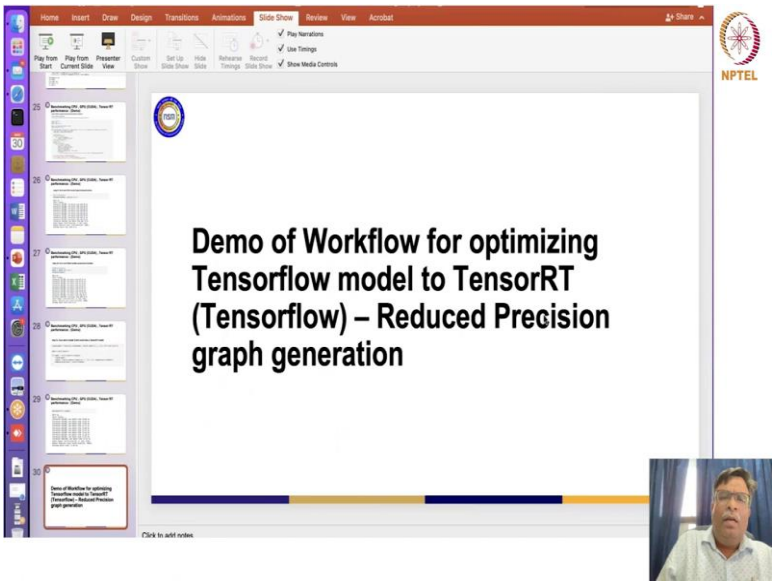
Applied Accelerated Artificial Intelligence
Prof. Satyadhyan Chickerur
Department of Computer Science and Engineering
Indian Institute of Technology, Palakkad

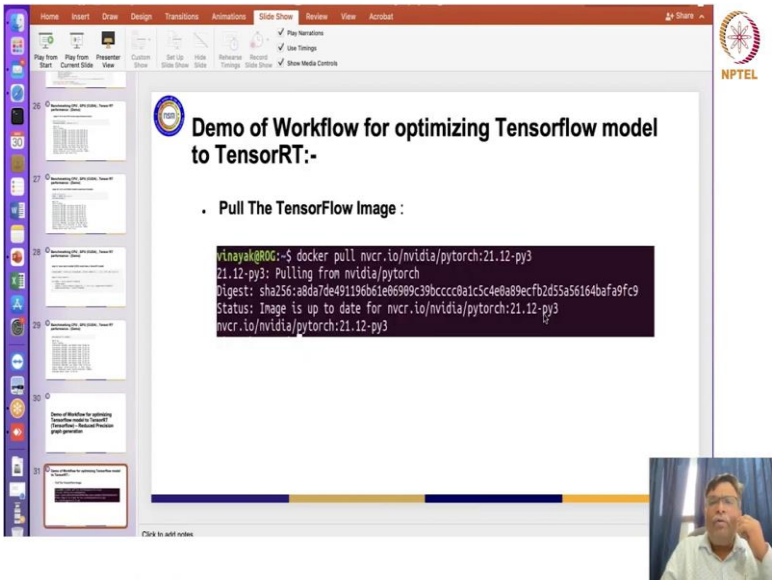
Lecture - 43
Accelerating neural network interference in PyTorch and TensorFlow Part 2

(Refer Slide Time: 00:15)



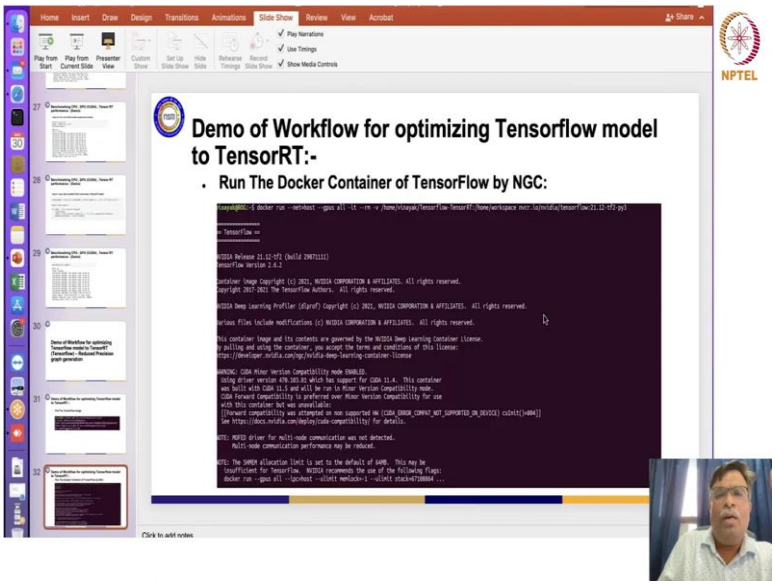
Now, one of the things which we would like to show you is a demo of workflow for optimizing the tensorflow model to TensorRT model using reduced precision right.

(Refer Slide Time: 00:29)



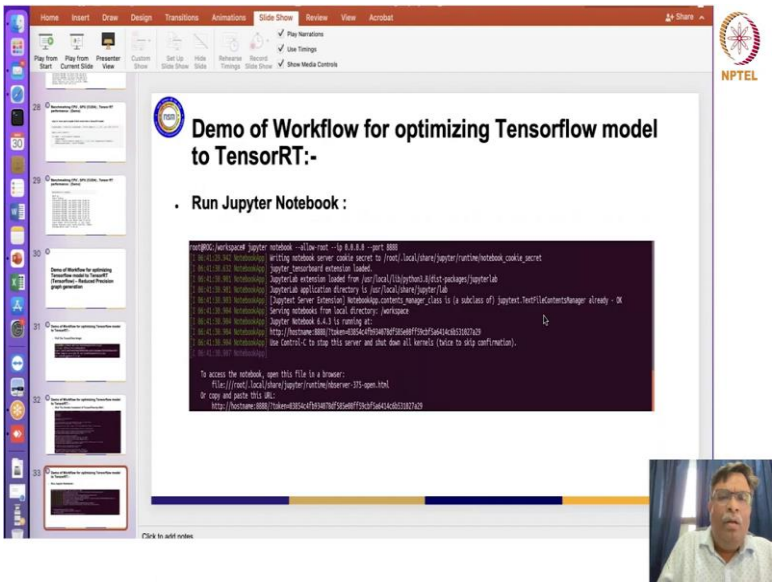
So, here we are trying to actually show you the graph generation ok. So, the what would be the graphs looking like ok.

(Refer Slide Time: 00:38)

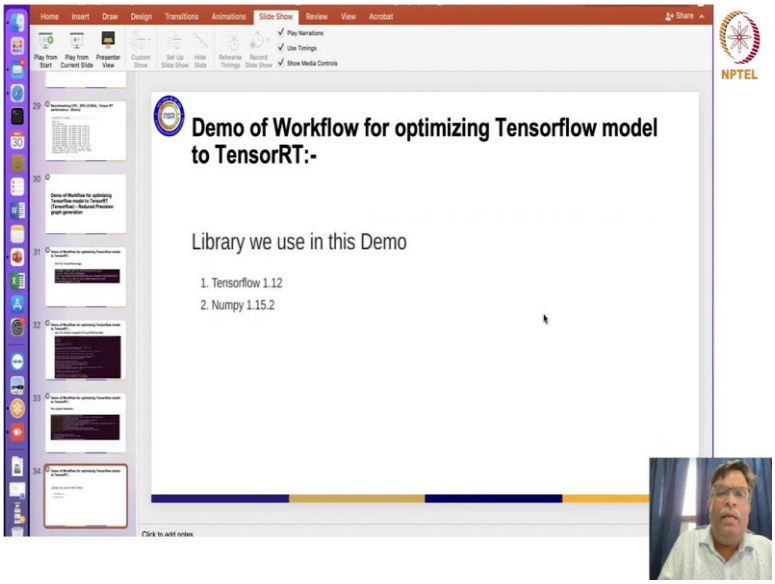


So, I will show you first the graphs which we generate and then we will show you on that system ok.

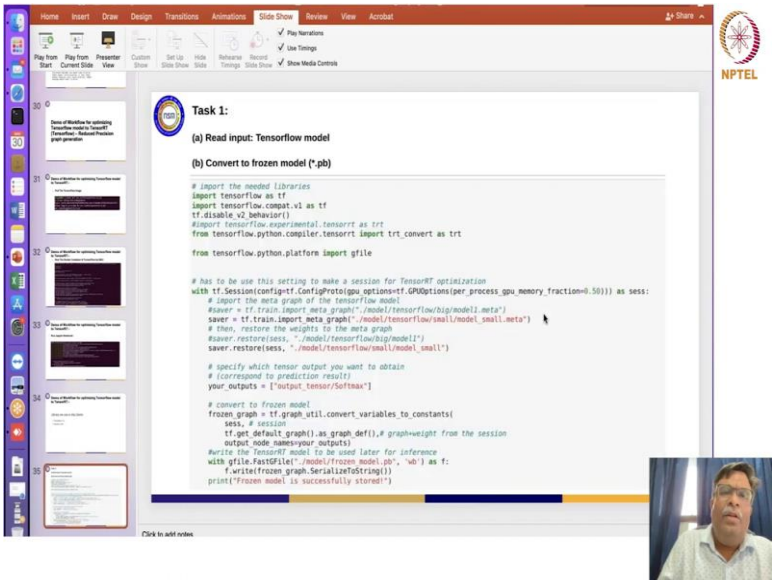
(Refer Slide Time: 00:46)



(Refer Slide Time: 00:48)

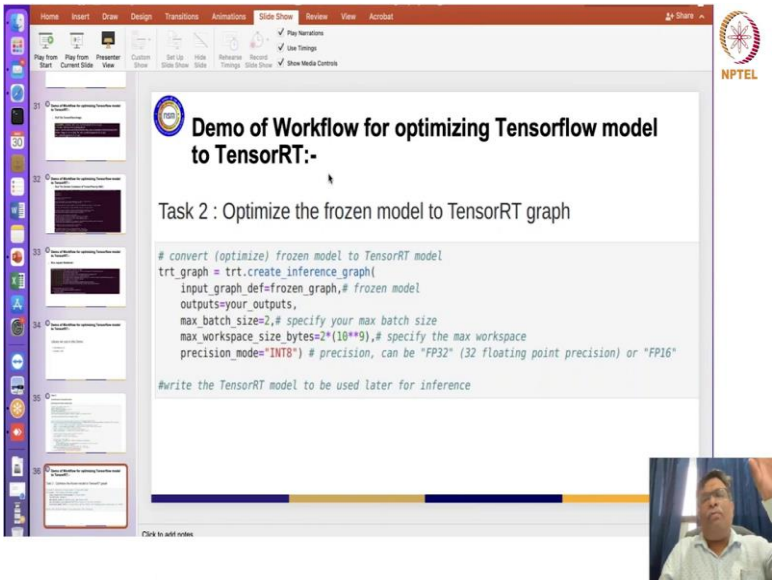


(Refer Slide Time: 00:52)

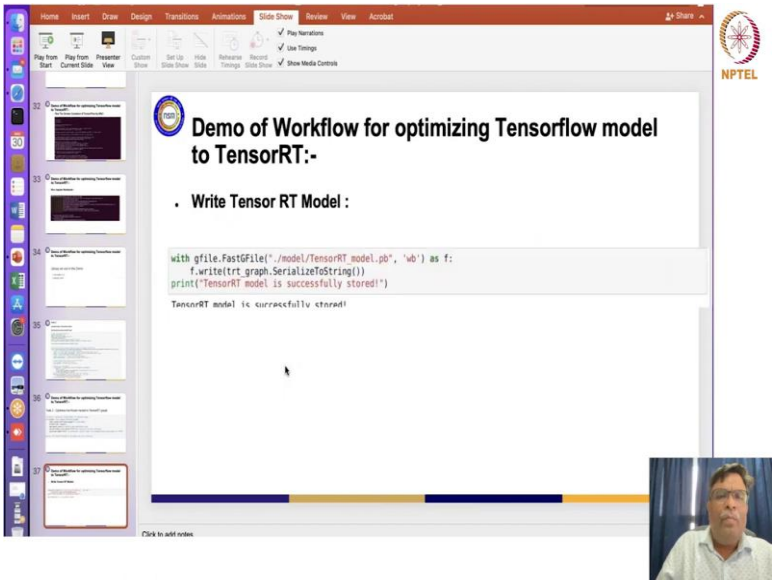


So, this is basically trying to pull the containers. Then you basically use this and then once you read a tensorflow model you convert to a frozen model which is *.pb ok this is what basically is a frozen model.

(Refer Slide Time: 01:07)

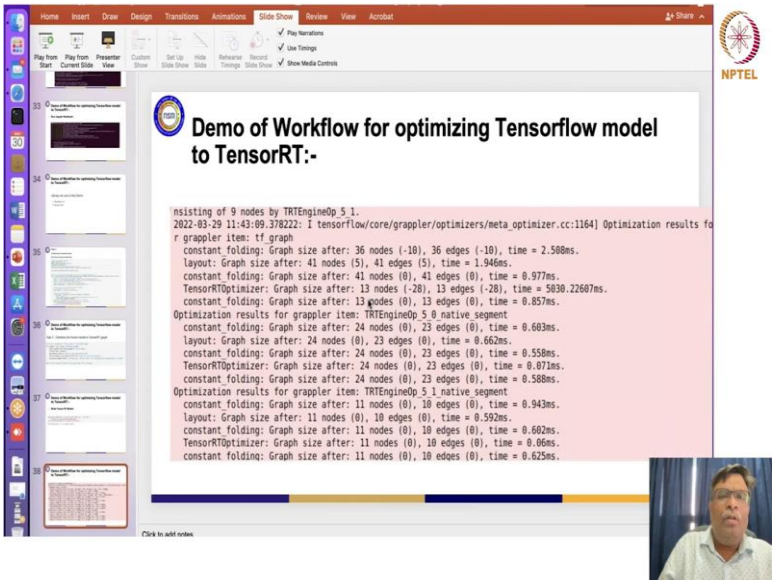


(Refer Slide Time: 01:27)

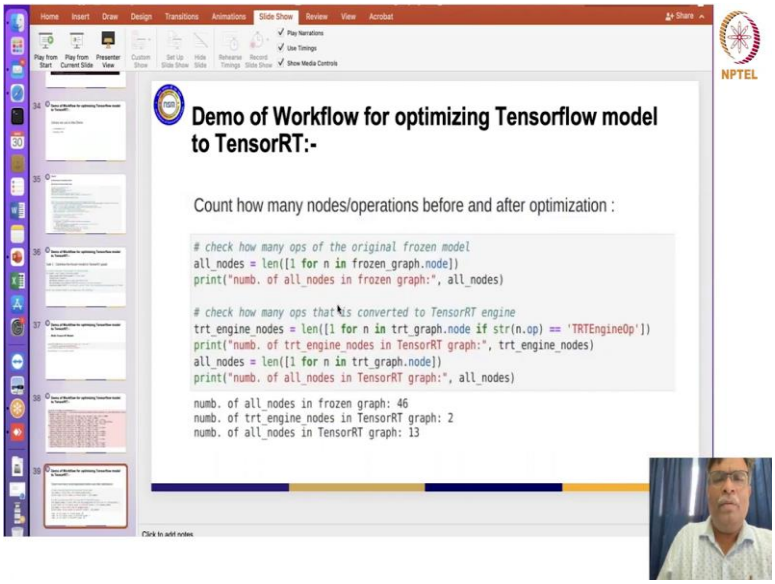


And then you optimize that frozen model to a TensorRT graph right this is how you do it you convert the frozen model to a TensorRT graph using trt graph. Then you actually put trt create inference graph all of these parameters right. And the precision mode which we discussed ok and everything like that. And then you basically get this something like a TensorRT model this successfully stored because we are printing it because we are trying to store this model ok.

(Refer Slide Time: 01:38)



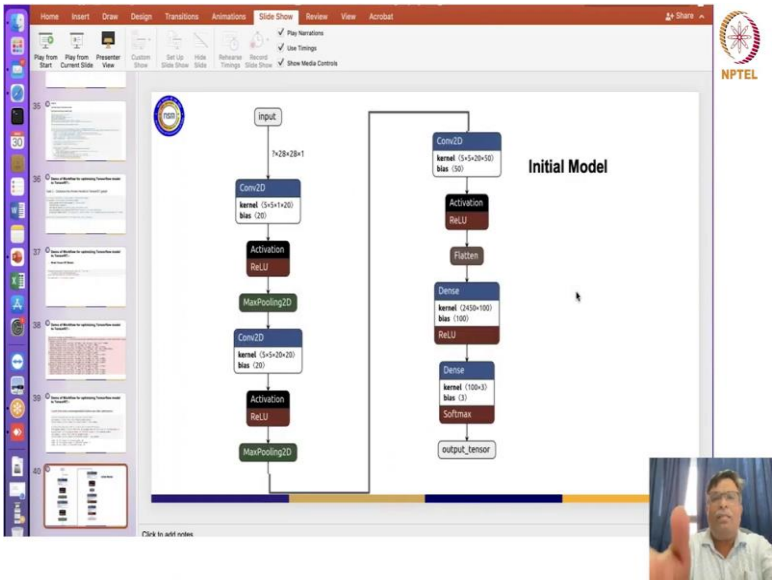
(Refer Slide Time: 01:55)



So, this is how we actually when you run you get something like the tf graph ok. What is the graph size? How many nodes? Ok. What is the graph size everything like this we will show you gradually of how actually it is optimized and ultimately what you get in the end is something like this. Number of all nodes in the frozen graph is 46 number of trt engines nodes in the TensorRT graph is 2 we will see this ok and number of all nodes in the TensorRT graph is 13.

So, this basically means you are trying to correlate a model which you have actually got it from a tensorflow ok to something which is optimized like this ok. So, how many nodes in the frozen graph? 46. Now what you do? You convert that into a TensorRT graph with how many nodes? 13 ok. And how many engine nodes in the TensorRT graph? There are only 2. We will see what does it mean by engine node and what are various graphs. So, why there are various nodes in the graph, but this is how it is.

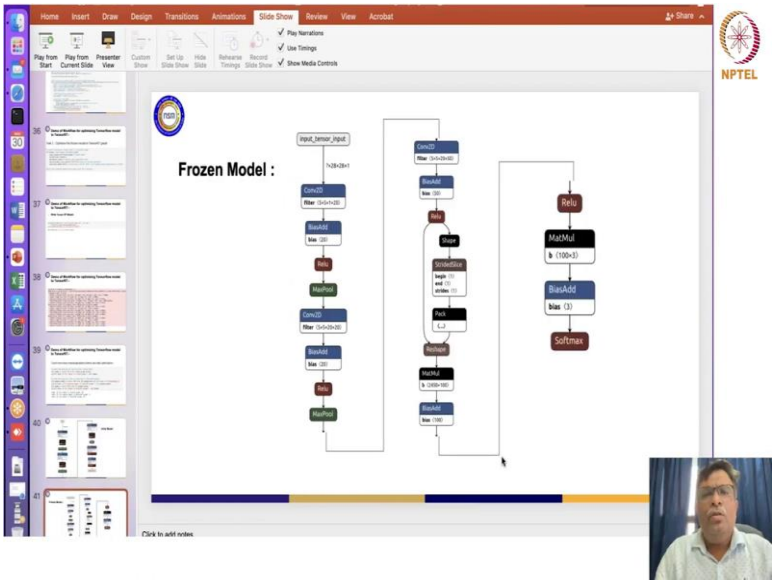
(Refer Slide Time: 02:46)



So, let us try to understand this I hope at least on the tensor board you would have seen this tensorflow graphs right. So, this is how a graph for a tensorflow program looks like ok this is a classification program you have input you have convolution layer with this size, this size of the kernel.

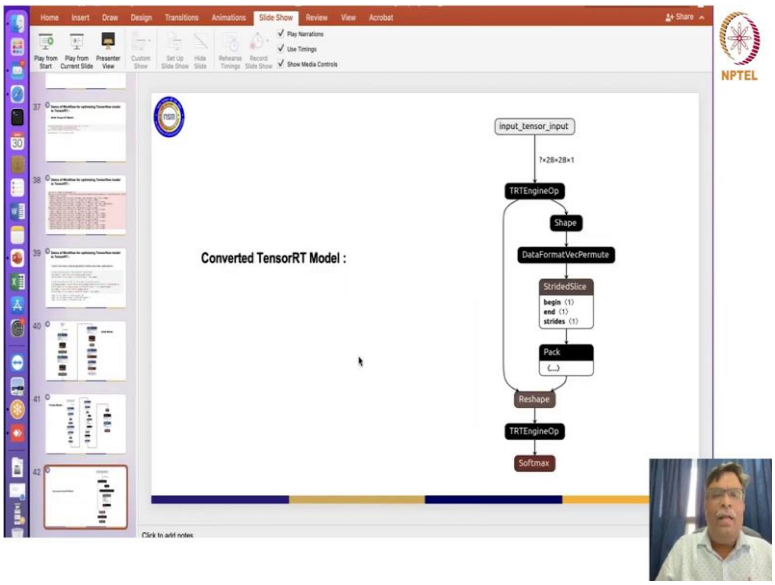
Then you have the ReLU activation then max pooling then again the convolution layer all of this like this is the initial model like you have a flattened thing you have soft max ok and then you have this output tensor. So, input and output tensor and this is the initial model ok. So, now, from this initial model after those steps you will get a frozen model. So, this is how frozen model is right.

(Refer Slide Time: 03:31)



So, you get a frozen model like this ok. This is converted ok this frozen model and then not going into the details, but as you go along you can actually understand right which of these layers ok are like basically you know finalized and frozen right. So, this and this is almost the same with only issue is you cannot change anything in this, this is not trainable now, but this is trained model this can be trained further right. And then the next thing is that you get a TensorRT model right.

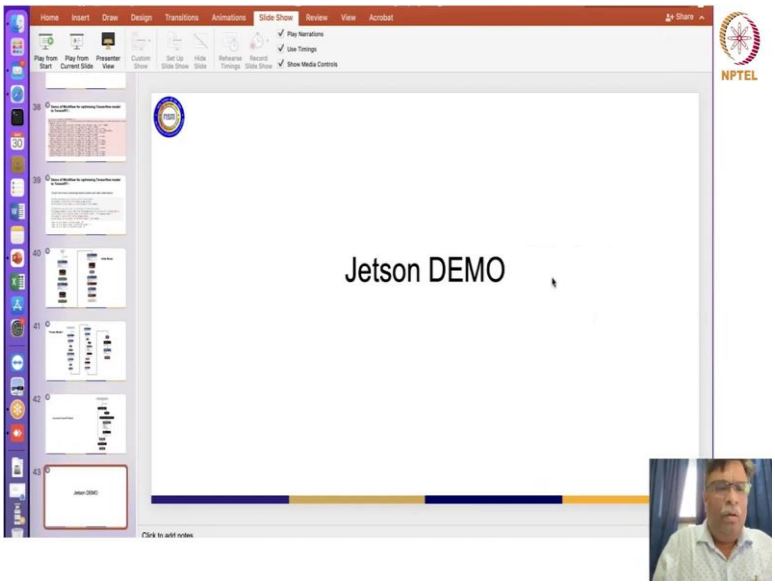
(Refer Slide Time: 04:14)



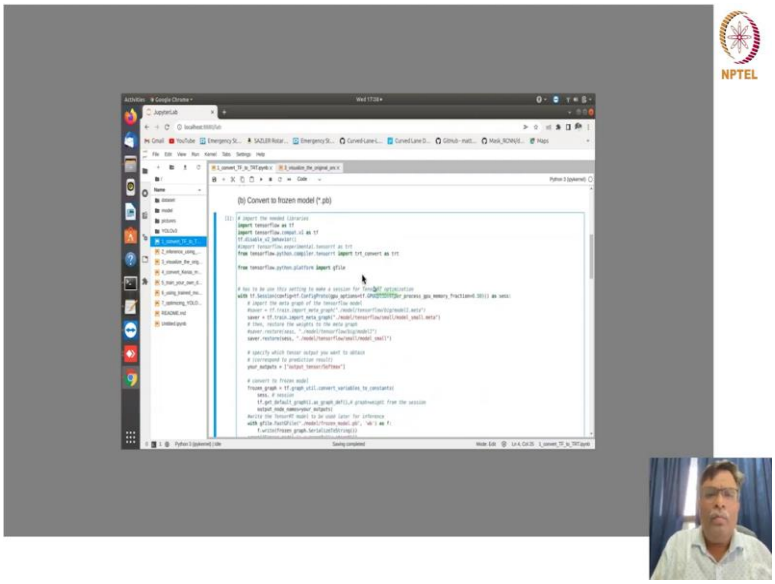
So, this is how the trt engine ok operator gets you this format. So, this particular model ok has been optimized to this wherein it uses fusing ok. Fusion of the layers horizontal fusion, vertical fusion, everything happens you get the tensor input ok. The same way and then you have this final output at the softmax layer this is what actually is converted from a frozen model to a TensorRT model.

We are not going into the details at present, but this is a very very good detailed what to say content ok which basically requires you a very very good understanding of how basically it has to be done if you understand the internal intricacies of the tensorflow the graph ok and all of that. So, we are just trying to show you in a short time as to how everything happens ok.

(Refer Slide Time: 05:30)

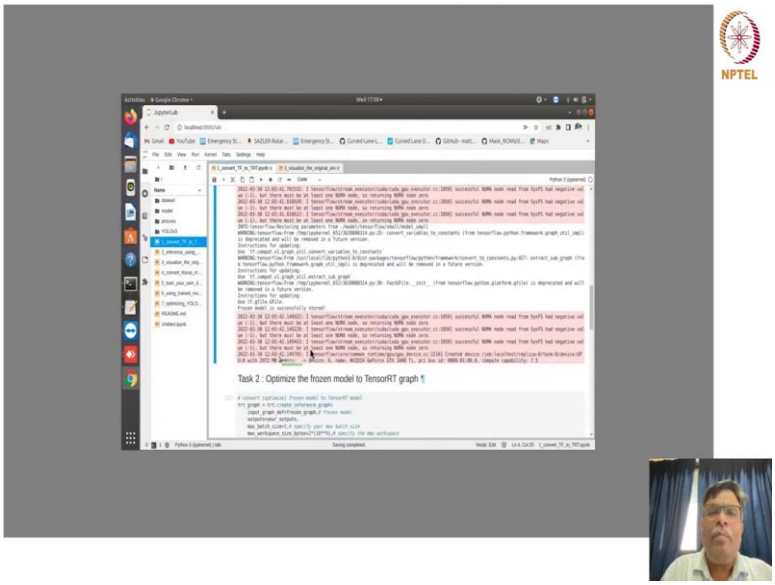


(Refer Slide Time: 05:43)



So let us try to actually do some demo now of how everything is done. So, let me just. So, basically we are going to import ok the tensorflow model this is how you actually import it ok. So, you are trying to import a model which is a small model ok. And then you basically try to actually convert that one minute convert that to a frozen model.

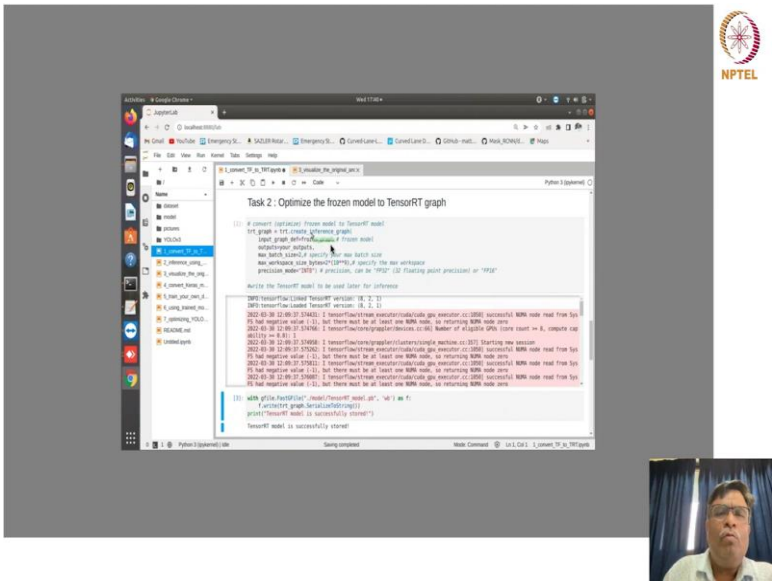
(Refer Slide Time: 06:14)



So, what you are trying to do is you are trying to actually generate a frozen graph ok from basically this. And if you see here you will actually try to use the what to say the frozen model ok is to be generated using CPU, GPU, CUDA, GPU executor right. So,

here what we are trying to do is we are trying to actually use a tensor flow. We are trying to understand which nvidia geforce gtx on which we are running ok on which device we are trying to convert it and then we get a frozen model here right.

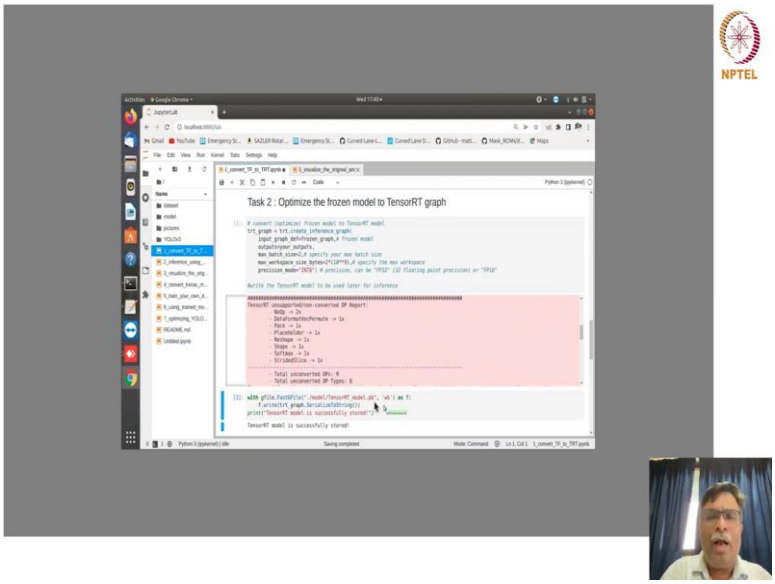
(Refer Slide Time: 06:57)



And when we try to optimize the frozen model to a TensorRT graph ok. So, this is a frozen model now. So, we are just trying to understand and generate a trt graph ok. So, we are going to create a inference graph and this is how the thing is my maximum batch size.

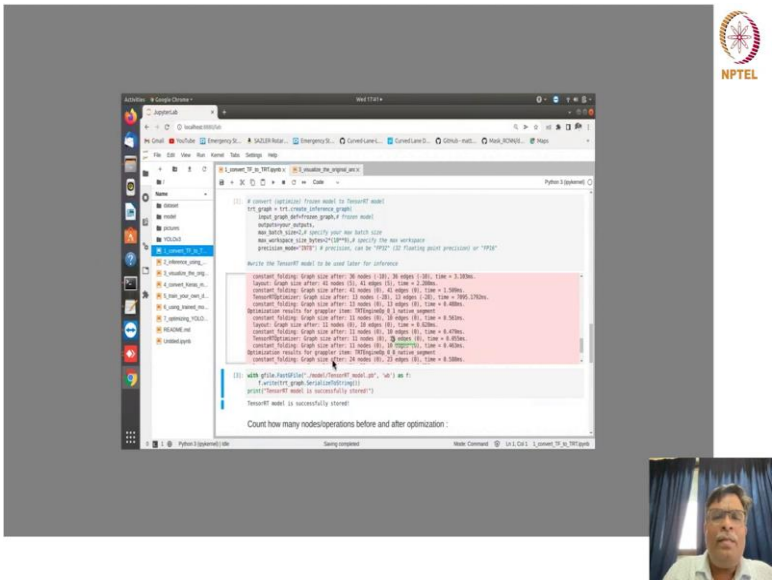
What batch size what is the precision we are trying to do ok. And then what is the input of to this particular graph definition. So, this is a frozen model right which we have used. So, we are using this frozen model in the previous step to generate a trt graph definition right and from that we are trying to create a inference graph right.

(Refer Slide Time: 07:44)



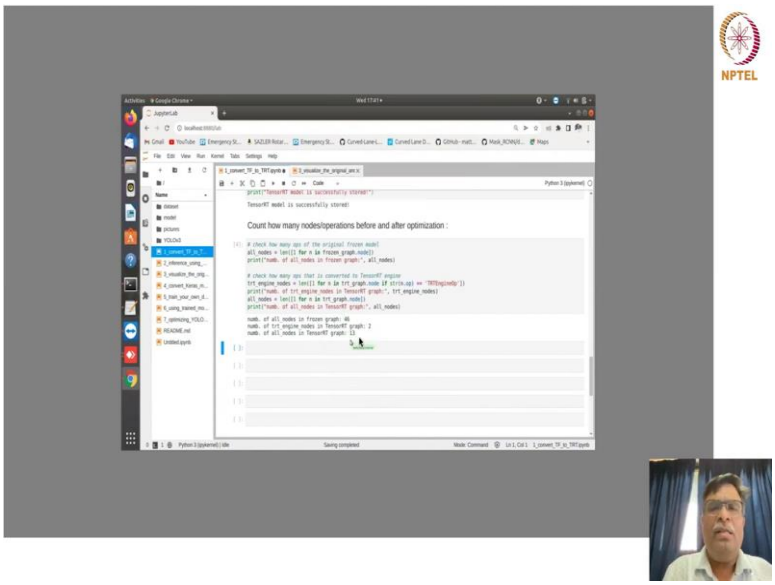
So, and we are going to write it in a dot model pb file ok. So, if you see here if you see here the TensorRT unconvert unsupported or non converted right operations. As I showed you there are certain specific supported conversion operations for TensorRT as I showed you in one of the PPT slides. But now here these are certain unsupported conversion operations right. So, total unconverted operations are 8 ok total unconverted operation types are 8 and total unconverted operations are 9 ok. So, this is how basically it is to be understood.

(Refer Slide Time: 08:32)



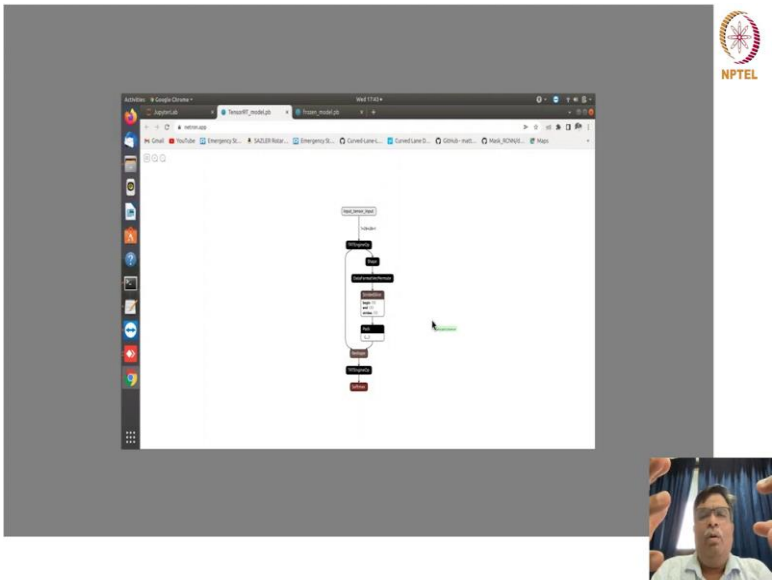
And then you basically try to actually ok understand right how is this graph size ok getting modified right. So, it is all linked to tensors and then the graphs right.

(Refer Slide Time: 08:47)



So, this is how it is going to actually do and then you basically ultimately ok. See a tensor model which is successively successfully stored and you can actually count ok how many nodes and how many operations were there before and after the operations this is what I showed you in the slide as well. So, number of nodes in the frozen graph is 46; number of nodes in the RT graph is 13 and then convert it into TensorRT model you can visualize this ok.

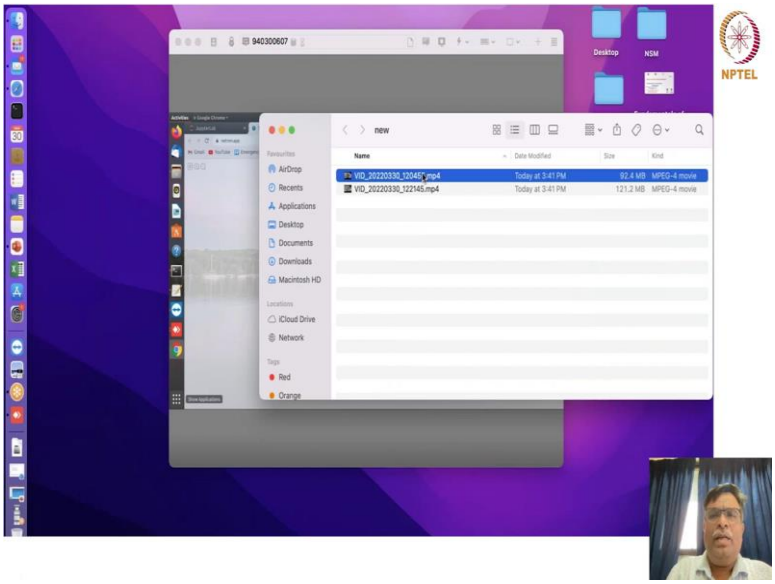
(Refer Slide Time: 09:22)



So, this is just to make you feel ok that once it is actually optimized for inferencing ok. How small the scale of the whole graph becomes ok and that is why the inferencing is such a serious issue wherein optimization is of utmost importance right. So, I hope this has made certain things clear on the part of how the optimization happens right. So, this is how the fusion happens and when we run the program.

Now we will show you right how the optimization or how the fusion happens right it shows you step by step ok. We will try to show you that now before that I will just try to run one or two videos in the meantime we will set that up for you to actually see ok. So, I will run the videos and in the meantime I will just set the camera also. So, give me a sec and I will try to run the video which we actually showed you yesterday certain applications.

(Refer Slide Time: 10:34)



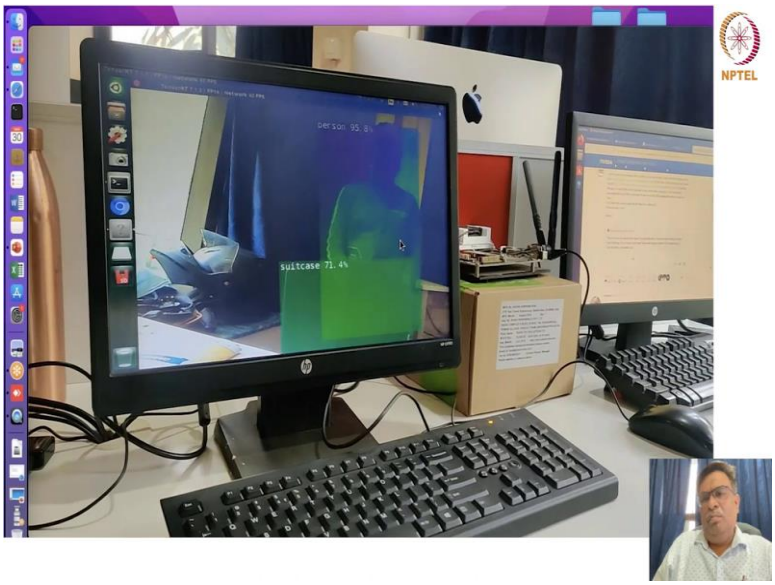
(Refer Slide Time: 10:36)

So, today we again try to do the same program run on a TensorRT optimized program ok on our jetson tx 1. So, you see this its a very very small program right it is not very what to say. Done for a specific application, but these are all available online we will give you the links. So, that you can also download and do.

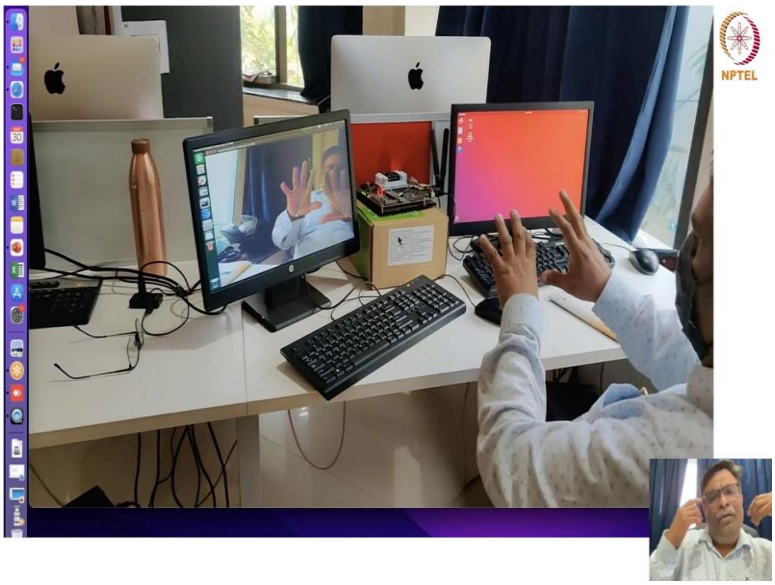
So, it can detect multiple persons multiple chairs ok not very much using a very very highly trained model, but ok. For some of the things right like suitcase and chair it is

showing, but ok these are certain things which you need to rectify, but we are just trying to show you that like this is how it can be done on a inferencing edge device right.

(Refer Slide Time: 11:38)



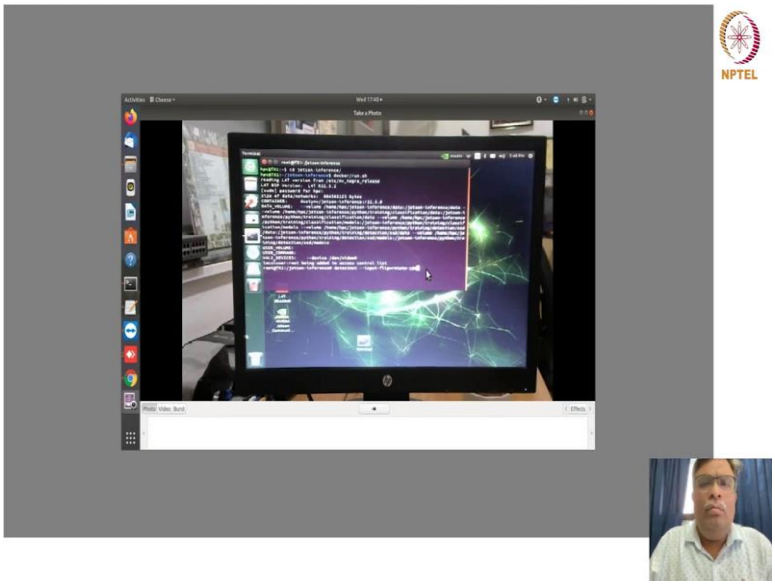
(Refer Slide Time: 11:49)



So, this is one thing which we try to show you and another one is wherein we do something of detection ok pose detection. So, this basically is a pose detection program which we are trying to run with a inbuilt onboard camera of on jetson tx 1. So, this is what we actually tried to do.

So, you have got a pose detection, you have got classification then you basically have a lot of other programs which are there ok. So, suppose got stuck or something, but yeah so something like this. So, let me just try to run it and 1 minute yes.

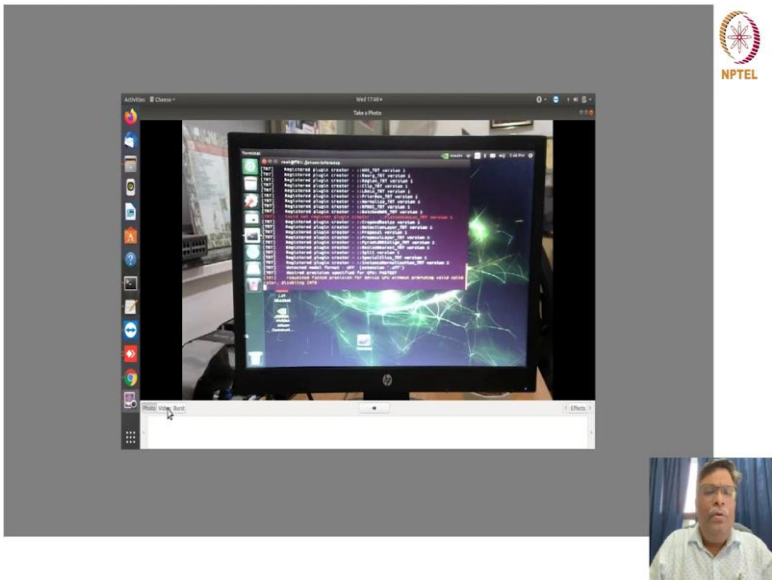
(Refer Slide Time: 12:54)



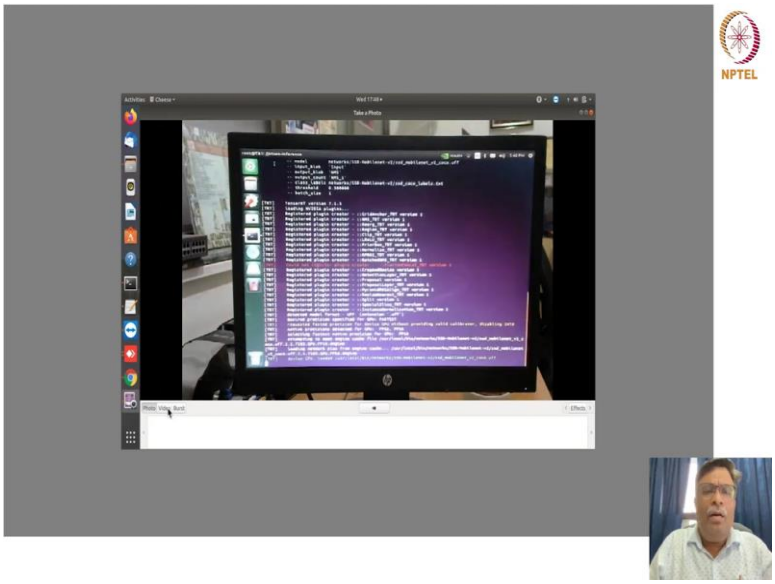
So, if you can see the monitor which basically is connected to the jetson tx 1. What we are trying to do is I do not know whether the resolution of the camera is good in a sense, but we have to enlarge it a bit. Let me check if we can enlarge it 1 minute no this is the maximum size let us see what happens it ok.

So, yeah somewhat it increase let me just check once again no ok. So, what we are trying to do is now we are trying to run detect net program ok for actually classifying right people ok and chair and all of that right. So, let us run the program and then see what basically happens ok image gets stuck it seems I do not know here we will have to actually convert it into video thing yeah.

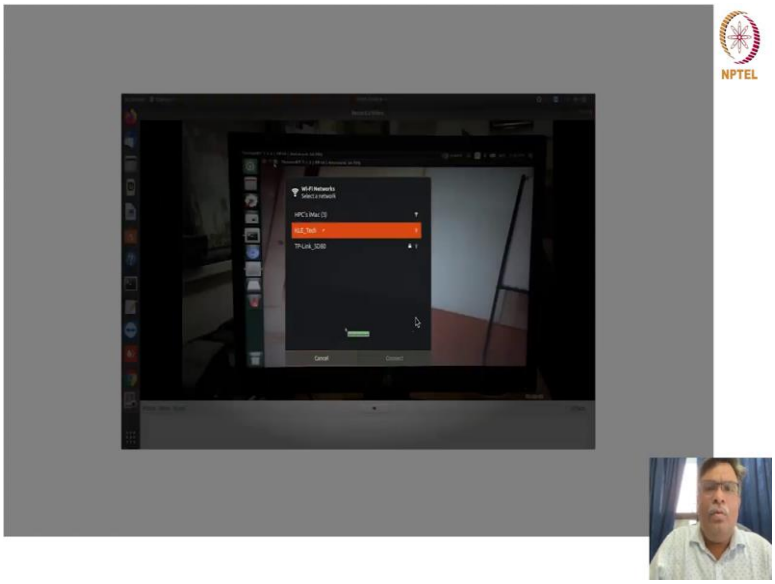
(Refer Slide Time: 14:20)



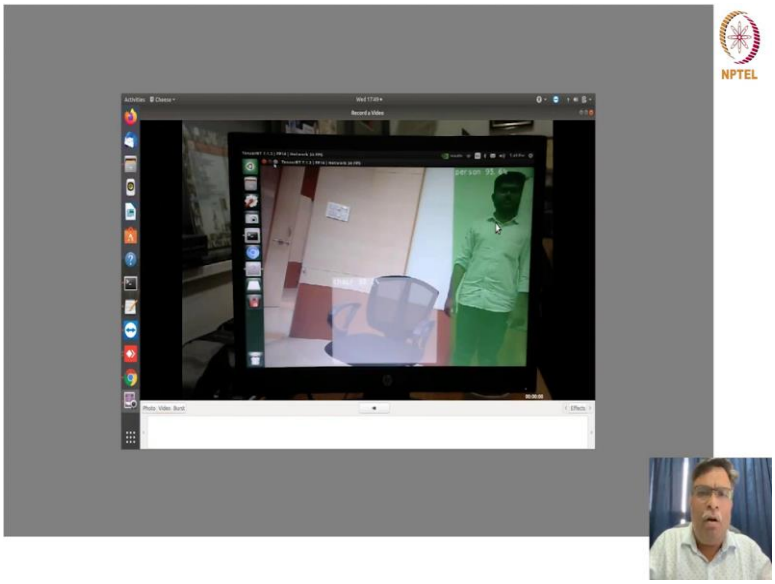
(Refer Slide Time: 14:31)



(Refer Slide Time: 14:34)



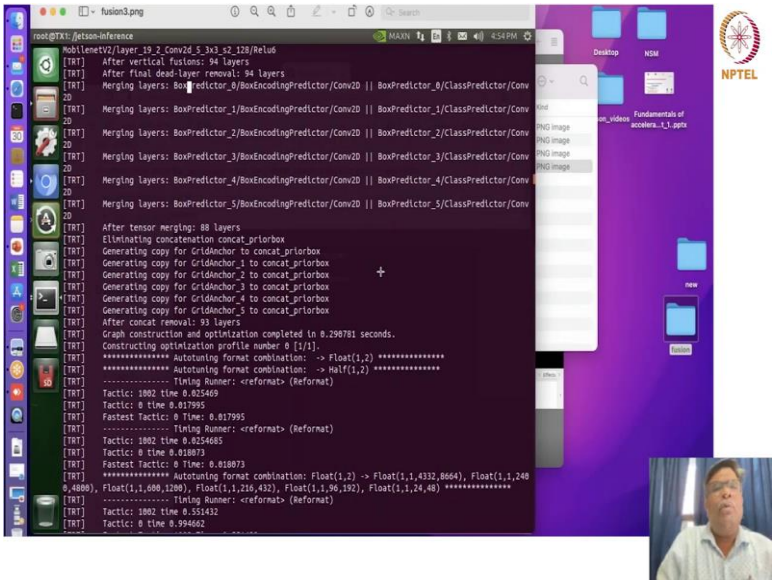
(Refer Slide Time: 14:40)



So, if you are seeing this you just convert ok. So, here if you are seeing this that this 1 minute it is a bit slow because of the lag, but the idea is that this is happening and then if we show you the conversion of things now the camera is not correct. So, it is not actually showing, but I suppose it is visible yeah.

So, if we stop this we can show you the scrolled history and there you can actually see the fused layers and all of this let me just stop this. And let me show you that show you right it is going to show you that merging the layers ok.

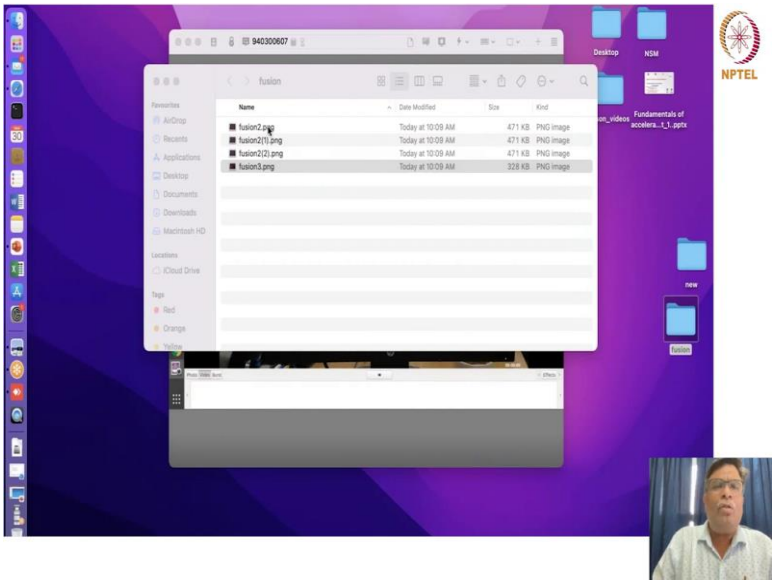
(Refer Slide Time: 15:40)



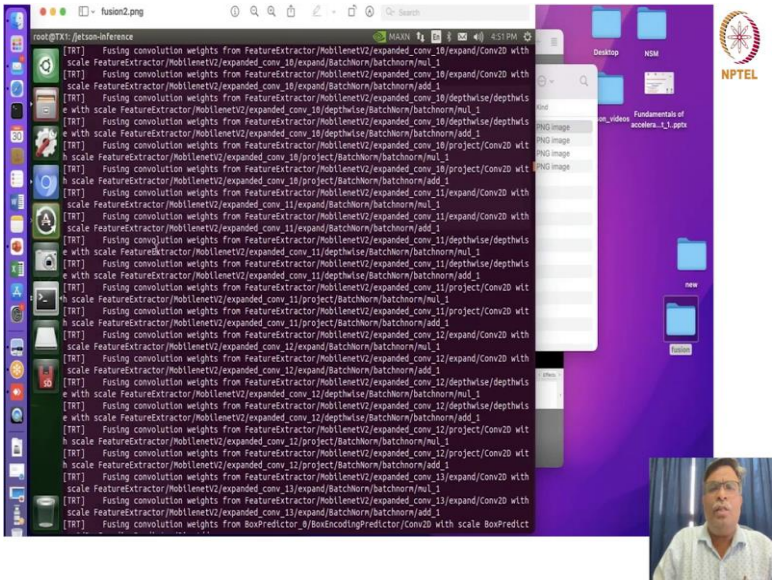
After vertical fusions there are 94 layers after you remove the dead layers right. There 94 layers again ok. And how much auto tuning format combinations right all of this after concatenation removal how many layers are left ok. So, how many layers are left and all of this is going to give you a fair idea of understanding and after you actually merge the tensors ok how many layers are there ok.

So, this basically is trying to give you understanding of how this mobile net program right. Of course, this is a different program from the program which I just now showed right that was for detect net and this I am showing you for mobile net. But that actually also will do the same thing of merging right and fusion and everything of this sort is going to happen.

(Refer Slide Time: 16:48)



(Refer Slide Time: 16:55)



So, in the beginning what is going to happen is you will get something like this right fusing convolution weights from feature extractor for that particular program which particular layer right convolution 11 layer. Then fusing convolution layer to then scaling the feature extractor with scaling and everything right.

So, fusing convolution weights and all of this are actually done for lot of things ok. So, this is the next thing. So, it is actually a continuous thing of trying to get all of this right. So, we will try to show you another program which basically is using I do not know it

get stuck again yeah. So, it basically is a detected program ok which you are trying to actually see here the device is a GPU it has loaded things ok and it is a bit slower.

So, it will take time yeah actually the idea is I cannot do SSH onto it because the GUI issue is there. So, I could not have shown you that this basically is a pose detection right which we are trying to do and of course, we can do SSH onto it and I can show you the running command after the GPU thing is over. So, I just wanted to show you this GUI stuff right the window stuff otherwise it would have got stuck.

So, that was the reason I did not use the thing SSH and do it because again it will create a lot of issues on my x server thing. So, that was the reason I had to do something like this. So, I hope it is understandable to some extent in such a short time as to how do you convert ok the program a tensorflow program to a TensorRT program and utilize it on something like a jetson tx 1.