BITS Pilani
Pilani | Dubai | Goa | Hyderabad

# Deep Learning

## Module # 5 : Convolutional Neural Network [CNN]

primitive

Conv Layer
pooling
Fc layer

# LENET

- Every convolutional layer includes three parts: convolution, pooling, and nonlinear activation functions.
- Using convolution to extract spatial features.
- Conv filters were 5x5, applied at stride 1.
- Subsampling average pooling layer. Subsampling (Pooling) layers were 2x2 applied at stride 2.
- tanh activation function.
- Using MLP as the last classifier.
- Architecture is [CONV-POOL-CONV-POOL-FC-FC]

CNN

LENET
RESNET
ALEXNET
VGG-16
Google net / Inception net

Use → ?

Transfer learning

Image data set → CNN → ?

# LeNet-5

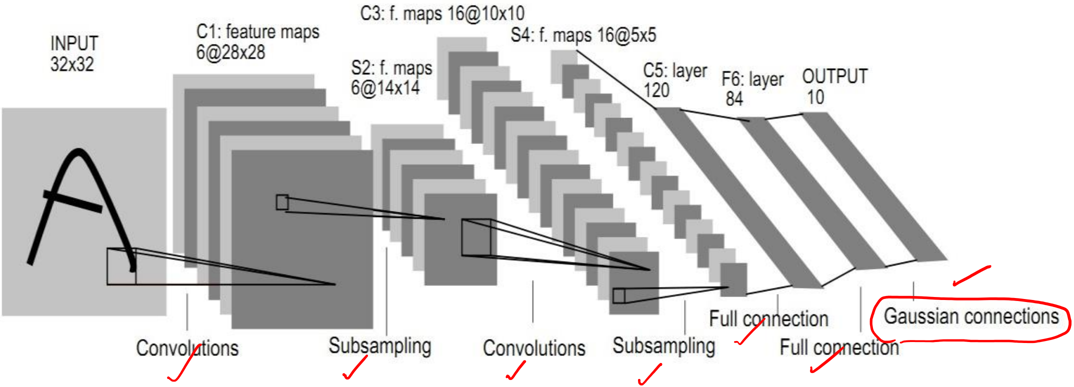
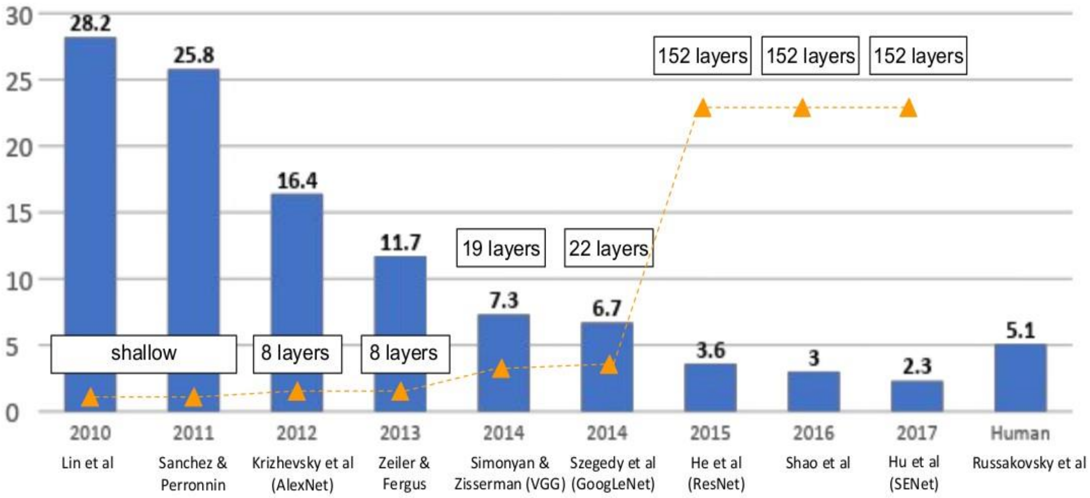



Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

# ImageNet Large Scale Visual Recognition



# IMAGENET DATA

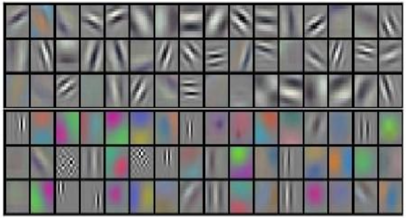


Figure 3: 96 convolutional kernels of size 11*11*3 learned by the first convolutional layer on the 224*224*3 input images. The top 48 kernels were learned on GPU 1 while the bottom 48 kernels were learned on GPU

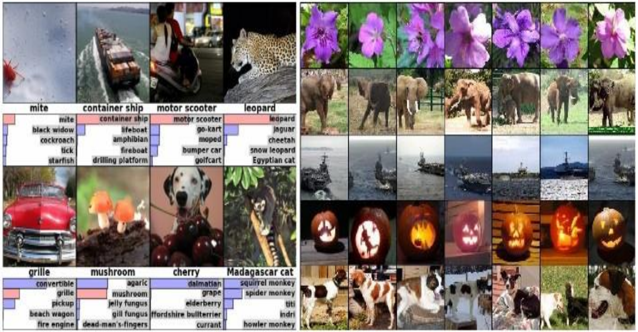
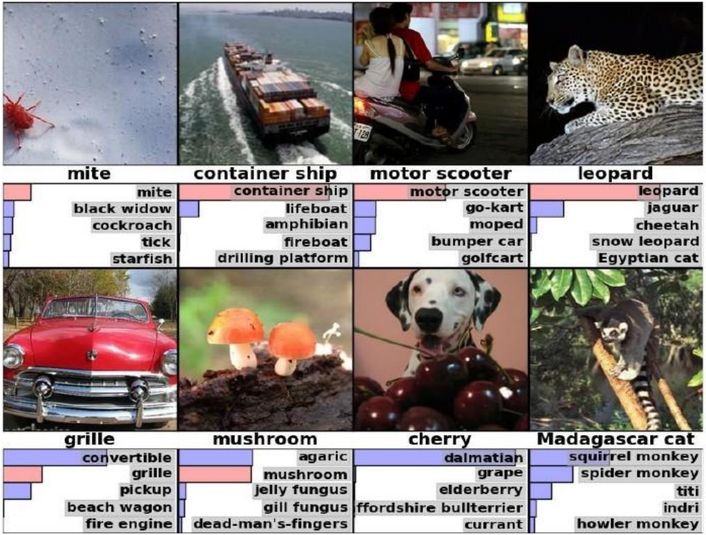



Figure 4: **(Left)** Eight ILSVRC-2010 test images and the five labels considered most probable by our model. The correct label is written under each image, and the probability assigned to the correct label is also shown with a red bar (if it happens to be in the top 5). **(Right)** Five ILSVRC-2010 test images in the first column. The remaining columns show the six training images that produce feature vectors in the last hidden layer with the smallest Euclidean distance from the feature vector for the test image.

# IMAGENET DATA




The ImageNet set that was used has ~1.2 million images and 1000 classes

Accuracy is measured as top-5 performance: Correct prediction if the true label matches one of the top 5 predictions of the model

# ALEXNET

- AlexNet competed in the ImageNet Large Scale Visual Recognition Challenge on September 30, 2012.[3] The network achieved a top-5 error of 15.3%, more than 10.8 percentage points lower than that of the runner up.
- Details/Retrospectives:
  - first use of ReLU
  - used Local Response Normalisation (LRN) layers (not common anymore)
  - Approx. 60 million parameters are learned.
  - heavy data augmentation
  - dropout 0.5
  - batch size 128
  - SGD Momentum 0.9
  - Learning rate 1$e$ −2, reduced by 10 manually
  - L2 weight decay 5$e$ −4
  - 7 CNN ensemble: 18.2%−>15.4%

Extract the feature

FC layer

Forward prop
Back prop

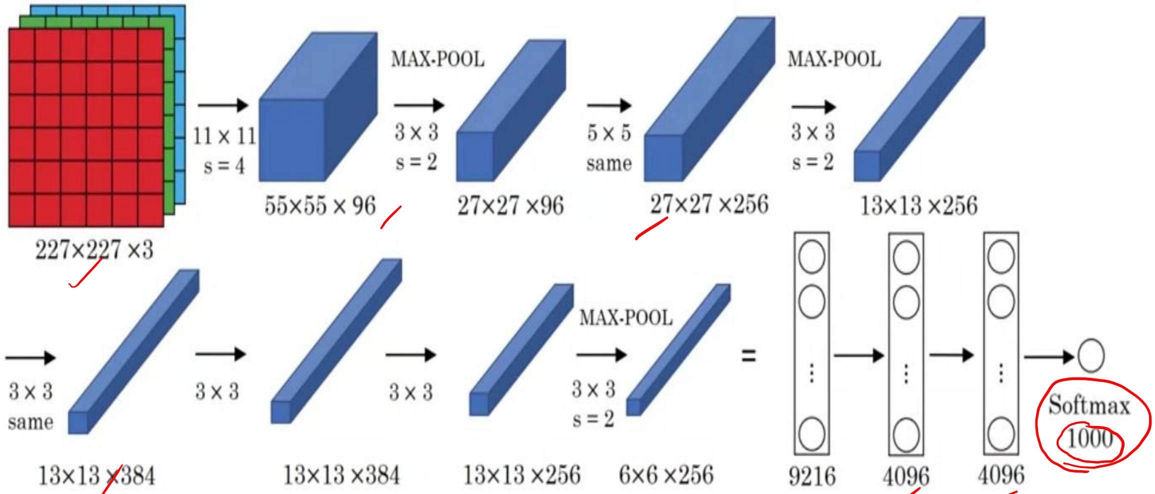
ANN

Fei-Fei Li, Jiajun Wu, Ruohan Gao

# AlexNet Architecture

- 227x227x3 INPUT
- 55x55x96 CONV1 : 96 11x11 filters at stride 4, pad 0 27x27x96
- MAX POOL1 : 3x3 filters at stride 2 27x27x96
- NORM1 : Normalization layer 27x27x256
- CONV2 : 256 5x5 filters at stride 1, pad 2 13x13x256
- MAX POOL2 : 3x3 filters at stride 2 13x13x256
- NORM2 : Normalization layer 13x13x384
- CONV3 : 384 3x3 filters at stride 1, pad 1 13x13x384
- CONV4 : 384 3x3 filters at stride 1, pad 1 13x13x256
- CONV5 : 256 3x3 filters at stride 1, pad 1 6x6x256
- MAX POOL3 : 3x3 filters at stride 2
- 4096 FC6 : 4096 neurons
- 4096 FC7 : 4096 neurons
- 1000 FC8 : 1000 neurons (class scores)

# ALEXNET




[Krizhevsky et al., 2012. ImageNet classification with deep convolutional neural networks]

Andrew Ng

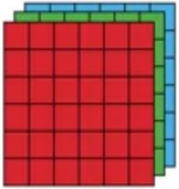
# VGG 16

- VGG stands for Visual Geometry Group with 16 Layers
- Plug and play in Caffe
- Deeper the better
- Details
  - ILSVRC’14 2nd in classification, 1st in localization
  - Similar training procedure as Krizhevsky 2012
  - Approx. 138 million parameters are learned.
  - No Local Response Normalisation (LRN)
  - Use VGG16 or VGG19 (VGG19 only slightly better, more memory)
  - Use ensembles for best results
  - All convolutions with a $3 \times 3$ kernel
  - All max-pooling layers with a $2 \times 2$ kernel
  - FC7 features generalize well to other tasks

Fei-Fei Li, Jiajun Wu, Ruohan Gao

# VGG 16

CONV = 3×3 filter, s = 1, same    MAX-POOL = 2×2 , s = 2



224×224 ×3

→ [CONV 64] ×2 → 224×224×64 → POOL → 112×112 ×64 → [CONV 128] ×2 → 112×112 ×128 → POOL → 56×56 ×128

→ [CONV 256] ×3 → 56×56 ×256 → POOL → 28×28 ×256 → [CONV 512] × 3 → 28×28 ×512 → POOL → 14×14×512

→ [CONV 512] ×3 → 14×14 ×512 → POOL → 7×7×512 → FC 4096 → FC 4096 → Softmax 1000

1000

problem

VGG16

own Archite

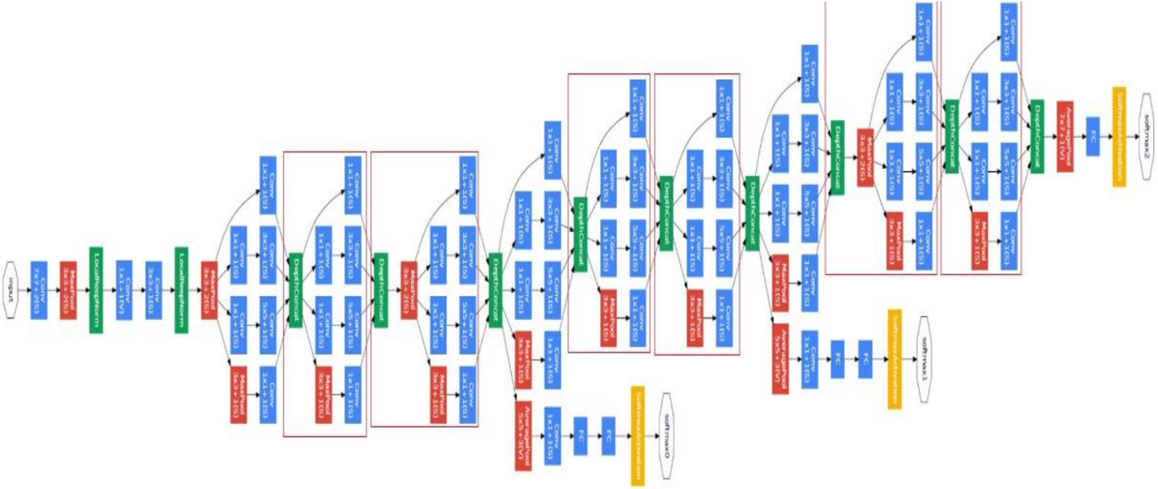
[Simonyan & Zisserman 2015. Very deep convolutional networks for large-scale image recognition]

Andrew Ng

# GoogLeNet / Inception Net

- ILSVRC'14 classification winner (6.7% top 5 error) 22 layers with weights
- Only 5 million parameters (12x less than AlexNet and 27x less than VGG-16)
- Inception Module - convolutional "blocks" – efficient
  - Design a good local network topology (network within a network) and then stack these modules on top of each other.
- Linear layers at the end
- Max pooling in between, multiple Conv layers between pooling
- Great ideas for data augmentation
- Deeper networks, with computational efficiency
- No FC layers.
- After the last convolutional layer, a global average pooling layer is used that spatially averages across each feature map, before final FC layer.

# INCEPTION NET



Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014
Razavian et al, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition", CVPR Workshops 2014

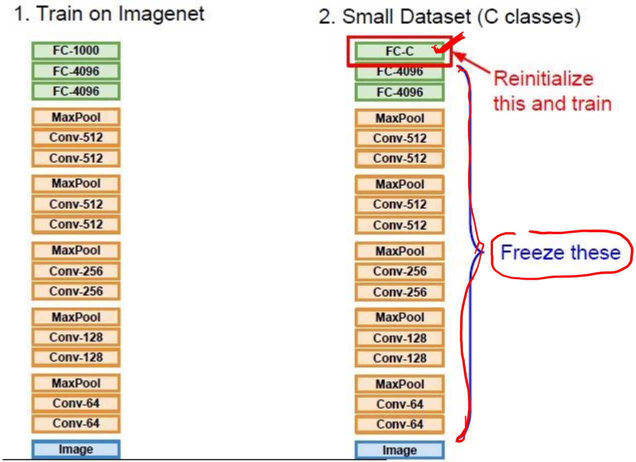
# Transfer Learning with CNNs

1. Train on Imagenet

# TRANSFER LEARNING

Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014
Razavian et al, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition", CVPR Workshops 2014

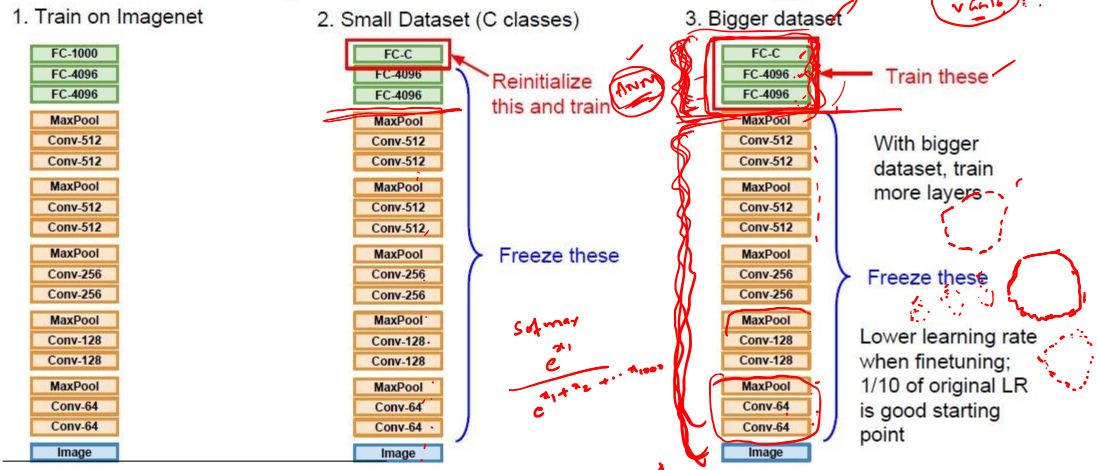
## Transfer Learning with CNNs



# Transfer Learning

## Transfer Learning with CNNs

Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014
Razavian et al, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition", CVPR Workshops 2014



# Transfer Learning for Image Data

- Use a deep learning model that is pre-trained on large dataset like ImageNet or MS Coco.
- Oxford VGG Model
- Google Inception
- Model Microsoft
- ResNet Model

# TRANSFER LEARNING FOR TEXT DATA

- Embedding is the mapping of words to a high-dimensional continuous vector space where different words with similar meanings have similar vector representations.
- Google's word2vec Model
- Stanford's GloVe Model
- FastText
- Gensim

# Transfer Learning - when to use?

- You need a lot of a data if you want to train/use CNNs / RNNs.
- Task A and Task B have the same type of input. Eg: Input is images for both tasks.
- We have lot of data for training Task A and relatively low data for training in Task B.
- Low level features obtained from Task A could be more helpful for learning Task B.

# Thank You!