(Refer Slide Time 29:12)

(Refer Slide Time 29:28)

(Refer Slide Time 29:31)

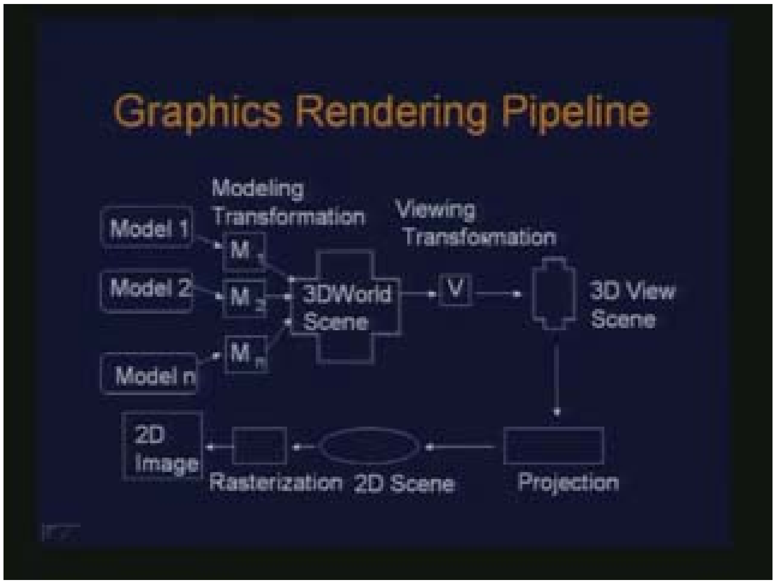
So here again the issues of the amount of data which is generated, the amount of resources which are used up for generating the frames is enormous. So this is a completely virtually created character. We still notice that it is not a real person but we can appreciate the realism which it has even now.

(Refer Slide Time 30:19)



Now, if I see a very gross picture of what we are talking about Computer Graphics then basically I can look at this as rendering process and I can have the definition of a 3D scene which in turn gives me a 2D image through this rendering process. So all I see is that the conversion of a scene into an image. Therefore let us try to look at the anatomy of this process in much more elaborator way and see this process as a pipeline.

(Refer Slide Time 30:57)



We have just elaborated that conversion we had from 3D scene to 2D image. I have a 3D scene which in fact is constructed using various models so I have model 1, 2 to model n.

And I apply some transformations through which I take model one to this 3D [w…]31:37 so an example could be let us say I want to do the reconstruction of this room.

There are several chairs, there are people and there are walls and so on and many things we have. Now what I can think of is that since chair is a single object then I do not have to do the modeling of every chair if I find that the other chairs are just an instance of one chair. I have a model of one chair and all I need to do is apply an appropriate transformation to that model and put it in the 3D world.

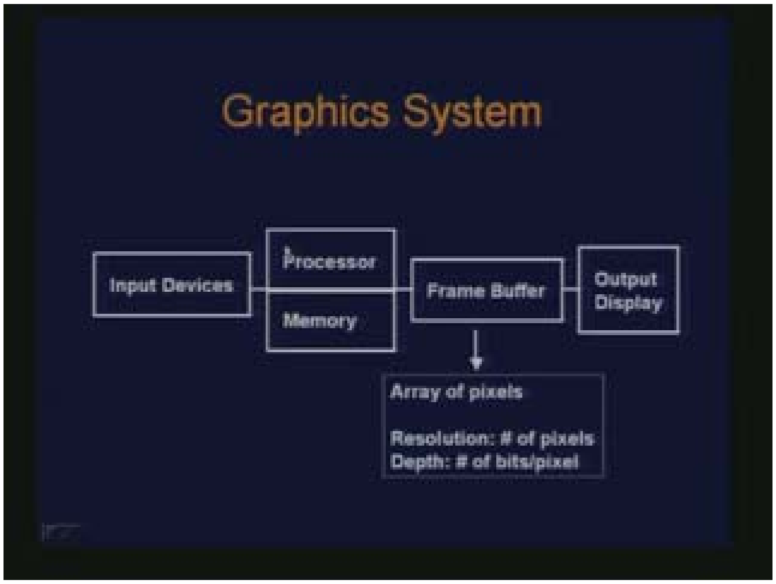
So this modeling transformation is basically taken from the coordinate system that I have specified for model to the coordinate system I have for the 3D world. So, once I take a chair to the scene it does the transformation of these coordinates and through that it enables me to specify instances of the same model. I can have various modeling transformations attached to this model and can put it in the scene. Therefore once I have got this 3D world built up as an assembling of various models using modeling transformations now I need to worry about how I view them. If there is a situation where I am setting and that is where I put the camera.

Once I have built the scene by inserting the camera again gives me specification of how do I want to see that scene? So again there is a notion of transformation which is for the viewing of the scene because this scene going to be look differently if I see it from this point, from that point and from other points. So that is what we are trying to capture through viewing transformation. Viewing transformation in turn means that I specify this camera and the attributes of that camera into the scene. So this transformation gives me another 3D scene which has the viewing specification. Now remember that the output is going to be an image.

Therefore all these processes ultimately have to be mapped to be something in two dimensions. So there I need some mechanism by which I can project this 3D scene on to something which is two dimensional. And since I have specified the viewing or the camera I have a way to perform this projection.

Here we are taking about how I project this 3D scene to get this 2D scene. Once I have got this 2D scene it could be in the representation I used to define the world which could be a continuous representation of the object. Ultimately the 2D image which we display is displayed on a raster device which is nothing but a collection of pixels. Then I need to have a process of rasterization where I will convert this representation whatever I obtained in the 2D scene to this 2D image. This is how the whole pipeline is for converting my 3D scene to this 2D image.

(Refer Slide Time 36:45)



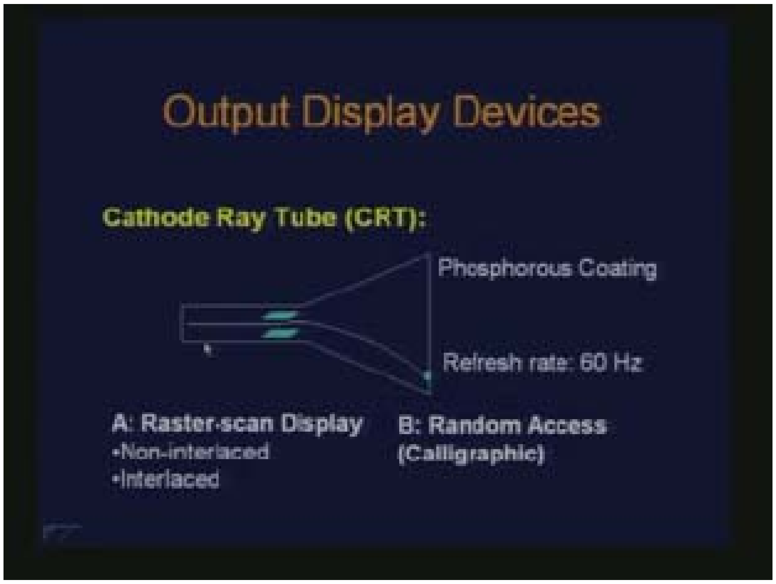
Now, if I want to see graphics more in terms as a system where I had the hardware which are used for the system I can see this as basically a collection of some input devices for instance keyboard, mouse or a unit which has the capacity of processing that is some memory unit which does this.

Then there is a frame buffer which is nothing but as far as the representation is concerned it is something like array of pixels and the motivation of having this is that the output display is something like a CRT monitor. But these days we also have TFT, LCDs etc.

Now the idea here is that this processing I do here needs to be map to this output display through this frame buffer which is containing the information in something like raster form, grid of points, grid of pixels which I will map to the output display whatever I have. So this frame buffer which we have is nothing but something like a collection of or a grid of pixels where the spatial resolution gets determined by the number of pixels I want to display or I have and the depth is the number of bits I assign for each pixel. This basically gives me the capacity or the resolution of this frame buffer or the memory space I need.

Now let us try to see the linkage which we want to establish between this frame buffer in the output display.

(Refer Slide Time 39:18)

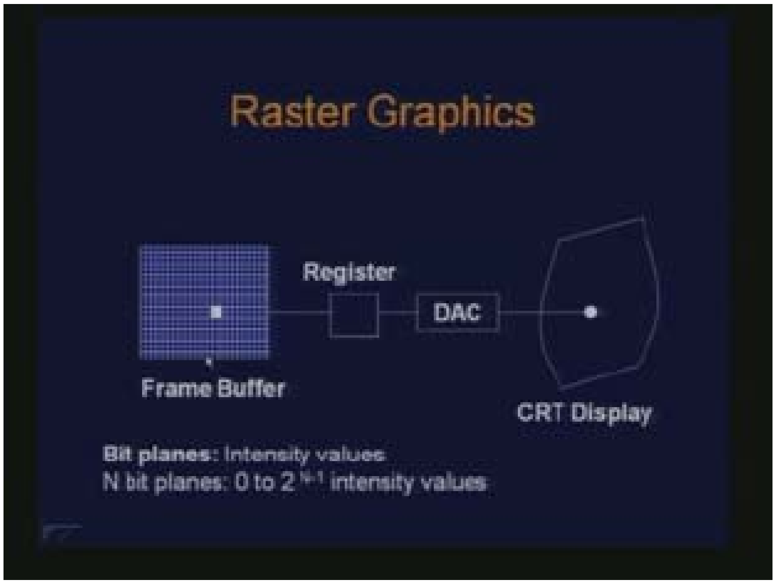


So you are familiar with CRT displays cathode ray tube displays and its working. What is happening is that these are some sort of deflection rates we have and there is this phosphorous coating on this screen which basically gets on and off with a refresh rate of 60Hz. That is why we cannot see the discontinuity; we are not able to visually perceive it with this rate. [So what can actually look at this display devices into the category of raster scan display and the random access display]. Raster scan is that I am basically doing a scan wise display right from the top to the bottom and random access is that I do a display in a calligraphic fashion so wherever it comes I just push that on.

Again in the case of a raster scan display I have what we call as non-interlaced and interlaced. So non interlaced means I go every scan one scan after the other one and in interlaced I basically do a half scan first and then the other half and this half is decided by, first I do odd scans and then I do even scans. Therefore this may increase the effective rate of refresh and that is what is done in many of the monitors we have, it is interlaced.

The resolution is not really affected because the time we are taking to do the display is very short. But the visual difference, as far as this is concerned is not so much because I am going to see just one scan and then the other scan. So it is a result of two scans eventually which gets to my impression and that is what gets aggregated when I perceive it.

(Refer Slide Time 42:30)

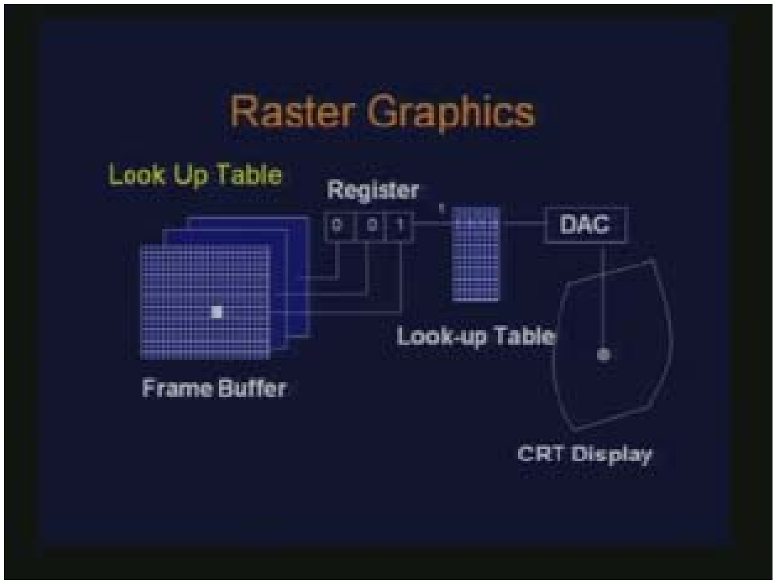


So now what we are looking at is some sort of an interface which takes us from the frame buffer which is grid of pixels or the array of pixels and the CRT display. This is an analog kind of a device. And now we are going to take some value of the point here which is the value for the pixel which we want to display and map it to this. now one can think of that the information we have here there is a register for the same number of bits I specify here for a point or a pixel and I have a digital analog converter which takes this value of the register and puts into the appropriate analog value I need for the change of the voltage or whatever.

Now what we are saying is that if I have the bit planes, bit planes are the number of bits I specify per pixel in this frame buffer. So the bit planes in turn give me the range of the intensity value I can have. So if I have n pit planes it means I can have these many intensity values 0 to 2 power N minus 1. Therefore what is happening is the range of my intensity of the color I want to have in my CRT device gets completely determined by the number of bits I assign per pixel in the frame buffer.

Now if I want to augment this, that means though I had a 3-bit value here I had a 3-bit value here so I had a 3-bit register here which should have given me 8 values but instead of 8 I want to have 60 as the range of the intensity something which I can have as a 4-bit pixel value. So how do I augment that? One way is that you can use the look up tables which is an ideal way to handle this. What happens is that, this is the scenario, I have this frame buffer where I take each of these bits and I have a register here which is a 3-bit register.

(Refer Slide Time 45:32)



Hence depending on the value of this register which earlier was getting mapped to the intensity value now I do something different. Therefore I define a lookup table for which I have clearly the width of this larger than the number of bits I have for the register and that is where I will do the augmentation. So I use this value in this register just as an index to the table. So I do not use this as an intensity value but I use this as index to the table. And in that table I have the entry of the width of my choice which is larger to the size of the frame buffer. So this way I am augmenting the intensity. The only limitation is, simultaneously if I want to look at I can still have those many colors, I can have only eight colors but the width of them or the values can range from 0 to 50 if I have a four bit width of a lookup table. But that may not be a big constraint. You may still want to use limited number as far as the colors are concerned but what you want is the width to be more or the number of bits per pixel to be more.