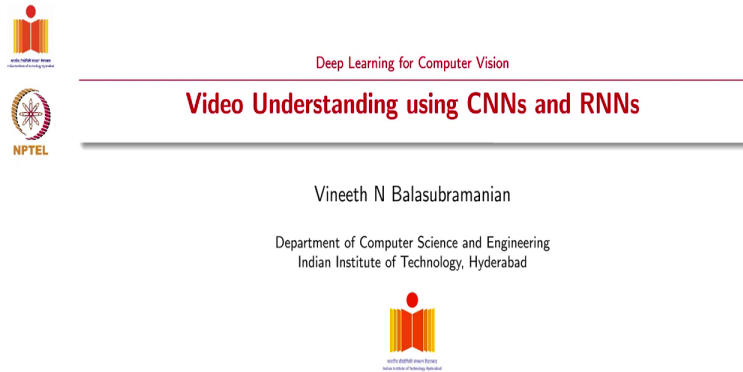


Deep Learning for Computer Vision
Professor Vineeth N Balasubramanian
Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad
Video Understanding using CNNs and RNNs

(Refer Slide Time: 0:17)



Deep Learning for Computer Vision

Video Understanding using CNNs and RNNs

Vineeth N Balasubramanian

Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad



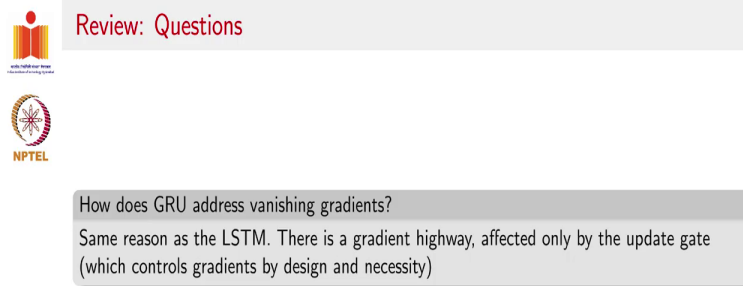
Vineeth N B (IIT-H)

§8.4 Video Understanding

1 / 16

Having seen RNNs which deal with temporal sequences. We will now move to temporal sequences in vision, computer vision, which are videos. So, we will try to see how videos are processed using CNNs and RNNs.

(Refer Slide Time: 00:36)



Review: Questions

How does GRU address vanishing gradients?

Same reason as the LSTM. There is a gradient highway, affected only by the update gate (which controls gradients by design and necessity)



Vineeth N B (IIT-H)

§8.4 Video Understanding

2 / 16

The one question that we left behind in the last lecture was, how do GRUs address vanishing gradients? And the answer is the same as LSTMs. They also have a gradient highway between a hidden state at a particular time step and the hidden state at the previous time step, which was in the final hidden state update equation.

You would notice that that was given by $(1 - z) * (\text{the previous hidden state}) + z * (\text{update gate}) * (\text{the current hidden state})$. So, you would notice that in that equation the previous, the gradient of the current hidden state with respect to the previous hidden state depends only on a gate which does not affect the gradient per say as long as the update gate allows the gradient to pass through and that is by design.

(Refer Slide Time: 01:37)

Why do we need to understand a video?

NPTEL

Credit: Smarter Everyday (Youtube)

Vineeth N B. (IIT-H) 3 / 16

Moving on to videos. Let us first ask the question. Why do we need to understand a video at all? So, here are a couple of examples. So, this is an example of an automated baseball bat hitting a baseball and if we wanted to understand, where would this baseball bat hit the ball or what direction will the ball fly or what kind of an activity is this so on and so forth. One would be to look at all the frames while coming up with the decision and one single frame in the sequence may not let you answer that question for example, as to where the ball will go.

(Refer Slide Time: 02:21)

Why do we need to understand a video?

NPTEL

Credit: Veritasium (Youtube)

Vineeth N B. (IIT-H) 58.4 Video Understanding 4 / 16

This slide features a title bar with the text "Why do we need to understand a video?". Below the title bar are the NPTEL logo and a small video player showing three sequential frames of a screw being turned. At the bottom, there is a small portrait of the speaker and a progress bar indicating the current slide is 4 out of 16.

Another example is if we wanted to understand in which direction this screw was being turned in, one of these frames will not give you the information of which direction the screw was being turned in. You would need the sequence of frames to be able to make this decision.

(Refer Slide Time: 02:44)

Why do we need to understand a video?

NPTEL

How to understand a video?
Let's forget everything we learn't and see if we can figure it out by ourself!





Vineeth N B. (IIT-H) 58.4 Video Understanding 5 / 16

This slide features a title bar with the text "Why do we need to understand a video?". Below the title bar are the NPTEL logo and a video player showing a sequence of frames of a screw being turned. A text box is overlaid on the video player with the text "How to understand a video? Let's forget everything we learn't and see if we can figure it out by ourself!". At the bottom, there is a small portrait of the speaker and a progress bar indicating the current slide is 5 out of 16.

So, the question that we have is, how do we then use whatever we have learned so far. So, we have seen feed forward neural network CNNs and now RNNs. How do we use these to understand a video, can one of them suffice, do we need multiple of them, let us try to see if we can figure this out by ourselves and go ground up.





(Refer Slide Time: 03:06)

How to understand a video?



Vineeth N B. (IIT-H) [§8.4 Video Understanding](#) 6 / 16

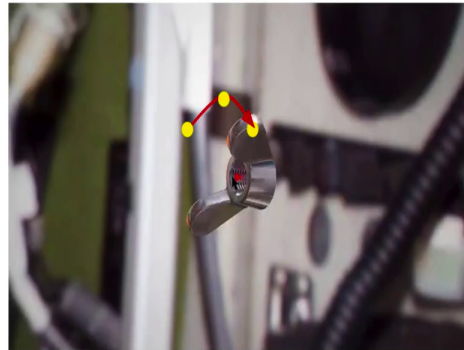
How to understand a video?



Vineeth N B. (IIT-H) [§8.4 Video Understanding](#) 6 / 16



How to understand a video?



Vineeth N B. (IIT-H)

§8.4 Video Understanding

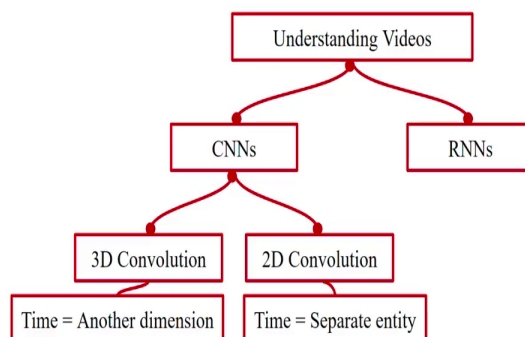
6 / 16

So, if we have to understand a video, you can look at one specific location of, let us, if we took the example of the screw video, you could take one position of that screw and track it over time and then be able to say what direction the screw was turning in. So, how do we get this information? How do we actually track a particular point? And then see where it goes over time?

(Refer Slide Time: 03:37)



How to understand a video?



Vineeth N B. (IIT-H)

§8.4 Video Understanding

7 / 16

There are a few possibilities. Considering the focus of this course is on deep learning, We are going to look at deep learning (ways) approaches to understanding videos. So, broadly speaking one could use only CNNs to understand videos or can use CNNs and RNNs together to understand videos. We will see both examples in this lecture. Even within CNNs there are two

ways to perform video based understanding. One is to do what is known as 3D convolution and the other is to see if you can repurpose 2D convolution itself to understand videos.

In a 3D convolution approach the time becomes another dimension of analysis just like how images are two dimensional signals, videos are three dimensional signals, time becomes another dimension. So, every operation that we saw so far you have to add another dimension in the operation. For example, 3D convolution.

Another way of looking at it is to use 2D convolution but to consider time as a separate entity not a separate dimension by itself and we will see an example of this to make this clear. One question you could have is, were not we doing 3D convolution all the while. The inputs to our CNNs so far were volumes. We had R channel, G channel and B channel.

So, were not we actually already doing convolution over volume, was not that 3D convolution. If you have not thought through that before, not really, the reason why we do not call that 3D convolution is we only moved convolution along the two dimensions of the image. The depth was always fixed, the number of channels that you had, you did not move across that dimension when you performed convolution.

The movement was only along two dimensions and so whatever we saw so far is 2D convolution. Although (we have) the filter extends into the third dimension and we process it, We actually do not move on windows in the third dimension. But now when we talk about videos and 3D convolution here, we are talking about moving in the temporal dimension.

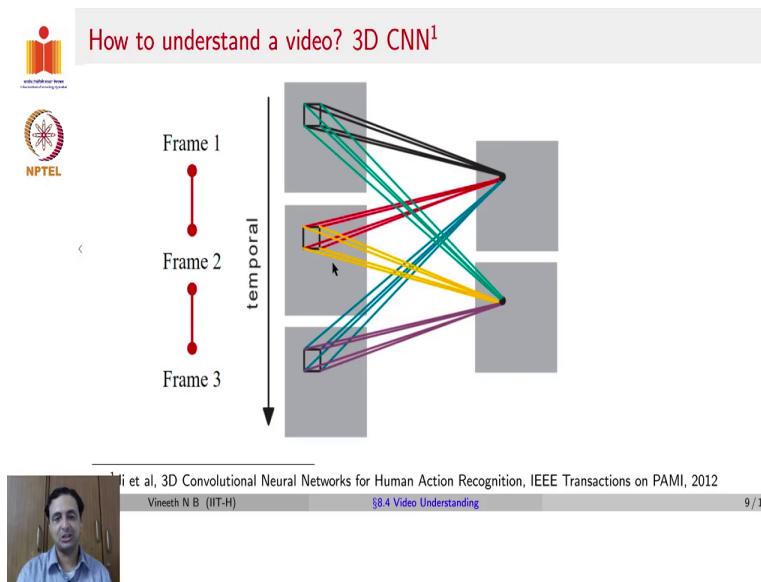
(Refer Slide Time: 06:27)

The slide features the NPTEL logo and the title "How to understand a video? 3D CNN". It contains a diagram with two rows of gray rectangles representing frames. Each row has a white trapezoidal shape on the left and a black dot on the right, connected by a horizontal line. A blue 'X' is placed between the two rows, with a red vertical line passing through it. Labels "Frame 1" and "Frame 2" are positioned to the left of the top and bottom rows respectively. At the bottom left is a small video feed of a man, and at the bottom right is a footer with the text "Vineeth N B. (IIT-H) §8.4 Video Understanding 8 / 16".

Let us try to understand 3D CNNs first as a way to understand videos. Traditionally speaking if you had one frame from a video or more frames, if you performed standard 2D convolution, you would look at a particular patch of each frame, perform convolution and you would get a certain output in the output feature map.

Unfortunately, when we apply convolution this way, the two frames are not connected and you are not sharing any information between the two frames. However, when you do video understanding the core idea is to be able to pass information or combine information from multiple frames to make a prediction if it was a classification problem for instance. So, how do we do this?

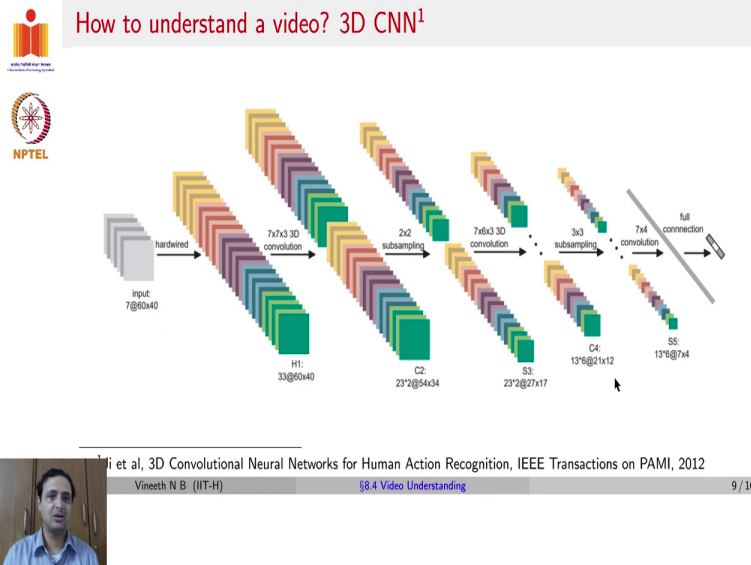
(Refer Slide Time: 07:26)



As I just mentioned, the idea is to extend convolution to a further dimension. So, if you had three frames instead of doing 2D convolution on each frame individually, what we are going to propose is to do convolution across the frames. Let us try to study this more carefully.

So, you can see here that the first patch has a set of weights which gives you a pixel in the output feature map, but that feature map is also influenced by another set of weights that is applied to the second frame and a third set of weights that is applied to the third frame. Similarly, you can see that in a second feature map you would have a different set of weights applied to each frame of your video. So, you have another dimension that you have to convolve on now, which is the temporal dimension.

(Refer Slide Time: 08:28)



So, we are going to look at one specific paper, which was perhaps the initial harbinger that brought on a lot of other work. This was known as 3D convolutional neural networks, which was published in, firstly, ICML of 2010, then in PAMI 2012, and let us look at the architecture first for this 3D CNN, the architecture has evolved over the years, but at least this should give you one example.

In this particular example 3D convolutional neural networks were used for human action recognition to look at a video and predict the action. So, the input was a set of 7 frames whose resolution was 60 x 40. It was just grayscale frames resolution 60 x 40. Those were the 7 frames that were considered. This particular work initially had one layer, where hardwired filters were used.

So, these filters here were not learnt there were five different kinds they had a grayscale filter, they had a gradient along the X direction, they had an optical flow along the X direction and the optical flow along Y direction, they had five different kinds of filters and those gave these sets of images. So, they obtained a total of 33 feature maps each of them still at the resolution 60 x 40, after this initial processing of a few filters that were hardwired. These were not learnt, these were filters that detected edges, detected optical flow so on and so forth.

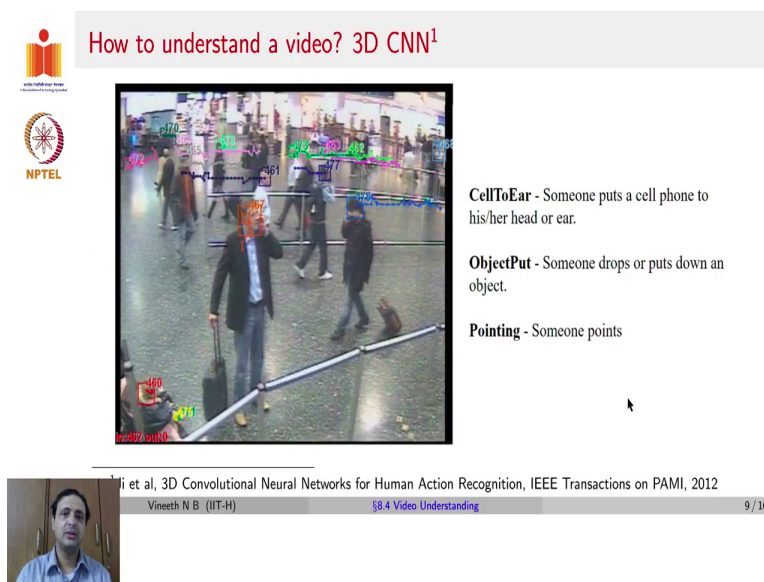
The next layer was 3D convolution which was done using learned weights. They performed 7 x 7 x 3 3D convolution where for each of these specific components as we said they had five

different kinds of feature maps. So, they ensured that they did 3D convolution within each of those five kinds of feature maps and that is what you see colour coded here. You can see these five colours and those five colours are also maintained in subsequent layers.

So, the 3D convolution is within only the yellow region, within only the red region, within only the purple region so on and so forth. And to get some variety, they also introduced a completely different set of feature maps with different set of weights, but performing the same operations. You could consider this similar to your AlexNets 48 feature maps and 48 features maps in the first layer going to two different GPUs. Although in this case the intent was not to send to two different GPUs, the intent was to get more variety in the feature maps. The rest of the layers follow a traditional CNN approach.

They then performed 2 x 2 subsampling within each of these feature maps, then they performed another (3 cross) another 3D convolution, a 7 x 6 x 3 3D convolution which again had two sets of feature maps. Then a 3 x 3 subsampling then a 7 x 4 convolution to bring everything together. Then a fully connected layer to make the final decision. And how are these learnt? Nothing changes from that perspective. You still have cross entropy loss at the end and you still can back propagate through all of these layers very similar to how we talked about for CNNs.

(Refer Slide Time: 12:10)



The slide displays a video frame of a busy street scene with several people. Overlaid on the frame are various colored bounding boxes and labels, such as 'CellToEar', 'ObjectPut', and 'Pointing', indicating detected actions. To the right of the video frame, a list of actions is provided:

- CellToEar** - Someone puts a cell phone to his/her head or ear.
- ObjectPut** - Someone drops or puts down an object.
- Pointing** - Someone points

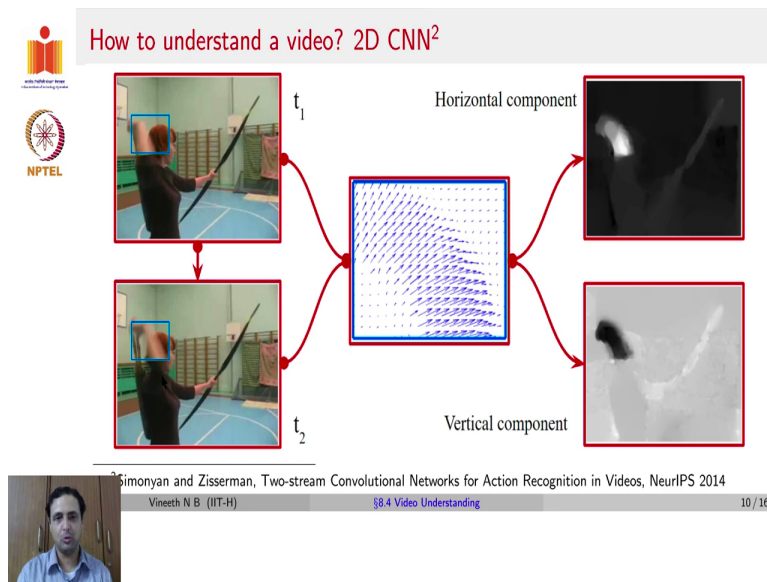
At the bottom of the slide, there is a small video thumbnail of a man speaking, and a footer containing the text: 'S. S. et al, 3D Convolutional Neural Networks for Human Action Recognition, IEEE Transactions on PAMI, 2012', 'Vineeth N.B. (IIT-H)', '58.4 Video Understanding', and '9 / 16'.

Using this kind of an approach they could show that you could perform action recognition, certain actions that they had in the data set was to check if there was a cell to your action where

someone puts a cell phone to the ear or to find an action where somebody left behind an object in a public environment or when someone points to another person.

So, you can see here if you observe carefully that for each of these boxes there is also a trajectory that follows so if you looked at this particular box here, you would see there is also a trajectory which shows the previous frames and the positions in the previous frames that led to this particular outcome on this frame and also a probability or score rather of the outcome. So, this is the 3D CNN approach to understanding videos and there have been several variants over the years in trying to perform such operations.

(Refer Slide Time: 13:13)



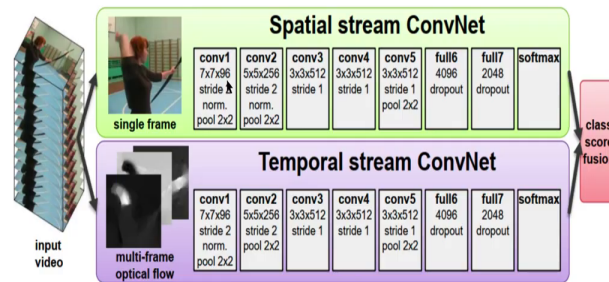
Let us now turn the problem around and ask, could we have used 2D CNN itself to solve a video understanding problem, turns out there is a very popular method that does this known as 2 stream convolutional networks, was very popular, is still popular for doing video understanding tasks.

Let us try to understand how this works. So, if you had a video where at one type step you see a person trying to pull out something from behind their shirt and the hand moves and something comes out from behind their shirt. Now, we could use a traditional optical flow approach to try to find out which pixels moved how much across the X direction and the Y Direction. So, using this optical flow, you could have a horizontal component, you could also have a vertical component of the optical flow. How do we use this?

(Refer Slide Time: 14:14)



How to understand a video? 2D CNN²



Simonyan and Zisserman, Two-stream Convolutional Networks for Action Recognition in Videos, NeurIPS 2014

Vineth N. B. (IIT-H)

58.4 Video Understanding

10 / 16

So, in the 2 stream CNN you have your input video which is a volume. So, you have a spatial stream of the CNN which takes a single frame which could be colour and now sends it through a standard CNN architecture to get your final output of what action this maybe so this has a conv 1, conv 2, conv 3, conv4, conv 5 fully connected, fully connected very similar to an Alex net architecture to get a final classification at the end.

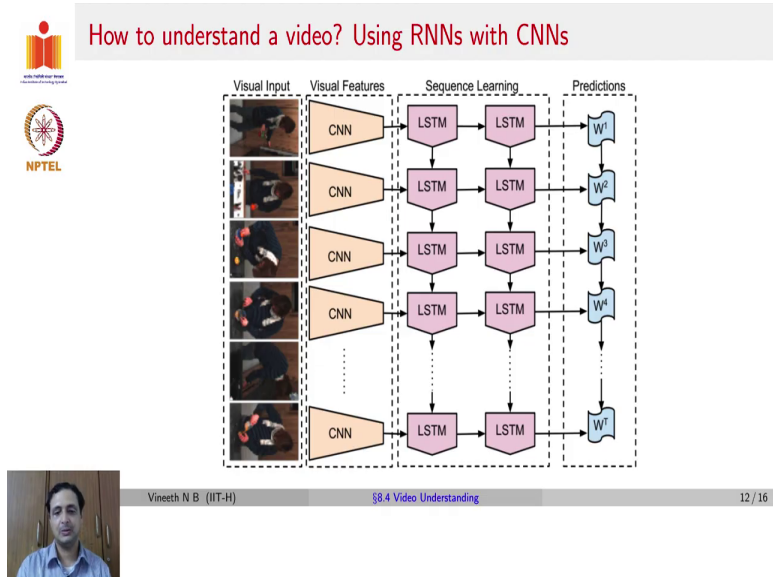
But this is only for a single frame. Now, the other stream, this is known as the spatial stream and the other stream performs a multi frame optical flow of how the content changes across the frames. So, you have multiple frames of the optical flow between every set of consecutive frames across all the frames in the input video.

Now, this set, this optical flow images becomes a set of channels across the temporal dimension and one could now perform a standard convolution, 2D convolution on this volume of optical flows each optical flow image is a 2D image and across time for between every successive frame you can get optical flows and you stack them all together to make a volume and this volume can be sent through another standard 2D CNN architecture, which we would call the temporal stream convolutional network and both of them provide their own classifications on what action this maybe.

And the final step is to perform a fusion of class scores, you can aggregate them in several ways, majority vote them or average them and then make a final decision so on so forth. And this gives

the final decision on what action this is. Note here that all these convolutions are 2D convolutions and not 3D convolutions.

(Refer Slide Time: 16:26)



What about using RNNs with CNNs to perform video understanding. So, given a visual input such as an image, you first extract visual features from a CNN. So, you would forward propagate that through a CNN and take, say the penultimate layer of AlexNet or something like that as the representation of the image. Now, these representations of the image can be provided to a set of LSTMs or RNNs to perform sequence learning to give your final predictions.

So, if you had a video where you have multiple frames, then you provide each frame to a corresponding CNN to get different feature representations, and then they can then be provided to an LSTM which gives, so you could assume now that each column here is your LSTM unfolded over time and this is a staged LSTM unfolded over time and finally the output of these LSTMs are the predictions that you make. So, you could have a prediction per frame or you could ignore all of these to make a prediction only at the last time step. That would be the way to combine CNNs with RNNs.

How do you learn such a network? Remember that every component that we see here is back probable, we know how to back propagate across an RNN, an LSTM, a CNN. So, given a cross entropy loss at the end or a sum of cross entropy losses for each time step. We could compute the

back propagated error, the gradient of that error with respect to every weight in the CNN or the LSTM. The CNN is the same model that is used on each of these images.

(Refer Slide Time: 18:27)

What can be done?

Train - UCF101

Spatial stream ConvNet

Temporal stream ConvNet

Action Recognition

Baseball Pitch, Basketball Shooting, Bench Press, Biking, Billiards Shot, Boccia/Under-Curl, Clean and Jerk, Diving, Dressing, Feinting, Golf Swing, High Jump, Horse Race, Horse Riding, Hula Hoop, Judo/Thrust, Juggling Ball, Jumping Jack, Jump Rope, Karate-kyu, Lunge, Military Parade, Moving Heavy, New Year's, Pista Training, Playing Cards, Playing Piano, Playing Table, Playing Tennis, Pile Push, Rowing/Canoe, Pull Ups, Push, Push Ups, Rock Climbing Indoor, Rope Climbing, Rowing, Salsa Salsa, Skateboarding, Skiing, Skijet, Soccer Juggling, Soccer Kick, Tennis Swing, Throw Discus, Trampoline Jumping, Volleyball Spiking, Walking with a dog, Yo Yo
Apply Eye Makeup, Apply Lipstick, Archery, Baby Crying, Balance Beam, Band Sawing, Baseball Bunt, Blow Drying Hair, Blowing Bubbles, Body Weight Squats, Bowling/Bowling Bowling Ball, Bowling Spare Ball, Bowling Trick, CDF Diving, Cider Bowling, Cricket Shot, Cutting in Kitchen, Field Hockey Penalty, Floor Gymnastics, Football Catch, Four Cross, Hair cut, Hammering, Hand saw-Flower, Handstand Pushup, Handstand Walking, Head Massage, Ice Dancing, Kaiting, Long Jump, Mapping Floor, Model Run, Posing Couch, Posing Duet, Posing Ideal, Posing Floor, Posing Shoes, Rhythmic, Shooting Bowls, Shot put, Sky Diving, Soccer Penalty, Soft Throw, Sume Wrestling, Tackle, Taki Tami Shot, Tying, Uncont. Bunt, Wall Pushup, Writing On Board

Soomro et al, UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild, 2012

Vineeth N B. (IIT-H) 58.4 Video Understanding 13 / 16

Now, with these approaches, be the two stream CNN or a CNN with RNN or a 3D CNN, What kind of problems in video understanding can we solve? So, if you took a data set such as the UCF101 which is a very popular data set for video understanding, you could perform action recognition like, is the person doing push-ups, rock climbing, is the person trampoline jumping so on and so forth.

(Refer Slide Time: 19:00)

What all can be done?

Train - Hollywood in Homes

Action Recognition

- Reading a book 13%
- Smiling 9%
- Holding a book 9%
- Using a book 7%
- Laughing 5%
- Smiling 12%
- Laughing 9%
- Using a book 9%
- Using a book 9%
- Using a book 9%

Sentence Prediction

- Cell A person is standing in the kitchen looking on a stove they then take a drink from a glass and drink it.
- Cell B person is standing in the doorway holding a pillow the person then gets up and walks to the door and sits down and drinks it.
- Cell C A person is lying on a bed with a blanket the person then gets up and walks to the door and sits down and goes back to sleep.
- Cell D A person is standing in the doorway drinking coffee before grabbing a towel from the closet and hanging it out the door.
- Cell E A person is walking up and turns a light on and off before going back to sleep.

Crundson et al, Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding, ECCV 2016

Vineeth N B (IIT-H) 8.4 Video Understanding 14 / 16

Or if you had a different kind of a data set, another dataset that is popular is known as Hollywood in homes, which is an activity understanding dataset again. In this you could perform action recognition such as smiling at a book so on and so forth or you could also perform sentence prediction such as caption generation for a full video or equivalent problems.

(Refer Slide Time: 19:29)

Other Tasks in Video Understanding

- Action Forecasting
- Object Tracking
- Dynamic scene understanding
- Temporal Action Segmentation
- ...

Vineeth N B (IIT-H) 8.4 Video Understanding 15 / 16

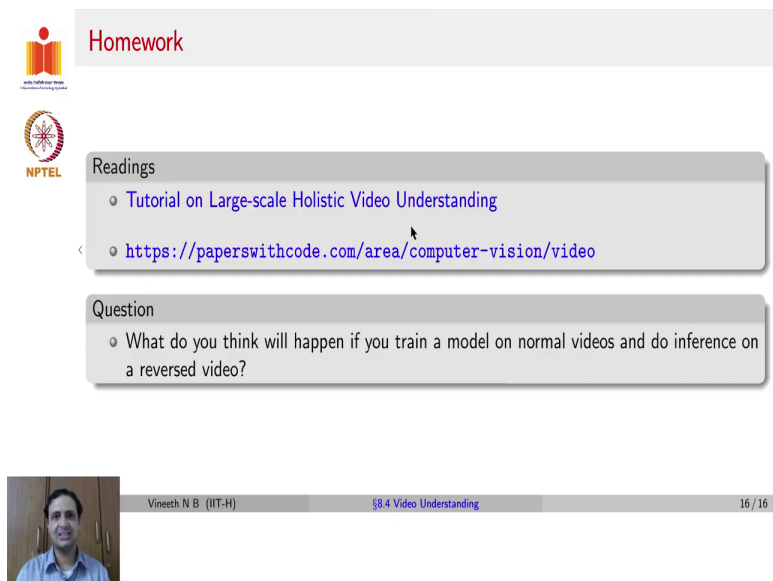
There are more problems that can be solved for video understanding, Action forecasting. So, if you had a sports video and you see a particular sequence of events so far, what would happen next. For example, if you had a water polo game and you had a sequence of moves that you have

seen so far, where will the person pass the ball to next or in basketball for instance? Object tracking is another important video understanding task.

Dynamic scene understanding, so far we have seen methods that, given a scene, can classify the scene into outdoor or indoor, say morning or night, urban or rural so on and so forth using image level classification methods. We know a CNN, you know a cross entropy loss given data, we can learn a scene classification model.

But what happens if you want to understand a dynamic scene? a dynamic video? Then we have to switch to video understanding models such as a 3D CNN or two stream CNN or a CNN plus RNN to solve such a problem. One could also think of other problems such as temporal action segmentation so on and so forth which would be an unsupervised kind of an approach which are also important problems in video understanding.

(Refer Slide Time: 20:55)



The screenshot shows a slide with the following content:

- Homework**
- Readings**
 - [Tutorial on Large-scale Holistic Video Understanding](#)
 - <https://paperswithcode.com/area/computer-vision/video>
- Question**
 - What do you think will happen if you train a model on normal videos and do inference on a reversed video?

At the bottom of the slide, there is a small video feed of a man, the name 'Vineth N B (IIT-H)', the title '§8.4 Video Understanding', and the page number '16 / 16'.

Your homework would be to get, to go through this tutorial on large scale holistic video understanding, which should give you a clearer picture of more tasks in video understanding and how these architectures can be adapted to solve these kinds of problems and this particular link also gives a good listing of papers in the space.

We will conclude the lectures for this week with a very interesting question, which is, what do you think will happen if you train a model on normal videos and now do inference at test time

you give a reversed video? Would it work? Would it not Work? Think about it and we will discuss this the next time.