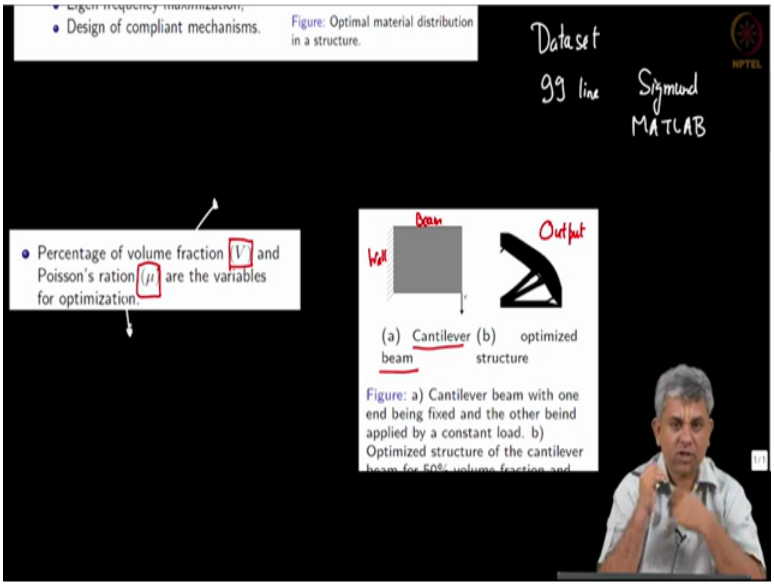
Machine Learning for Engineering and Science Applications

Professor Dr. Balaji Srinivasan
Department of Mechanical Engineering
Indian Institute of Technology, Madras

Application 3 Solution

(Refer Slide Time: 0:14)

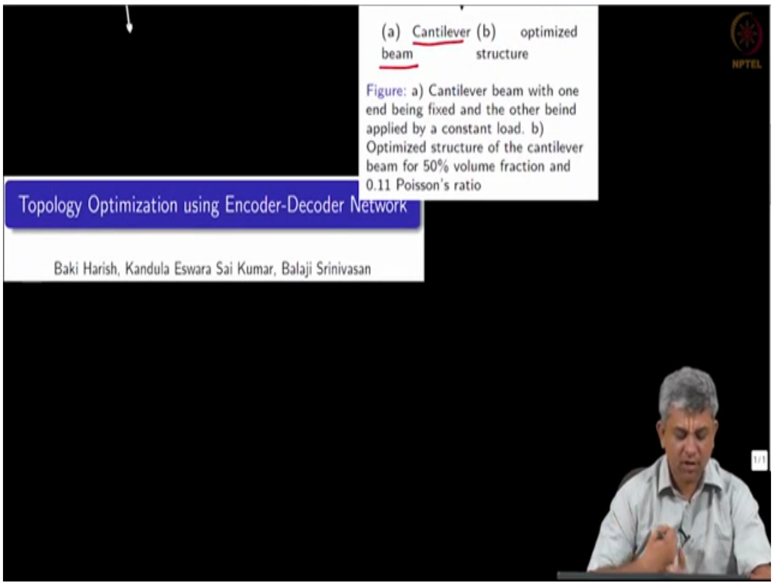


Welcome back, I hope you thought about the problem and notice that it is very similar to the fluid mechanics problem that we did in the last video with one or two notable differences. So the thing that is similar ofcourse is your inputs are somewhat similar though you do not have you know the signed distance function or anything as input, you can simply give the shape of the beam as input and in order to train you can give this as output, some optimized structure.

As it turns out the data set was generated by something called the 99 line code by Sigmund, this is available in MATLAB and this is what Harish and Sai used in order to create a data set, okay so they did that, so let us assume that you have this data set, okay. Once you have that data set, how are you going to give the input? Remember the input that we have has two other constraints also as input, one is the volume fraction and another is Poisson's ratio, so you want to specify both these as input, okay.

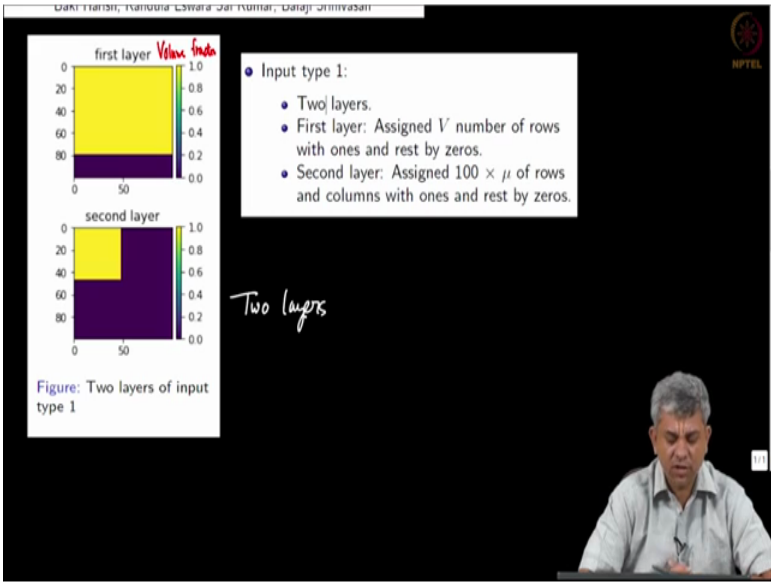
Now for a CNN it is ideal if we use images themselves as input and images as output, there are multiple other structures possible for this problem, but I will show you just one for this which is an image as input and image as output, okay.

(Refer Slide Time: 1:53)



So this work as I said is by Harish and Sai Kumar, so let us see what they used should input and output.

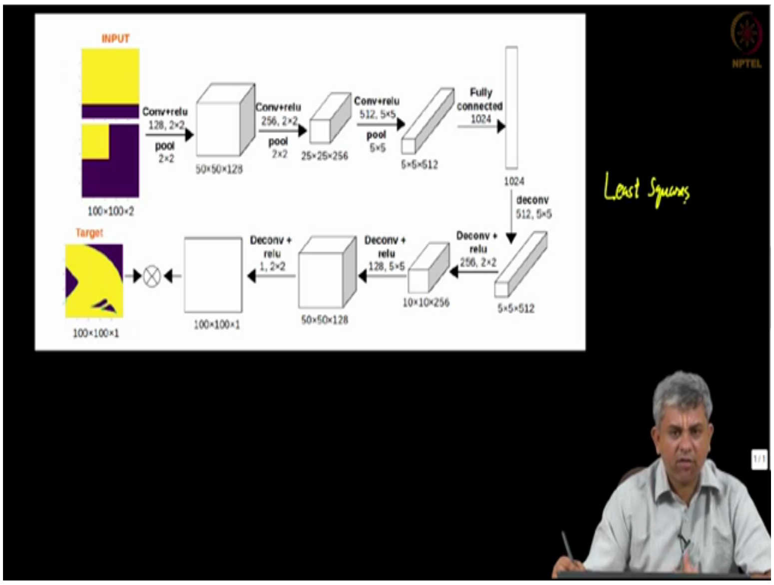
(Refer Slide Time: 2:01)



So here we see what they used as their input, what they did cleverly was to use the volume fraction as an image. So for example if your volume fraction is 80 percent you have an image which is 100 by 100 and about 80 percent of it is filled so that in some sense over iterations the algorithm learns to use only this much yellow and redistributes it amongst the figure so as to get the final topology optimized structure.

The second thing was to use Mu which is also a ratio which is also a decimal number also as a ratio except in this form, okay. So both these were given as two layers of inputs, remember when we were doing CNNs and we had multiple layers RGB. Similarly in this case Harish and Sai used two layers one of them for the volume fraction, one of them for the Poisson's ratio and then they gave the optimized structure as output, once again you train on multiple outputs, I am going to show you the network architecture that they used, it is very very similar to the structure that you saw for CFD.

(Refer Slide Time: 3:28)



So here is the structure that they used, here is the input, once again an encoder, decoder structure you will see that it is fairly similar to what we saw for CFD, once again you kind of systematically come down here, then systematically build it up back to the target, the loss function is ofcourse the least square loss function, you can use other loss functions also but that is what they used.

So you can use this, you can also treat it as a segmentation task for (())(4:06) and then they trained it, okay so you saw that you will see that they actually had pretty decent success with this kind of architecture.

(Refer Slide Time: 4:18)

Now here are some few results for this problem as obtained by Harish and Sai, you will see that for a volume fraction of 45 percent and a Poisson's ratio of 0.2, on the left side is ground truth, so you can see that 45 percent of the material has been redistributed in various ways and the right hand side is our CNN solution. Once again you can notice that actually visually they have it has succeeded quite well, ofcourse you can never create material of this sort, you know it has removed some arbitrary small points here and there.

But as an initial cut for what the shape will look like this is actually remarkably good. you will see some difference here ofcourse, okay but rather than that this is actually remarkably good solution. So this problem also shows you that you can take problems that are fairly complex atleast physically they are fairly complex, I do not think any of us can intuitively say you know where should be the distribution of these gaps just by being given 45 percent of the material should be there, you can see that remarkably which portion should be empty, which portion should be full has been predicted really really well by this CNN structure showing how powerful it can be even when you map images to images.

We will see I will actually discuss towards the end of this week some other examples of how you can combine CNNs with LSTM, I will just briefly describe it, I will leave it to you in order to actually read the original papers.

So once you have this powerful idea of using CNNs for field data, you can combine it with various things, you know various other architectures CNNs plus ANNs, CNNs plus LSTM to do all sorts of problems, I will discuss very briefly not in as much detail as I have done now, towards the end of this week I will have a very short section on what else can be done with CNNs in that.

In the next video I will discuss one final problem that will be the final application for this week, the final application is going to be how do we actually solve from scratch without any examples, in all these cases you had to be given examples, without examples how can we solve ODEs and PDEs. So we will start with that in the next video, thank you.