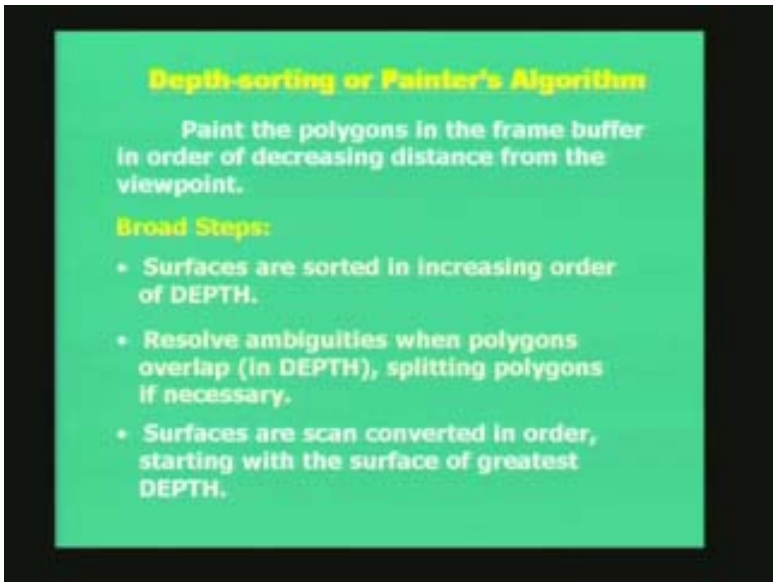
Computer Graphics
Prof. Sukhendu Das
Dept. of Computer Science and Engineering
Indian Institute of Technology, Madras
Lecture #29
Visible Surface Detection (Contd…)

So today we will discuss another Visible Surface Detection algorithm. In the last couple of classes we have discussed the concepts of back face culling and then of course the depth buffer or Z buffer based method for Hidden Surface Elimination or Visible Surface Detection VSD. Then of course we discussed the scanline based method which was an extension of these scanline polyfill algorithm in 2D.

Today we move on to other different type of an algorithm which is conceptual and you will find it to be very interesting and nice which is called the depth sorting or in other words Painter's algorithm. We will see what is the depth sorting or Painter's algorithm. So if we look back into the slide here we are talking of a concept where the polygons are painted in the frame buffer in order of decreasing distance from the viewpoint so that is the key idea or the principle of depth sorting or Painter's algorithm.

Remember, VSD algorithms are basically nothing but a depth sorting algorithm. It is a sorting algorithm where the key value or the number used or the parameter used for sorting is the depth value in 3D. And in this case the term itself is depth sorting but Painter's algorithm is probably a more appropriate one which we will see here. We need to paint the polygons in the frame buffer in order of decreasing distance from the viewpoint. And the broad steps in the depth sorting or Painter's algorithm are that the surfaces are sorted in increasing order of depth and that is the key broad step and to implement that you have to resolve certain ambiguities.

(Refer Slide Time 04:39 min)



When polygons overlap in depth and if necessary splitting the polygons so that is the second one. Then after this is done the surfaces are scan converted in order starting with the surface of greatest depth. So, if you imagine a three dimensional scenario where there are very large number of polygons say n polygons seen in a few tens of them to few hundreds, few thousands or even a few million polygons in a very complex scenario what you have to do is basically order them in increasing depth so that is what you have to do.

It is not easy to order polygons in order of web because of course if the viewing direction is from left to right then you could have one polygon here and other polygon there and then you can order. But of course if the polygons are overlapping the scenario like this a polygon one on a polygon two then we have the problem of ambiguity as to which one you should take the first one to solve them by some criteria and then once it is ordered as you see in the slide they have to be scan converted starting with the surface of greatest depth. The question which should automatically come into your mind is why it is also called the Painter's algorithm. It is also called the Painter's algorithm because if you visualize the scenario by which a painter draws a three dimensional scene in a canvas, drawing canvas the artist or the painter draws a three dimensional scene.

How will you start because let us imaging a scenario where he is standing on the bank of river or pond and at a distance he has not only water beyond that but he has the mountain and the sky with the sun rising and in the middle there could be some trees and plants and birds flying. So, if you imagine such a scenario in a beautiful spot or tourist spot or place for tourist to visit then what will the artist do? He will start by painting the background of the scene first because those are the objects which are at the farthest distance with respect to the artist of the viewer in this case.

So in real scenario you will probably start painting the background sky with the mountains and trees at a very far distance then of course you could probably paint the water body lying in between maybe a few birds flying or floating on the water and then he will draw some nearby structures which could be a small garden with some noise, flower plants and tea pots, shrubs and things like that.

So we will start from the objects in a sequence and starting from the objects which are at the farthest distance and put them in the background of the scene and then draw the ones which are nearer and so on till at the end he will overlap the scene with objects which are very close to you. Of course the entire view is imagined by the painter that he is viewing through a window which is our imagination of a two dimensional viewport then probably he will also draw the window in the front. So, that is the last one which he will draw because that is the nearest object in. So the painter's concept is to draw distant object first and then the closest object at the last. That is the sequence we also follow in the depth sorting or the Painter's algorithm and that is the term.

I hope the concept is clearer now that why it is called Painter's algorithm.
So coming back to the broad steps we will read it again. We have to sort the surfaces in increasing order of depth that is the number one broad step which you will see, find details of these and then when you try to sort surfaces in increasing order of depth I was just mentioning some time before that if you have surfaces which are well separated apart in depth there is

absolutely no problem in ordering them first. Of course you have to draw them later on but ordering is not a problem if the surfaces are well separated out. Now what will happen is assuming that you have the viewing direction like this going from your right from left or it could be from left to right whatever the case may be and you have the object number one in front of you and the object number two in front of you, then if they are well separated out it is of no problem but assuming you are looking into the ZX or ZY direction, this is Z let us say and this is Y or X and along the depth if you see when they are well separated out there is no problem. But if there is an overlap something like this let us say if it is an overlap then I cannot say that one object is in front of the other one, which one is in front is a case which has to be solved by different criteria then just looking into depth of certain points on the surface.

So you resolve this ambiguity as seen in the slide when polygons overlap in depth. That is what I was trying to illustrate with my hands as to what you mean by polygons overlapping depth and sometimes you may need to split polygons. Not all the time but sometimes we will see that and this was not an easy matter to handle in the case of the previous algorithm based on scanline. So we will see what Painter's algorithm does. And after this is done you may land up with more polygons because of splitting, surfaces are then scan converted in order starting with the surface of the greatest depth. Once the ordering is done you have to just scan convert in the descending order.

After the sorting is over let us say the first element of course if it is already sorted in terms of descending order with the first element having the largest depth so on so just scan convert in the least form left to right sorting in from the surface with the largest depth and the surface with the least depth used to at the end.

Let us look at this principle of depth overlaps.

We are talking of two surfaces here and say S1 and S2 are the two surfaces and the limits of the depth are Z_{max1} and Z_{min1} as given here for the surface S1 and for the surface S2 similarly you have the limits of depth such as Z_{max} to a Z_{min2}. And the principle here is the talking of each layer of paint which is nothing but polygon surface of an object because why do you call it a layer of paint? You should always try to imagine you as a painter trying to draw these surfaces with the greatest depth to be drawn first because that is farthest away from you so you paint that and then the one which is closer to you and so on. So I tell you the one which is closest to you.

Of course if there are only just two surfaces as shown here on the slide then of course you are talking of just two surfaces and the order of them and you scan convert one of them and the one which is closer to you. You start with one which is the farthest away. So the paint and surface, paints and polygons will probably be used interchangeably. Of course some points we only use polygons but keep this in mind when you are rendering polygons in some order we are putting as if some paints on the canvas to simulate a three dimensional scenario.

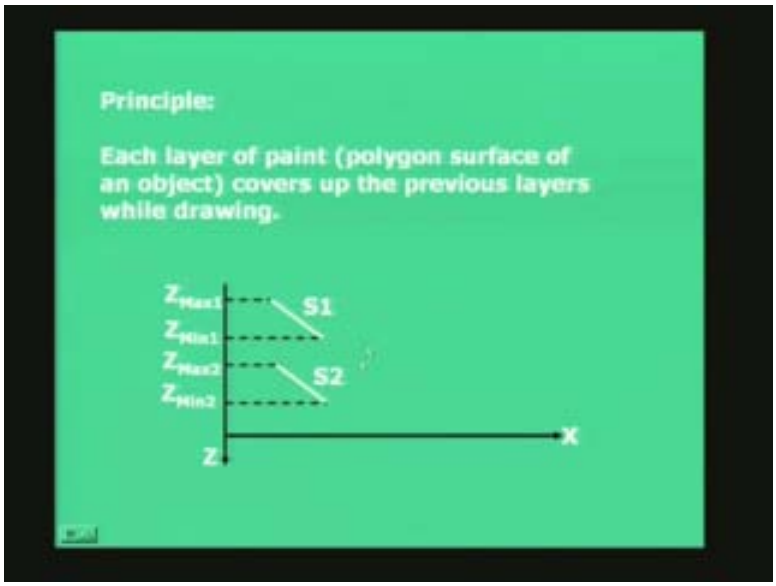
So coming back to this point where we talk of each layer of paint which is nothing but a polygon surface of an object it covers of the previous layers while drawing. That means if you draw S1 and then S2 assuming the viewing direction to be negative Z, remember positive Z this is the

scenario we are looking into the ZX plane Y axis coming out of you and getting inside the screen and we are looking into negative Z direction.

If that is so then you have the surface S1 to be the largest distance from you and the surface S2 to be closest towards you, you means the viewer and in this case you have to imagine you are sitting at the origin or you are sitting on the positive Z axis looking towards origin or sitting in the origin looking towards negative Z axis that is what you have to visualize.

And when you visualize that you will find a surface S2 is closer to you than surfaces S1. And when you order them in descending order starting from the surface of the greatest depth the list will contain two surfaces S1, S2.

(Refer Slide Time 12:52 min)



So in this case it is very clear that the list of ordering will be S1, S2. So the rendering order will be you render S1 first and then S2. If you see here that means the surface S1 will be drawn first and then the surface S2 because surface S2 will cover up the previous layer which is in S1 so that is fine. And how do you ensure here, what is the criteria you put ensure that surface S1is at a greater depth than with respect to surface S2 because you need a criteria based on which you will order the surfaces. You just check whether the Z_{min1} is less than $Z_{max}2$. Remember now, we are looking towards negative Z so absolute values of Z_{min1} will be more than Z_{max2}. But remember the concepts of depth sorting whether looking at negative or positive Z the greater or less than criteria could alter although the numerical value of Z_{min1} will be more than the numerical value of Z_{max2}. But since it is on the negative Z axis we will say that the Z_{min1} is less than Z_{max2}.

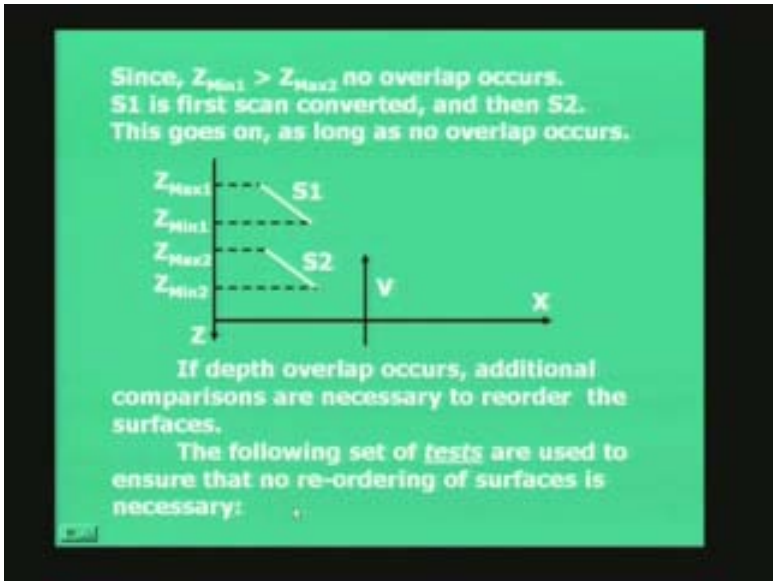
So you compare the largest depth range between S1 and rather max for S2 and minimum for S1 is what you compare and you check whether the min is less than max sort of a thing. If it is possible in these two cases we are talking of comparing between a pair of surfaces. And it holds good for n number of surfaces because you have to make that many comparisons between pairs before you do an ordering.

So in this case what I was talking was about of S1 and S2. So I am looking into Z_{min1} if it is less than Z_{max1} then I will say I am guaranteed to have as the figure says that the surface S1is behind S2 and that S1 has to be rendered first. So that is the first criteria which is used to order depth and I will say we have a sequence of tests which these surfaces are to be passed otherwise if this condition is violated and S1 and S2 are not well separated apart as the picture shows here then further tests will be done. I will say that this is test number 0 although I will not explicitly mark it, we will call this depth overlap. There is no depth overlap in this particular case I will call it the depth overlap of the first test of the 0th test in this case. So this is the case, very simple but if this does not happen that means S1 and S2 overlap and Z_{min1} is not less than Z_{max2} then you have to do something else.

So the first one is, although I did write Z_{min1} is a greater than Z_{max2}, well numerically you can apply a test if you want by looking into neglecting the sign but just taking the numerical value the view direction is negative Z so typically if you do not take absolute values but take real values Z_{min1} will be actually less than Z_{max2}. So you have to be careful with the sign depending upon the viewing direction. If this is the condition Z_{min1} is greater than Z_{max2} or less whatever the case may be no overlapping occurs S1 is first scan converted and then you to S2 so S2 will be overlapping or obscuring S1. And this goes on and on and no overlapping occurs between these surfaces and the rest of them. So you can carry on with n number of surfaces as long as there is no overlap.

If the depth overlap occurs that means if there is a depth overlap additional comparisons are necessary to reorder the surfaces. That is what you need to do, the test 0 fails which is the depth overlap test. We need to basically use more number of tests to reorder the surfaces if necessary.
And the Painters' algorithm recommends or suggests the following set of tests which are to be used ensure that no reordering of the surfaces is necessary.

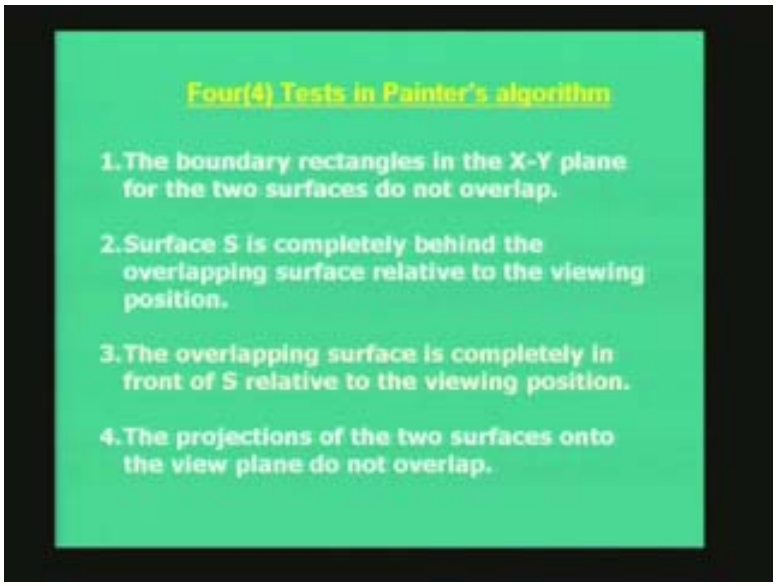
(Refer Slide Time 16:36 min)



So as I was saying that when you are comparing two surfaces S1 and S2 if there is no depth overlap let us say the view direction is like this and S1 and S2 is in this form, S1 and S2 then you can just scan convert them after the depth sorting is done. But if there is an overlap, there is a depth overlap the minimum max conditions which we just talked about is violated then we have to pass these surfaces through four more tests which we will see and those tests will tell you whether the resurfaces have to be reordered or just you can look at min max also and you can do scan control. So we will see that what are these four different tests one has to go through in scenario when S1 S2 do have a depth overlap.

We will go through list wise the set of four tests in Painter's algorithm and then we will take illustrations to demonstrate what these tests mean. The first of the four tests in the Painter's algorithm says that the boundary rectangles in the XY plane of the two surfaces do not overlap. This is easy to visualize but we will see with an example that the boundary rectangles in the XY plane of the two surfaces do not overlap that is the number one test. The number two test says that you remember, you are entering the number one test if the test number 0 which is depth sorting test fails itself, the depth test fails when entering one failing one we enter two and so on.

(Refer Slide Time 19:05 min)



The test two says that the surface S which is under test is completely behind the overlapping surface relating to the viewing direction. We will see the surface S which has to be rendered first whether it is completely behind the overlapping surface. We will see what this behind test is all about and how it is done. If this test also fails we move on test number three where we will say that the overlapping surface is completely in front of S relative to the viewing position.

What is the overlapping surface? The surface which is in front is the overlapping surface. And the surface which is in front is tested out whether it is completely in front of S related to the viewing direction. So this is test number three. If that also fails we move on to test number four which says that the projections of the two surfaces onto the view plane do not overlap. Now this

step number four appears to be similar to test number one. But there is a difference which we will illustrate but for the time being just understand that in the first test we are talking about bounding rectangles of the polygon. You remember, when we discussed about polygon scanline polyfill algorithm we discussed about minimum enclosure rectangle which has to be obtained first to know the limits of Y basically.

So in this case you get the limits X and Y of the two polygons S1 and S2 or the surfaces S1 and S2 and the first test checks whether the bounding rectangles are separated, that is number one. The fourth test of course you are entering when the test number one two and three all of them fail. When all of these fail then only you are entering test number four so we have entered test number four with the condition that the boundary rectangles are overlapping so the test one has failed, test two and three also has failed and you will understand that shortly, test number four test finds out whether actually the polygons overlap, the bounding rectangles could overlap. We will see with an example. But the polygons may not fail, that is test number one. Whereas test number four we actually find out whether the polygons physically one of them overlap the other when you take that projection on the view plane or projection plane and that is the difference between the test number one and four.

One is, bounding rectangles overlap; another is that the polygons themselves overlap. So coming back to the test we read out once again before taking up the illustrations to understand them completely. Of course one and four is easy for you to visualize, we have to take examples for two and three to understand and will also take up case studies in this class today. Test number one says that the bounding rectangles on the boundary rectangles in the XY plane for the two surfaces do not overlap. Number two, surface S is completely behind the overlapping surface relative to the viewing position. And the test number three says that the overlapping surface is completely in front of S relative to the viewing position and the test number four projections of the two surfaces onto the view plane do not overlap.

I hope you have copied these four tests which we will examine one after the other and of course I will display them once again when we discuss each one of them separately and we will take them right now. But again I repeat that you have depth sorted let us say based on Z_{min} and Z_{max}. e have tested test number zero which is thus the minimum maximum test for depth overlap and if that fails only then you move for test number one failing which successively go to test number two failing which go to test number three and so on.

If any one of these tests pass either zero, one, two, three or four so in the true sense you have five tests although there are four tests which have been talk about the zero test is simple depth min max check which is not considered in the sequence. So you get into the four tests only this zeroth test overlap I am using this word zeroth test but it is basically min max checking of the depth which you do and any one of these test passes out you just scan convert in the order which you have done based on the extent of the depths of the two polygons S1 S2. And if all the test fails starting from the first zero th one to one two three four if all of them fail it is only in that case you need to reorder the surfaces and draw not S1 S2 but S2 S1.

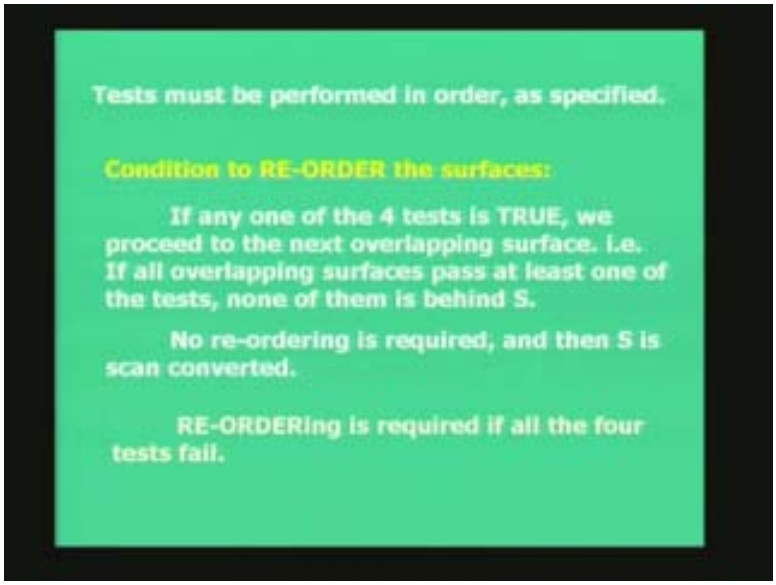
You only need to do that when all the test fail and you are coming out of the test four you have failed actually. You mean the surfaces have failed in this Painter's algorithm, the sequence of the

Painter's algorithm test. So we will move on to the illustrations where we must keep in mind the test must be performed in order as specified in the previous slide. You cannot reorder the tests as you are reordering the polygons.

So condition to reorder the surfaces which I just discussed right now that if any one of the four tests is true we proceed to the next overlapping surface that is if all overlapping surfaces pass at least one of the test none of them is behind S. So what we are talking about now is trying to generalize all the tests that talks about comparing two surfaces. You can visualize two or more there could be n number of surfaces in some order of depth and we are checking that after this ordering is done based on the Z X tens can we scan convert starting with the polygon or the paint with the maximum depth.

Now to do that you have to keep comparing each pairs in sequence and see that at least one of the tests are passed. Only in that case we can proceed with the next overlapping surface to test and scan convert the other works. What you mean is that if all the overlapping surfaces pass at least one of the tests none of them is behind S. So you are testing a surface S let us say which is in the background or the farthest distance with respect to all other which are in front. There is a surface S and you have surface one two three four and so on. And as you keep going you basically keep comparing all these surfaces with S. And if one of the tests pass in all of these cases when you are comparing pairs of surfaces then you are guaranteed, what you are guaranteed is that all the surfaces are in front of the background surface S and then S can be converted peacefully. Other surfaces have to be of course equivalently compared between each other. This is the condition we talk about to reorder the surfaces, this is important.

(Refer Slide Time 25:27 min)



Remember, you reorder only when all of them fail otherwise you keep the ordering as it is. No reordering is required in this case if at least one of the test is true and passed and then S is scan convert. Reordering is required; I repeat again, if all of the four tests fail this is very important.

So remember, the condition to reorder that no reordering is required and then S is scan converted if at least one of the four tests is true whereas reordering is required if all the four tests fail.

Now test number one, let us go to the checking of the tests here, what are the criteria which are used. The first one is very simple as read out earlier that the boundary rectangles in the XY plane for the two surfaces do not overlap. So, if you have depth overlap but no overlap in the X directions, this is a scenario as we are checking here this is the case where S1 S2 has a depth overlap. If you remember, in a few slides back this was the view direction, you had two surfaces S1 S2 there was no doubt of overlap you could have rendered them in that sequence S1 S2. But now these two surfaces are not separated anymore they overlap so it is lying like that.

And remember, this is the view direction and that is what is given in the figure. If you carefully watch that the minimum and maximum values of the Z limits of the surfaces S1 S2 have depth overlap. Now we check whether there is any overlap in the XY plane. I have taken the projection of this three dimensional scenario with two surfaces on the ZX plane but you have to check it similarly on the ZY plane also. Basically what you do is you project the surfaces now only on the X Y plane. So if you project the surface on X Y plane open up the XY plane and check the minimum enclosed rectangle. You check the minimum enclosed rectangle and check if the minimum enclosed rectangles overlap.

How do you test that? You remember scanline polygon filling algorithm, what is the simple test you check whether a point is inside the polygon in this case a rectangle or not, inside outside test, inside outside test if you remember since rectangle has only four vertices the minimum enclosed rectangle of a polygon S1 will have four vertices it is a rectangle minimum enclosed rectangle for a surface S2 also will have four vertices, 4 and 4 so what you have to do is take up each of these four vertices say in a sequence for surface S1 find out for the minimum enclosed rectangle for S1 and check it out if it is lies within the minimum enclosed rectangle of S2 and do the reverse. That means, take these points of four vertices of the minimum enclosed rectangle for S2 and check if it lies within the minimum enclosed rectangle for surface S1 so that is how to use it.

So as I said here I am looking into the ZX plane so as here I am taking the projection of the X Y plane also on the X axis and as you can see here that the X extends do not overlap. That means if I am taking a minimum enclosed rectangle and assume for the time being that the same thing happens in Y that Z extents overlap, the Z limits overlap but the limits along the X and as well as Y do not overlap. And if that is so it shows that the minimum enclosed rectangle for surface S1 does not overlap the minimum enclosed rectangle for surface S2 and vice versa. That means there is no point or part of the minimum enclosed rectangle, mind you, not the polygon, it is considering only the minimum enclosed rectangle of surface S2 is within the minimum enclosed rectangle of surface S1. And if that is so as the figure shows in along X axis there is no X extent overlap similarly Y extend you can also check whether Y extents overlap.

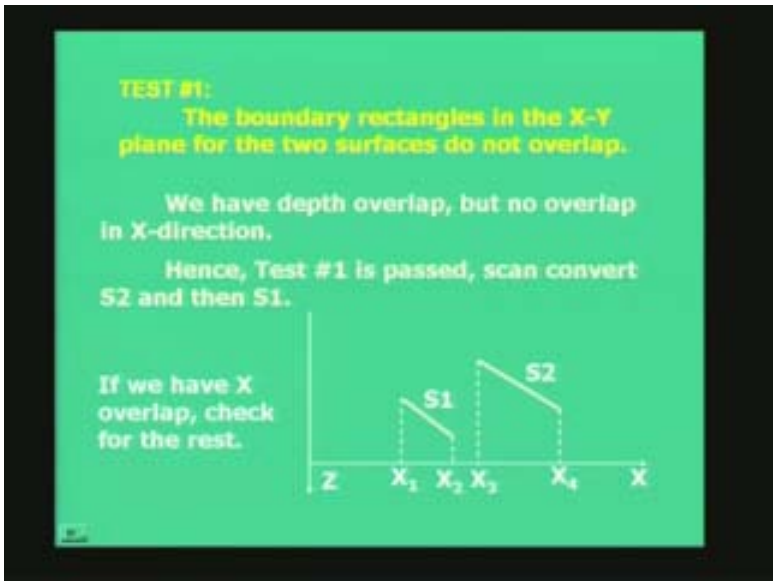
We can also do that test, it is equivalent to the inside outside test for a point within a rectangle so just check X and Y extents and there is no overlap like Z extent then what you will say is test one is passed and you can draw the surface in this case of course in any order but the order is of course disorder by the maximum Z extent of a surface.

So if you look into this figure the surface S2 will have the maximum Z extent compared to S1 so S2 will be drawn first and S1 and although there is an extent overlap since the extents along the X and Y do not overlap in this case we can draw it in any order in this case I can draw S2 and then S1 depending upon depth ordering and there will not be any problem. This is a scenario where we have just seen a case where there is a depth extent overlap but no overlap in the XY domain. Of course I shown the illustration only in X but you can visualize the same thing in Y that is possible.

We look into the image plane later on of X and Y extents both simultaneously when we come across test four so let us not talk about that right now instead only consider test one. We are entering test one only test zero fails so this figure shows that, remember there is overlapping in the depth or the Z extent of the polygons but there is no overlap in the X and Y limits or extents of this polygon so test one is passed and you render it.

Now, other condition can also exist that when will this test one will also fail. It is very easy for you to visualize that S1 S2 which were shown here has also X and Y extent overlap. So if you carefully look into this particular case where S2 is less than S3 or S3 is more than S2 which are the closest S limits of surfaces S1 and S2 and then S2 is more than S3 or S3 is less than S2 then surface S1 and S2 overlap and S1 fails so go to test two.

(Refer Slide Time 30:55 min)



So test one is passed in this case and you scan convert S2 and then S1. If you have an X overlap check for the second test or the rest of the test. So go to test number two, what does test number two say? Surface S is completely behind the overlapping surface relative to the viewing position. That is what you do in test number two. But let us look at this figure and understand why we have come to test number two at all.

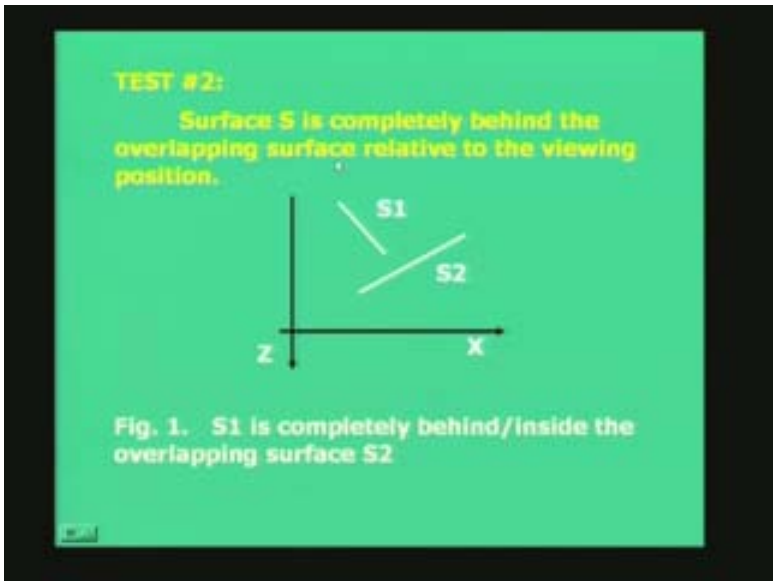
We have come to test number two because test number zero and one both of them have failed. The limits of the Z extents and limits of extents of the two surfaces overlap. That means also the minimum enclosed rectangle of the two surfaces overlap and the Z limits also overlap.

So test number zero has failed, test number one has also failed and we have entered test number two and that is what we read out the test number two where we say that the surface S is completely behind the overlapping surface relative to the viewing position. The surface S means S1 has to be rendered first in terms of order of depth. You see S1I has to be drawn first and then S2 and the surface S which talks about in this test is nothing but the surface S1 in the figure and we want to test whether it is completely behind the overlapping surface.

What is the overlapping surface? The surface S2 is the overlapping surface. In the words of the test here surface S is S1 and the overlapping surface is surface S2 because along the viewing direction which is negative Z S2 is the one which is overlapping S1. You test out whether the S1 which will be drawn first as scan converted first is it completely behind S2 which is overlapping that. And in the figure one we see that although the Z X and X extents of these two surfaces overlap. S1 is completely behind or the term inside is used the overlapping surface S2.

Now the terms behind or inside is used interchangeably. Behind or front, inside or out we will see these terms why they are used interchangeably. So, in this figure we see that S1is completely behind or inside the overlapping surface S2 and so you can go ahead and render S1 first and then S2. So the test number zero and one fail but in this case the test number two has been passed and therefore what you do is you render S1 and then render S2. There could be a condition when test number two fails then you take test number three.

(Refer Slide Time 32:04 min)



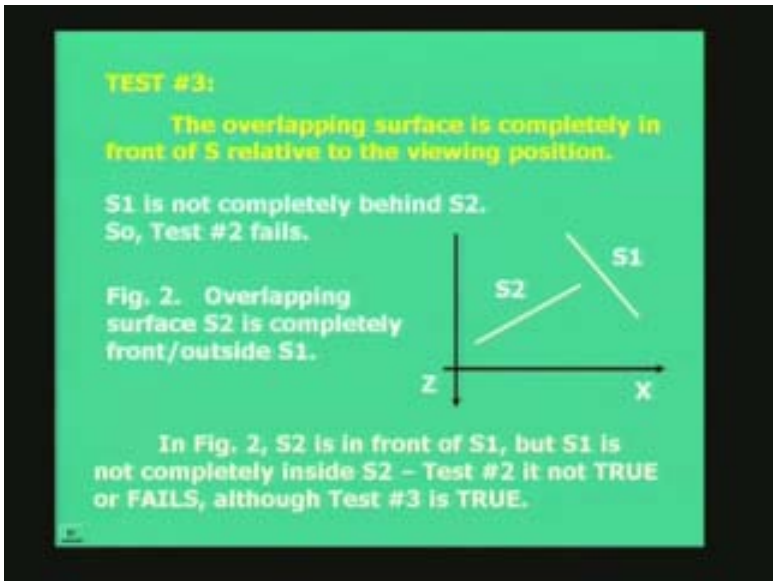
Let us look into the scenario where test number two will fail. To do that we look into this figure here. Why does test number two fails and that is why we have to go to test number three. S1 is not completely behind S2 as you can see here now. You check this figure and you will see that

test number zero of the Z extent overlap fails that is number one and X and Y limits also overlap so test number one fails and test number two also fails because the overlapping surface is S2 and the S1is under test but it is not completely behind S2.

I hope you understood why Z extent overlap fails X and Y minimum enclosed rectangle limits fail and the condition completely behind that is S1is not completely behind S2 so test number two has also failed and that is why we have entered test number three. What does test number three say? Let us read it, that the overlapping surface which is in this case S2 is completely in front of S which is in this case the surface S1relative to the viewing position.

Again viewing position is negative Z so you are sitting at origin and looking towards negative Z direction and so with respect to that I check out whether the overlapping surface which is S2 is completely in front of S related to the viewing position. So we know why with this figure or this arrangement we have come to test number three and what is this test number three? I have to check whether S2 is in front of S1 or not in this particular figure. So the test number three does pass because the overlapping surfaces S2 is completely in front or outside. These terms are again used interchangeably as we have seen in test number two we had behind or inside here you have front or outside. So these terms are use interchangeably behind or front is equivalent to say inside or outside. We will see why these terms could be used again interchangeably.

(Refer Slide Time 35:30 min)



We come back to this figure. In this case test number three has also passed in this particular figure arrangement and in figure two which is in this case S2 is in front of S1 but S1is not completely behind S2. That is why test number two is not true or it fails although test number three is true in this case or it is passed. I hope these words and the figure illustrates why this figure will show the test number zero one and two all of them fail and why the test number three will pass. It is because S2 is definitely in front of S1but since S1 was not completely behind S2 and that is why test number two had failed earlier.

And we went into test number three and that has passed and it is scan converted. What will be the scan conversion sequence? Scan conversion is done for the one which has the maximum Z out of the two so S1 will be scan converted first and then S2.

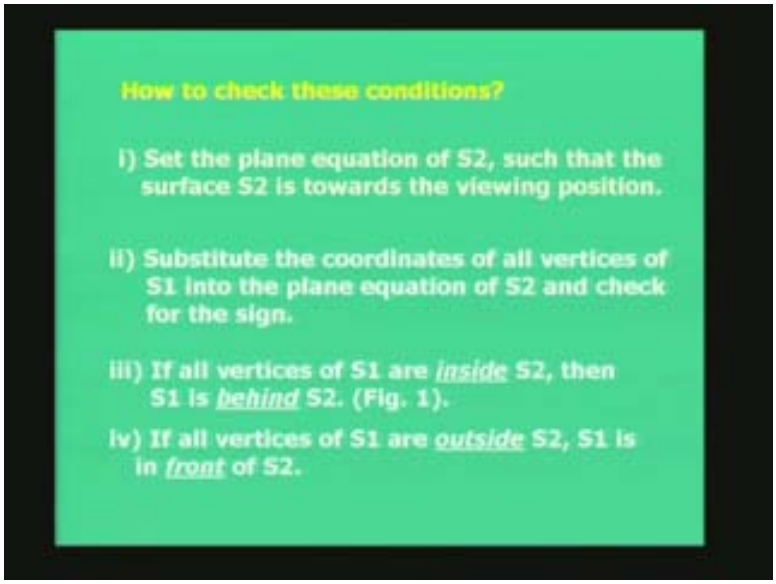
I hope you have drawn the figure otherwise if you look back here again after the test number three is passed use scan convert S1 and then use scan convert S2. I hope this is clear and you can have a scenario where test number three also fails. We will look into that scenario and then we have to go to test number four. We will look into the scenario where test number three also fails that means this surface S2 or the overlapping surface is not completely in front it could be the case not completely in front and the other one is not completely behind. This front and back test also has to pass test two and three, both these tests fails and we go to test number four.

But before we go to test number four which is a very simple case of a polygon's overlapping how to check this front and back conditions, that is important. We have to do little bit of mathematics again to check out on how to carry out this test, do that and you set the plane equation of S2 such that the surfaces S2 is towards the viewing position that is number one. Then substitute the coordinates of all the vertices of S1 into the plane equation of S2 and check for the sign, this is very simple.

You remember, the Bresenham's logic of the implicit equation form FXY equal to 0 could be for a surface for a line, inside outside test, half planes for clipping etc. Bresenham's positive negative zero on the line which is what you do in this particular case also. Take the vertices of one polygon substitute in the plane equation of the other one because the polygon table will have the surface coordinates you can get the surface normal and the surface equation. Substitute that and that value of that expression will tell you whether you are inside or outside based on the sign of that value. If it is zero you are on the surface negative on one side and positive on one side so that is what you do.

I repeat again, set the plane equation of S2 such that the surface S2 is towards the viewing position this is important so that the normal is pointing towards the viewer and then substitute the coordinates of all vertices of S1 onto the plane equation of S2 and check for the sign. If all vertices of S1are inside S2 only then we say that S1is behind S2. You remember figure one which was the case when we were discussing test number two.

(Refer Slide Time 38:40 min)

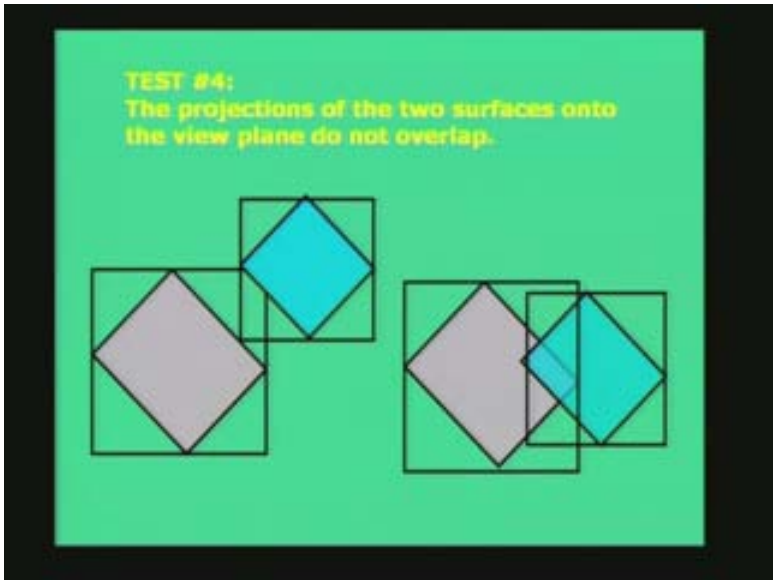


And if all vertices of S1 are outside S2 S1is in front of S2 this is with respect to figure two. So if all vertices of S1 are outside S2 then this is the simple way of substituting vertex into the other equation and that is what you do to check this condition. Now, going forward and looking into test number four. We enter this test only when test number zero one two and three all of them fail.

Again I repeat, Z extents fail, polygon minimum enclosed rectangle overlaps, front test fails behind test fails that is test number two and three and then test number four. If you see here in this particular case you can easily visualize that I am now working in the XY plane that is the image space and the minimum enclose rectangle or overlapping so test number one would have failed, test number zero also would have failed, the Z extent overlap and also for the time being the front and behind test also fails. In this case this is an example which we test whether the polygons themselves overlap or not.

So these rectangles as you see whether the blue one is overlapping the other one or vice versa and if not test number one four is passed in this case. And test number four is passed you just scan convert the wave we have been talking about in the previous one. When can test number four fail? Test number four can fail if the scenario something like this, not only bounding rectangles are overlapping the polygon themselves overlap. One of the polygons is entering in to the domain of the area covered by or the zone covered by the other polygon in both these cases.

(Refer Slide Time 41:22 min)



So this is the case when test number four fails so all tests would have fail now. Starting from zero, Z extents, test number one, minimum enclosed rectangle overlap, test number two front test, test number two behind and front in that sequence and then test number four where the polygon themselves overlap that also have failed. If that is the case it is only in this case we reorder the surfaces otherwise if any of these tests would have been passed 0 1 2 3 and 4 up to test number 4 any one of them would have passed what you would have done is you could have gone and scan converted in the way which would have been ordered earlier.

Only in the case when the entire test fails including the last one the very end that is the polygons overlapping then only we just swap the position in the order in this sequence of the two surfaces S1 and S2 which were ordered earlier based on maximum depth extent. That is what we have used to order them and then if all the tests would have failed if it is S1 and S2 just swap the position and make it S2 S1 and that is what you do.

So, coming back to this figure that is what we meant by that the test number four where it says that the projections of the two surfaces on to the view plane do not overlap so the left hand side figure says test number four is passed and the right hand side figure says the test number four is failed. How do you go ahead with this test? Minimum enclosed rectangle I said that you could take the vertices and check whether it is inside the other rectangle or check the X and Y limits extents that is very simple whether a polygon is inside another polygon or not.

A part of the polygon is entering into the two dimensional area of the other polygon. How do you test this? Can you think and give me a reply? Well, if you start to think about checking the vertices you can do a inside outside test of a point inside a polygon. Everybody knows that because we did that in the scanline polyfill algorithm that you can take any point and check whether it is inside another polygon.

So I can take the vertices of one polygon and test whether it is inside the other and so on and so forth you can do that.

This figure suggest we can do a little work but I can draw figures where none of the vertices of the polygons is inside the other one. Vertices of S1 are not inside S2 and vertices of S2 are not inside S1, this is not the case but still there could be a small part which could lie inside. Or basically one of the vertices have to definitely go inside another polygon otherwise it is not possible. But I can draw cases where definitely vertices are not inside but a section goes inside.

If you see carefully that you can draw a polygon, two polygons which can do that so vertices test is not sufficient. What you have to do is actually check out, can you guess what? Instead of vertices check the edges. Check if the edges of the two polygons any pair of edges, edge E1 taken from surface S1, edge E2 taken from surface S2 whether they actually intersect. They have to interest if there is an actual overlap between the two polygons. And we know to verify or obtain whether two edges are actually intersecting or not. Remember parametric equation of a line, find out two lines, two parametric equations which is a line clipping problem, we did this to find out valid values of T whether the edges actually intersect or not. So use that for pairs, form pairs of edges from two polygons, edge EI is from polygon one and edge EJ is from polygon two and check all EI EJ EI EJ combinations are intersecting.

If you find any one of these in fact more than one you should be able to obtain, if there is an actual overlap then you will find that one part of the polygon S1 has gone inside polygon S2 and test number four also has failed. But if it is not, that means none of the surfaces S1intersect with the edges of polygon S2 or none of the edges of the polygon S2 intersect with the edges of the polygon S1 and thus test number four has passed. You have passed test number four keep the ordering go add and scan convert in that order which you have put before the test number zero itself. So I hope you have understood the method by which you can obtain all the tests to be done. Remember, it is very easy I repeat it in sequence quickly, test number zero Z extent overlaps, test number one X and Y extent overlap that is also very easy to do and test number two and three the front and behind test you have to take the 3D vertices of a polygon S1 and check with respect to the plane equation S2 and vice versa.

Those were the test number two and three in 3D. Test number four finally is the edge based test where you actually check whether the edges intersect or no. That is what you do with test number four.

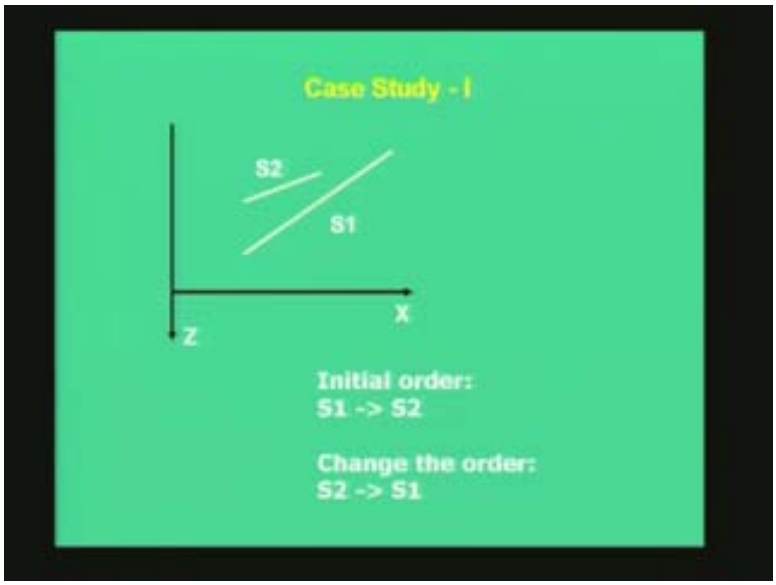
And if any one of these tests pass keep the ordering go ahead and scan convert in the order in descending order of depth and that is what you do. But however, if all test fails including test number zero one two three and the fourth test the last one which was just discussed now. Then I am sorry you have to reorder, swap the positions of S1 S2 which you have ordered earlier based on the maximum depth extents so that is what you do in test number four. So let us take a case study, a couple of case studies before we wind up today in the class for depth buffer order.

Case study one, two surfaces S1 and S2, well the initial order is S1 and S2, why? It is because the Z extent maximum depth extent of surface S1is having more depth and surface S2 that is the one which you have.

That is why S1 and then S2 is the sequence of the order. Now start the testing where you have the Z extent overlap so test number zero fails, that is very clear depth extents overlaps so before the zeroth test fail. X and Y extent overlap, yes that is also test number one fails. Test number one fails because extents overlap. Why also extent could be overlapped? Remember, we are taking the projection ZX plane, you can take on the ZY also so test number one overlaps.

What about test number two? What was test number two? The surface S1should be completely behind S2 because you are drawing S1first and S2 first that is the initial order. So based on that S1should be completely behind S2 but what does the figure say? It is impossible, S1is definitely not behind S2 in fact it is on the front the other way round, so test number two also fails.

(Refer Slide Time 48:26 min)



I hope you have understood why test number two fails because for the test number two to succeed S1should have been completely behind surface S2 which is not possible in this case so test number two also fails.

What is test number three? Test number three says that S2 should be completely front of S1 that also fails because S2 is not in front of S1it is the other way round. In fact S1is also not completely in front of S2 but S2 is definitely not in front. So test number three also fails what is test number four? The polygon themselves overlap in the X extent or X or Y. So, all the four tests have failed. Z extent fails X and Y extents of the minimum enclosed rectangle fails completely behind completely in front both have failed and the polygons themselves overlap so this ordering is not the correct ordering.

Here if you carefully see the viewing impression as given in the slide with the viewing direction going towards negative Z actually you should draw S2 first and then S1 because if you draw in this order what will happen is the S2 which is obscured or occluded by surface S1which is drawn later on will appear in the front no this is wrong. So logic says, by Painter's logic S2 should be drawn first and then S1 where the depth is ordering independence. So all the four tests have

failed, please go and change the order and then scan convert S2 and S1. If you scan convert S2 S1, S2 will be drawn and the S1will be completely obscuring the surface S2 which is correct in terms of the Painter's logic of the depth ordering.

We will move on to the case II which is a very interesting phenomenon. If you see here you see the initial order here which is very interesting S1 S2 and S3, why S1 S2 and S3?
You look into the maximum Z value for each of the surfaces and put that to obtain the initial order which I said earlier. So S1 has the maximum depth value again view direction along negative Z direction S1has the maximum depth then surface S2 and then surface S3. I repeat, S1 has the maximum depth then the surface S2 has a maximum depth and then surface S3 so the order is S1 S2 S3. Now the question which will come first is can I draw a surface S1 and then draw surface S2 and then surface S3. When the figure says you cannot do that if you look from the Painter's algorithm Painter's logic perceptually and visibly but you will go to the logical test.

Let us compare surface S1 and S2. Well, surface S1 S2 will not have a problem, it will fail the zeroth test of the Z extent overlaps but as you can see is surface S1 which is on the left and surface S2 on the right their X extents also definitely do not overlap also assume Y extent do not overlap so the test number one will be passed for surface S1 and S2. So you can draw surface S1 and then draw surface S2 so those two surfaces are ordered correctly for the time being and that part is okay. But there is another polygon in fact there could be many but we will just take an example of how to order three it is true that you can visualize for more than three four five up to a large value any number of polygons, several hundreds or thousands of them.

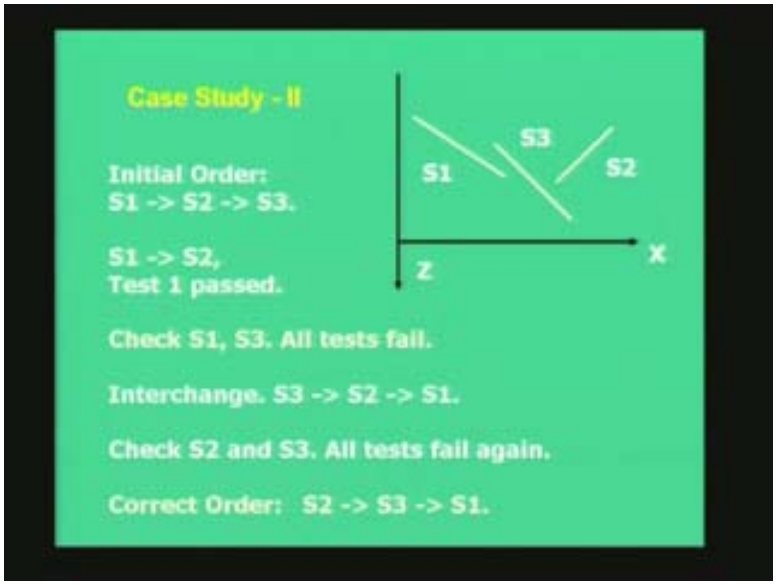
So coming back to the figure S1 S2 is correctly ordered whereas now you have to check between surface S1 and S3 now. That is, can you draw S1 before surface S3 you are sure that you can draw surface S1 before drawing surface S2 but is it the case with surface S3? Well, if you compare surface S1 and S3 test number zero fails because there is an overlap in Z extent, test number one also fails because the overlapping polygons have the minimum enclosed rectangles which is also overlapping and the behind test now.

If you look completely at the behind test do you think that the surface S3 is completely behind S1 no that is number one and then do you think that the surface S1 and S3 is completely in front so if you see here S1 one and S2 two has passed the test number one but check for S1 and S3 we will find that all the tests have failed because either S3 is completely behind S1neither S1 is completely in front of S3 as you can see here and the bounding rectangles also overlap. This is what you have for the surfaces S1 and S3 where all the tests will fail. And if all the test fails for surface S1 and S3 let us assume for the time being that as you can see most of the test S1 test number zero one two and four will also fail, completely the front tests will also fail which is the overlapping surface.

Overlapping surface will be S3. Interestingly overlapping S3 is not in front of S1 and S1 not behind S3. Remember, you are comparing S1 and S3 in the sequence that you want to draw S1 first and then S3. So you are expecting S1 to be behind S3 which is not and you are expecting S3 to be in front of S1 which is neither not the case so test number two and three will fail and test number four also fails. So you have to reorder S1 S3, interchange S1 S3 positions and this is the new sequence now which you have to generate. As you can see after swapping the S1 S3

positions from this initial order you have S3, S2 and S1. Probably you do not have to worry about S2 and S1 that test will anyway pass whether it is S2, S1 or S1, S2 but now we have to test S2 and S3. S2 and S3 is another sequence where we will say can I draw S3 first and then S2. Check S2 and S3 you can visualize that all the tests will fail again, why? Z extents overlap you can easily check along the viewing direction X extent overlap, S3, S2 basically means is S2 completely, well S3 will be drawn first so it will be behind no, S2 is in front no and extent overlap also test number four fails so S2 and S3 also fails that is also not the correct order.

(Refer Slide Time 53: 25 min)



So what do you do? You need to swap S2 and S3 positions. So the correct order which you will get is finally S2 S3 and S1. As you can see visualize if you can draw in this order where you can draw S2, S3 and S1 the S3 will be overlapping a part of little bit of S2 when you actually draw and paint and S1 will be overlapping little bit of S3. And this sequence is correct because S1 and S2 there is absolutely no problem and if you try to pass the test between S2 and S3 and between S3 and S1, one of the tests will pass and you will be able to correctly order that.

So we will stop with this particular case of the two test cases which you have studied two polygons and three polygons and we know when will all the tests be passed when the order could be maintained. When will be the case when you have to reorder and keep shuffling and reorder and get the correct sequence. We will continue in the next class with one or two more case studies where we will see very interestingly that there could be a scenario with just two or even three polygons where you keep reordering and reordering the surfaces.

And there could be a scenario where you could now entering in infinite loop even with two or three surfaces where you never stop but simply keep reordering and you will never be able to get the correct order, what you do then? Well there is a solution which we talked at the beginning of the class today, if you look back into your notes or if you are carefully note it down or we will talk about that in the next class that when we have to always keep reordering you cannot get a

correct sequence at all for two or three set of surfaces or even it could be more then you have to split the polygons into two parts if necessary.

We will see those case studies in the next class where the reordering is never possible in a correct sequence you can get into an infinite loop you have to somehow stop the sequence of testing split the polygon and go ahead. We will look into the case studies in the next class to wind up the discussion on Painter's algorithm and move on to new and other different types of algorithms on Visible Surface Detection, thank you very much.