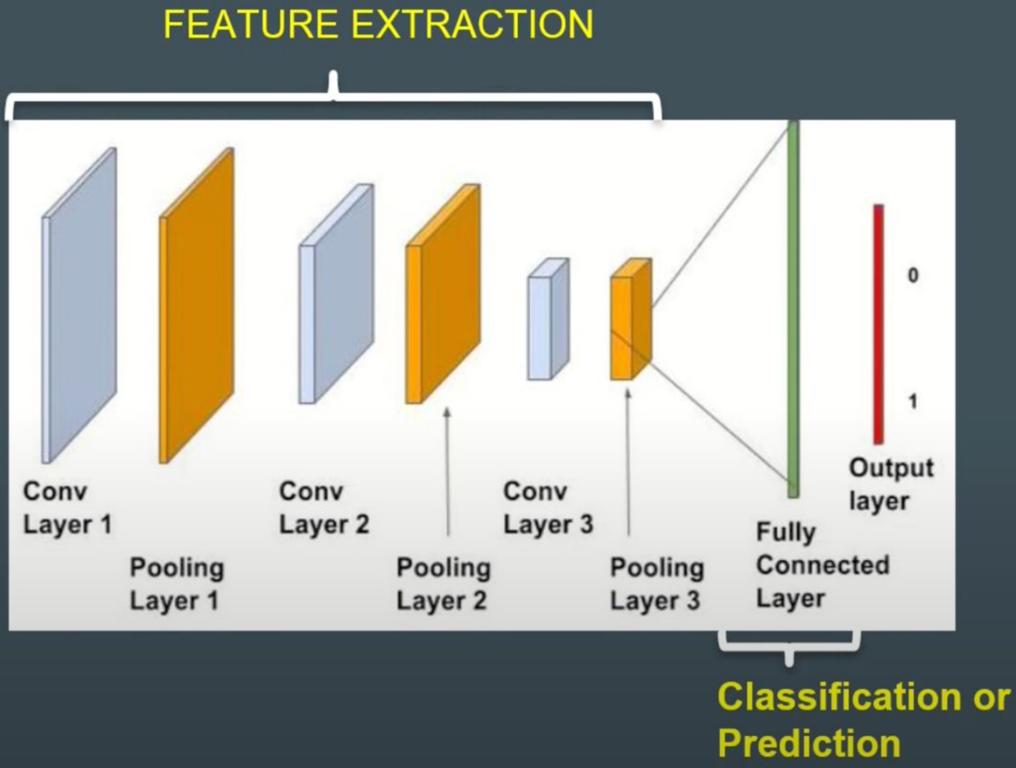
Transfer learning and Fine-tuning

A Recap – This is what has happened in CNN



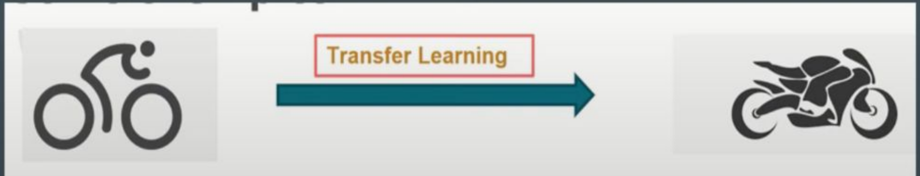
Let's come to the point – What's transfer learning?

Using a Pre-Trained Model

- Can you make it simple?

- Transfer learning make use of the knowledge gained while solving one problem and applying it to a different but related problem

We have ability to use our knowledge in one area to another with ease. Our knowledge acquired for one task can be used for solving other related tasks



If you know riding cycle, that knowledge can be used for Bike driving!

Which datasets/problems can we apply transfer learning?

Where do we get these models which we transfer knowledge from?

Imagenet Dataset & ILSVRC challenge

Based on statistics about the dataset recorded on the ImageNet homepage, there are a little more than 14 million images in the dataset, a little more than 21 thousand groups or classes (synsets), and a little more than 1 million images that have bounding box annotations (e.g. boxes around identified objects in the images).

The ImageNet Large Scale Visual Recognition Challenge or ILSVRC for short is an annual competition helped between 2010 and 2017 in which challenge tasks use subsets of the ImageNet dataset.

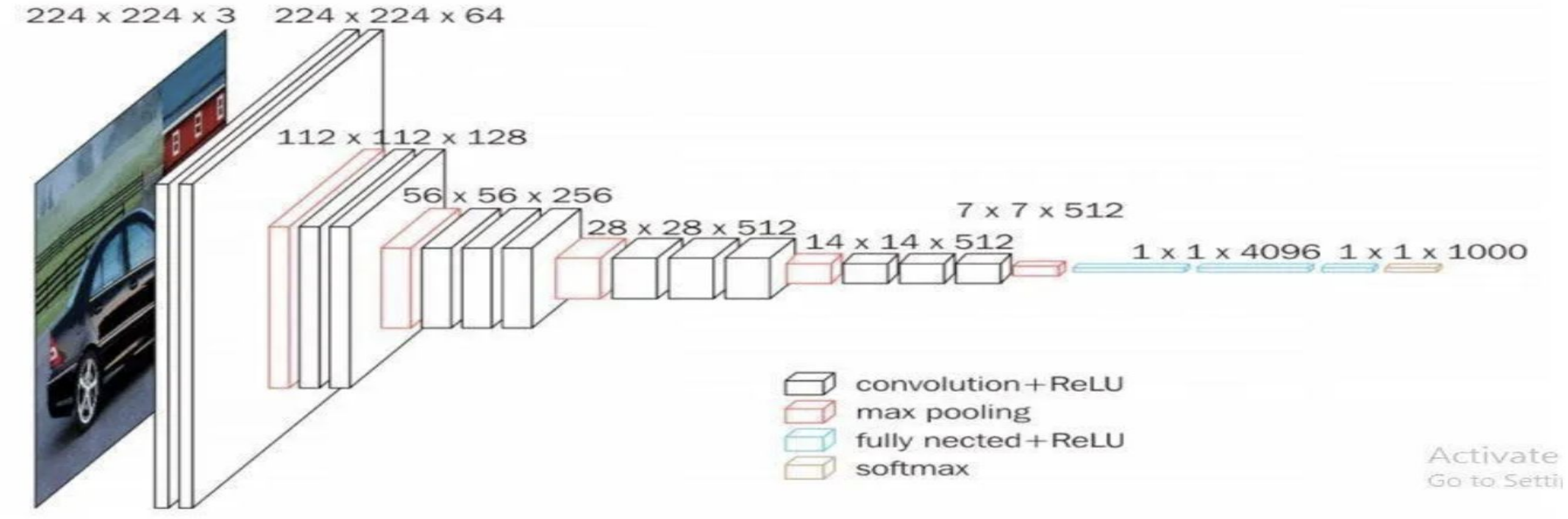
The goal of the challenge was to both promote the development of better computer vision techniques and to benchmark the state of the art.

The annual challenge focuses on multiple tasks for “image classification” that includes both assigning a class label to an image based on the main object in the photograph and “object detection” that involves localizing objects within the photograph.

Top models in the challenges

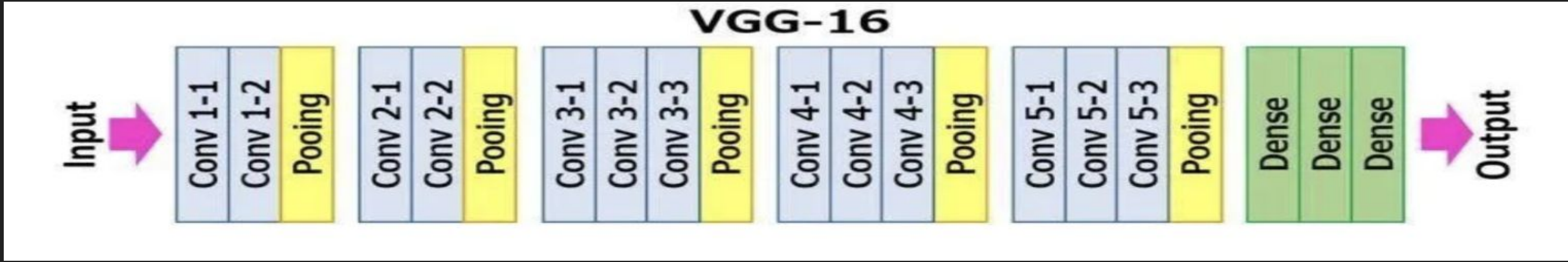
Model	Year	Top-5 Error Rate	Number of Parameters
AlexNet	2012	15.3%	~60 million
ZFNet	2013	11.2%	~60 million
Inception V1 (GoogLeNet)	2014	6.67%	~6 million
VGG-16	2014	7.3%	~138 million
ResNet	2015	3.57%	~25 million
ResNeXt-10	2016	4.1%	~6 million
PNASNet-5	2018	3.8% (Bonus)	~86 million

VGG16 Architecture

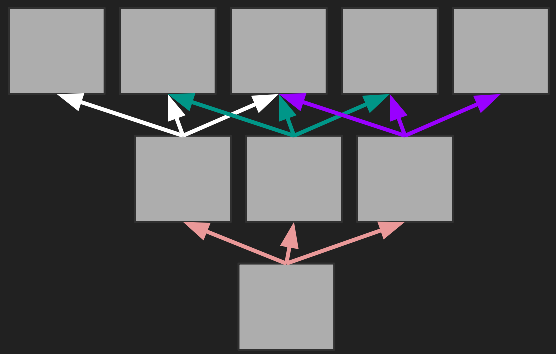


Vggnet-16

- The 16 in VGG16 refers to 16 layers that have weights. In VGG16 there are thirteen convolutional layers, five Max Pooling layers, and three Dense layers which sum up to 21 layers but it has only sixteen weight layers i.e., learnable parameters layer.
- VGG16 takes input tensor size as 224, 244 with 3 RGB channel
- Most unique thing about VGG16 is that instead of having a large number of hyper-parameters they focused on having convolution layers of 3x3 filter with stride 1 and always used the same padding and maxpool layer of 2x2 filter of stride 2.



Advantages of Vggnet



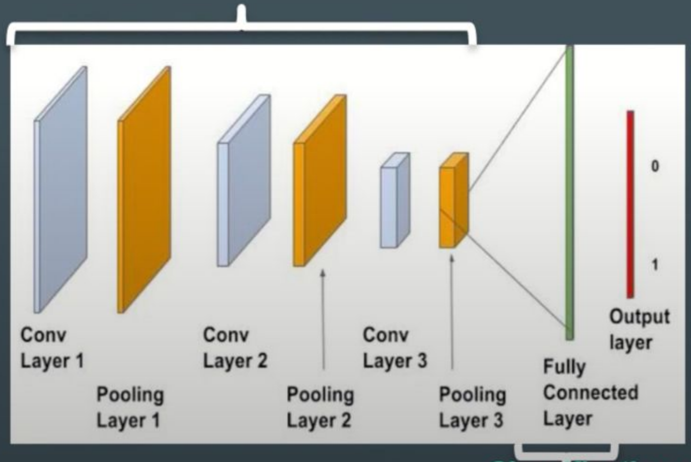
It uses 3*3 filters:

1. two 3*3 convolution filter is equivalent to one 5*5 convolution filter.
2. two 3*3 convolution filter will have less parameters than one 5*5 convolution filter.
3. two 3*3 convolution filter will make network more deep and extract more complex features than one 5*5 convolution filter.
4. Using stacked layers increases discriminative power as we multi non linearities instead of single one

Well, let's make it better

- What if we remove the last layer alone and replace the same with some more probabilities than binary classification?
- Shall we take all the knowledge that existing model has trained on the Neural net and apply it to our application!
- To make it simple, We will change just the classification / Prediction layer and change the weights according to or application Keeping the "feature extraction" part untouched.
- Can we apply more technical terms to our explanation:
- We shall retrain just the classification part and freeze the other feature extraction layers in our network

FEATURE EXTRACTION



Classification or Prediction

Let’s get deeper

- That is - We shall freeze all the convolutional and max-pooling layers -so that they do not modify their weights, leaving only the fully connected ones free.
-
- Retraining is done only for the fully connected layer/(s)
- That is., we change only the classifier part to work for our application.
- So what do we do in Transfer learning?
- We take advantage of the knowledge of another problem to solve the one we are dealing with by taking advantage of the feature extraction stage and fine tuning the classifier part alone
- What’s fine tuning?.. You never mentioned – Yes, we have to do that now!

Can we fine tune now?

- There is a slight difference in “Fine Tuning” and “Transfer Learning”

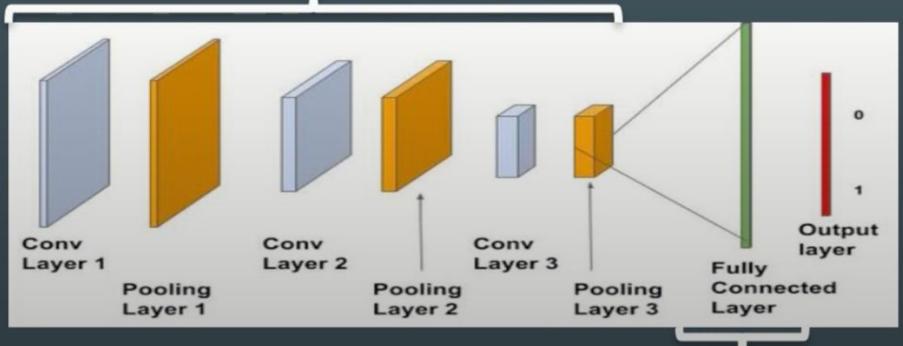
With fine-tuning,

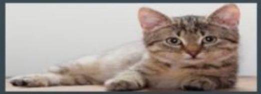
- We initially change the last layer to match the classes in our dataset [same as transfer learning]

Then what’s the difference ....

- Yes here it’s more flexible so that , we can also retrain the layers of the network in feature extraction part as we desire to perform

FEATURE EXTRACTION





Can we fine tune now?

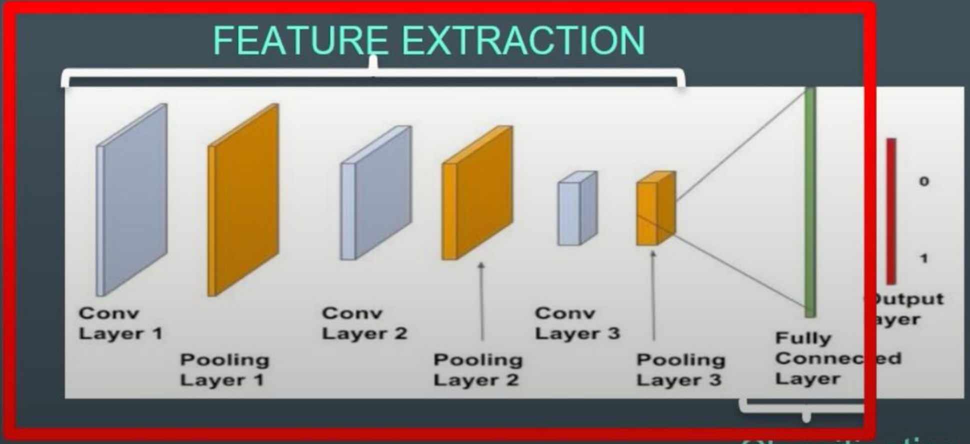
- There is a slight difference in “Fine Tuning” and “Transfer Learning”

With fine-tuning,

- We initially change the last layer to match the classes in our dataset [same as transfer learning]

Then what’s the difference ....

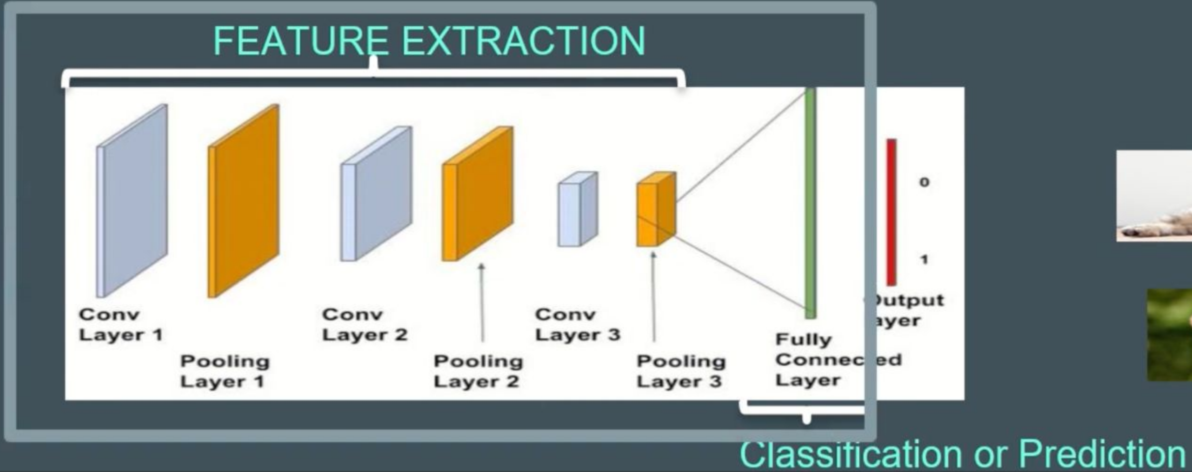
- Yes here it’s more flexible so that , we can also retrain the layers of the network in feature extraction part as we desire to perform



Contd.,

Re train not only the classifier layer but also retrain the feature extraction stage (the convolutional and pooling layers)

- As the first layers detect simpler and more general patterns, and the more we go deeper through the layers ,more details specific to the dataset and the more complicated patterns is detected, hence better allow the last block of convolution and pooling layers to be retrained.



So? Which one to choose?

- Now that we know what is fine tuning and transfer learning, another major question is
- “How to choose whether to use TL or FT?
- Answer is simple...
- Start with Transfer Learning... if needed move to Fine tuning other detailed layers.
- We can get this better now!

Problem vs. ideal choice

- The new dataset is small and similar to the original one
- The new dataset is large and similar to the original
- The new dataset is big and very different from the original

- Transfer Learning preferred , if fine tuning – a chance for Overfitting exists
- Fine Tuning is preferred
- Train from scratch using suitable methods