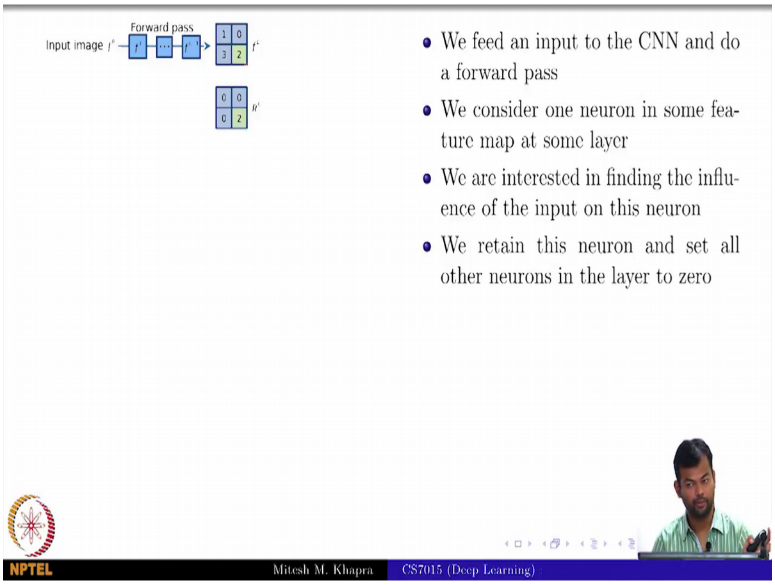
Deep Learning
Prof. Mitesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture - 97
Guided Backpropagation

So, we will see what guided Backpropagation is. So, idea here is a bit hacky a bist heuristically, but it still works very well so, let us see what it is right.

(Refer Slide Time: 00:20)

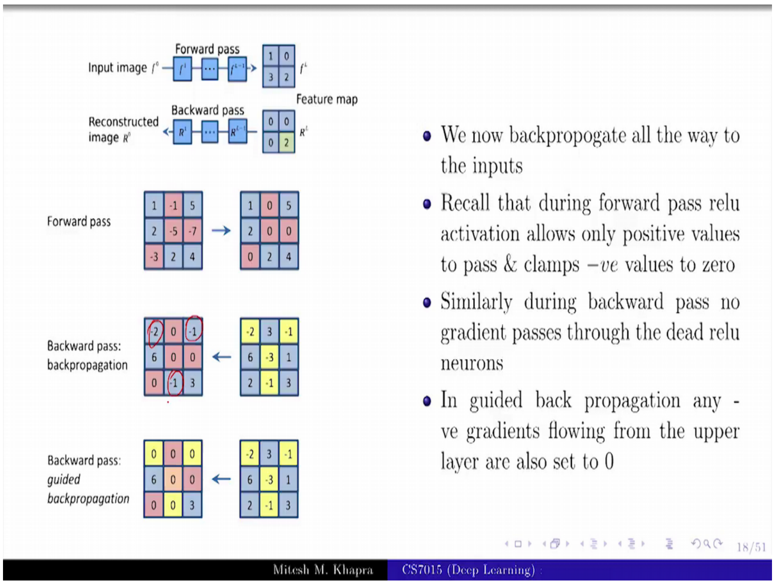


- We feed an input to the CNN and do a forward pass
- We consider one neuron in some feature map at some layer
- We are interested in finding the influence of the input on this neuron
- We retain this neuron and set all other neurons in the layer to zero

So, suppose you feed an input image to a convolutional neural network, that image will go through all the convolution layers and say it one convolution layer this is what your feature map looks like. I am operating at a very small scale, I am just considering a 2 cross 2 feature map ok.

Now, we consider 1 neuron in some feature map at some layer so, we will consider this particular neuron. And we are finding interested in finding the influence of the input on this neutron. So, this is what I will do is, I will set all the other neurons in this layer to 0 because, I do not care about them, I only care about this particular neurons, I just focus on that.

(Refer Slide Time: 01:00)



$$\begin{bmatrix}1&0\\3&2\end{bmatrix} \quad \begin{bmatrix}0&0\\0&2\end{bmatrix}$$

$$\begin{bmatrix}1&-1&5\\2&-5&-7\\-3&2&4\end{bmatrix} \rightarrow \begin{bmatrix}1&0&5\\2&0&0\\0&2&4\end{bmatrix}$$

$$\begin{bmatrix}-2&0&-1\\6&0&0\\0&-1&3\end{bmatrix} \leftarrow \begin{bmatrix}-2&3&-1\\6&-3&1\\2&-1&3\end{bmatrix}$$

$$\begin{bmatrix}0&0&0\\6&0&0\\0&0&3\end{bmatrix} \leftarrow \begin{bmatrix}-2&3&-1\\6&-3&1\\2&-1&3\end{bmatrix}$$

- We now backpropogate all the way to the inputs
- Recall that during forward pass relu activation allows only positive values to pass & clamps $-ve$ values to zero
- Similarly during backward pass no gradient passes through the dead relu neurons
- In guided back propagation any -ve gradients flowing from the upper layer are also set to 0

18/51

Mitesh M. Khapra CS7015 (Deep Learning)

And we now back propagate all the way back to the image right; that means, I will compute if I call this as h 2 then, I will compute dou h 2 by dou i 1 i 2 i 3 and so on ok.

Now, recall that during forward pass what happens is because you have relu neurons, any output which was negative that was clamped to 0. In the forward pass any output which was negative was clamped to 0, so what would happen to the gradients when they flow back, through those neurons? You already did this, if an relu neuron is dead, the gradients do not flow back right. So, the gradients will not flow back through these neurons; that means, that only the so, only these gradients will actually flow back, which correspond to non negative entries in the image before it or in the matrix above it right. Is that fine?

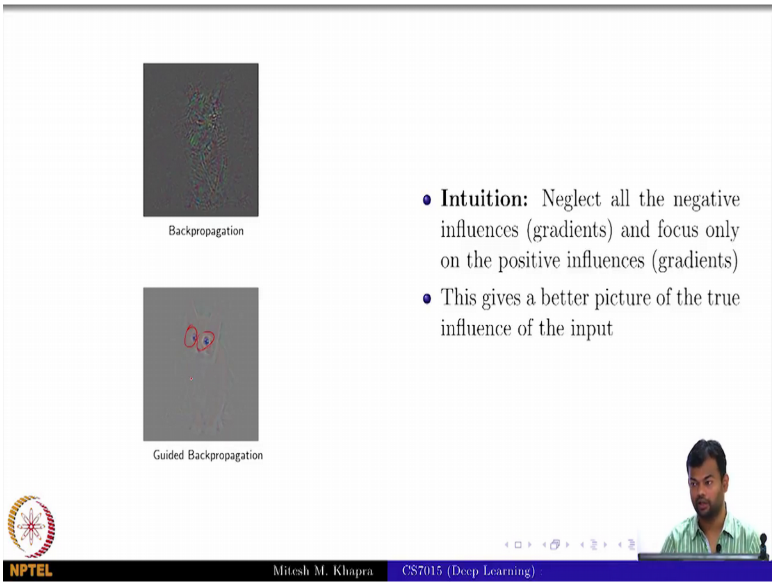
So, now these guys use this interesting idea that in the forward pass you do not allow negative things to go forward. So, the backward pass also do something similar, do not allow the negative influences to go back; that means, any gradient which is negative just clamp it to 0 ok. So, what I am going to do is all these negative elements in the gradient, I am going to set them to 0, you see that. So, this is just taking the same idea which you apply that forward propagation that relu clamps the output to 0 if, the influence was negative and the backward pass also do the same, any gradients which are negative just clammed them to 0.

So, the intuition here was that maybe there was a pixel which is really influencing the particular neuron and it stands out, but because there are some positive and negative gradients flowing back, they seem to cancel each other. And all these influences tend to be 0 because, that is what we observe that image was largely gray with very few non gray pixels.

So, this is very heuristically because, the reason I call it a heuristic is because, you are messing with the math right, the math tells you that the correct gradient has to go back irrespective of whether, it is positive or negative but they give this justification that on based on two things. And the forward pass you are not passing the negative gradients or negative outputs, so in the backward pass also kill them and this should avoid this canceling of positive and negative output.

So, this is known as guided back propagation because, you are meddling with the actual back propagation, you are doing something different.

(Refer Slide Time: 03:24)



- **Intuition:** Neglect all the negative influences (gradients) and focus only on the positive influences (gradients)
- This gives a better picture of the true influence of the input

And, so the idea was to neglect all the negative influences and when they apply this guided back propagation, this is what the influence looks like. So, you see that it is much sharper now, it is actually very nice its focusing completely on the eyes and you can see the layout of the cat much more clearly as in the earlier picture earlier image right.

So, this is a popular technique to use to for various things it is also among other things, for in for understanding what your convolutional neural network is doing right. So, this lecture is entirely about understanding what are the neurons learning, what are the weight matrices learning, what are the kernels learning and so on. So, these are all again tricks that you need to have in your repository to be able to do something more than just reporting accuracy ok. I get 70 percent accuracy on this status (Refer Slide Time: 04:15) right. So, this guided back propagation is 1 algorithm that you will implement as a part of the assignment so.