

AI Lab

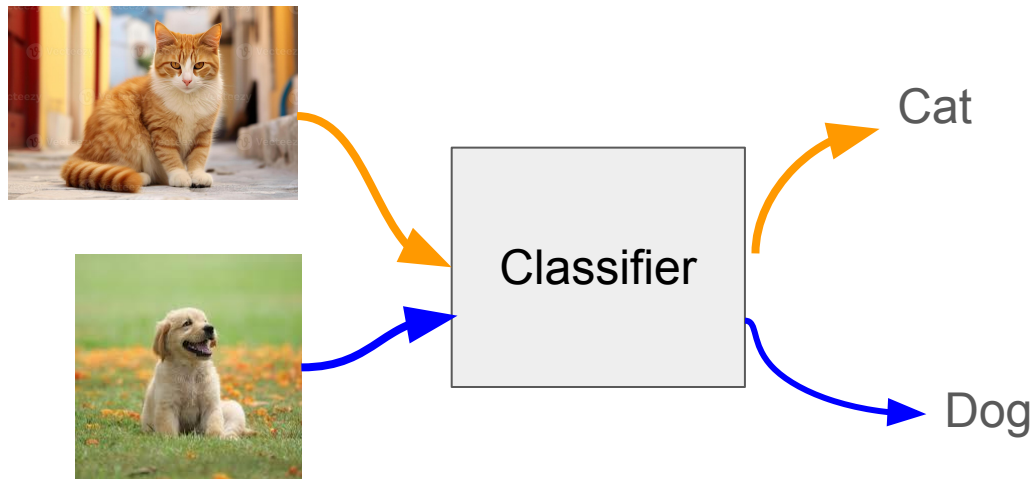
Lecture: 28.11.2024



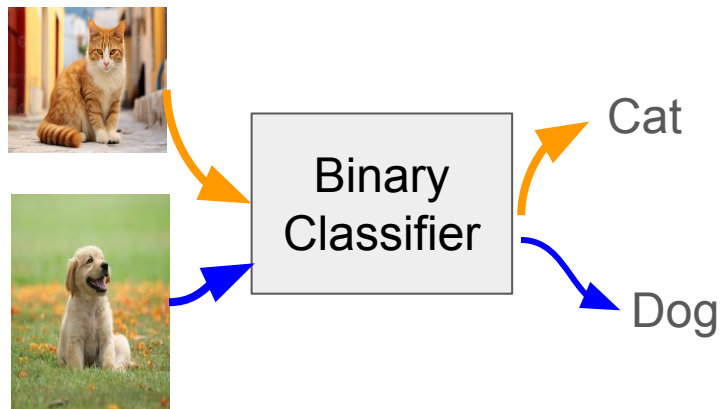
Sangeeta Biswas, Ph.D.
Associate Professor,
University of Rajshahi, Rajshahi-6205, Bangladesh

Classification

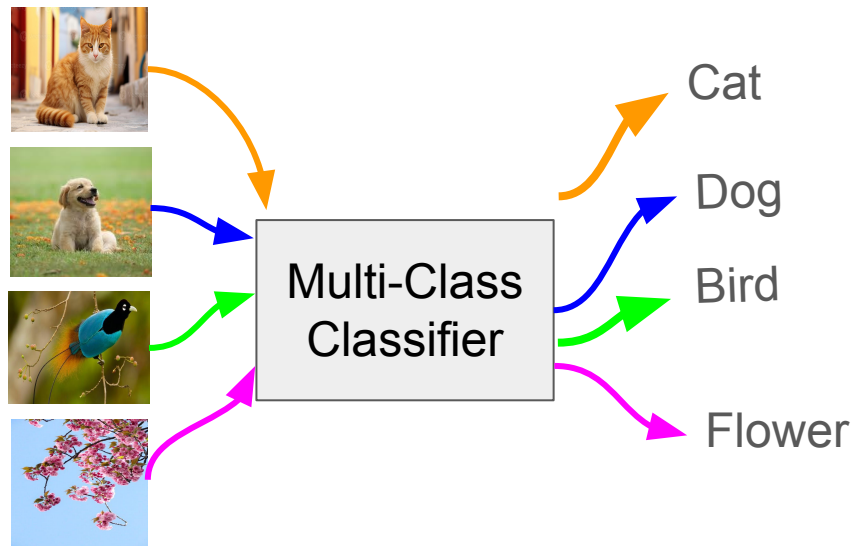
- Assigning objects to some pre-existing classes / categories / labels / groups.



Binary-Class Vs. Multi-Class Classification



Number of classes = 2



Number of classes = 4

Regression

Regression is a problem of predicting a continuous numeric value (e.g., a price, a temperature, a score) based on input data.

Regression



What will be the temperature tomorrow?

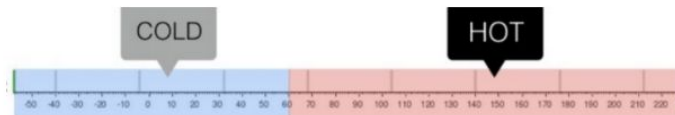


Fahrenheit

Classification



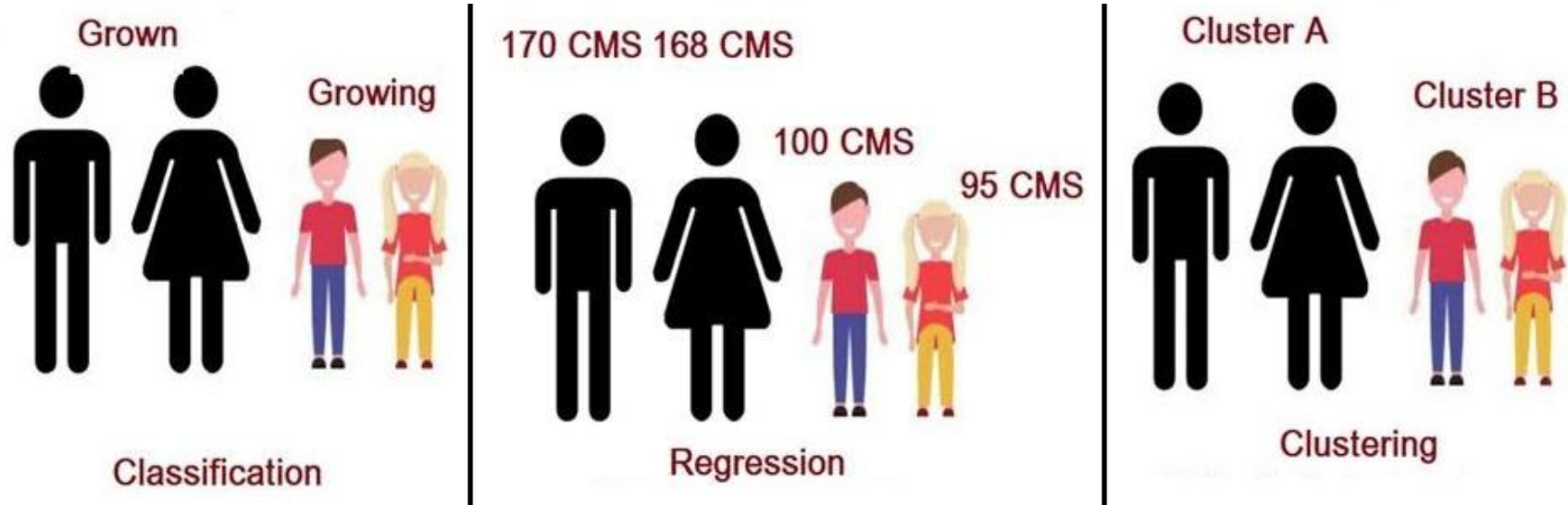
Will it be hot or cold tomorrow?



Fahrenheit

Clustering

Clustering is an unsupervised machine learning technique designed to group unlabeled examples based on their similarity to each other.



Fully Connected Neural Network (FCNN)

FCNN consists of a series of fully connected layers

It is also known as multi-layer perceptron (MLP).

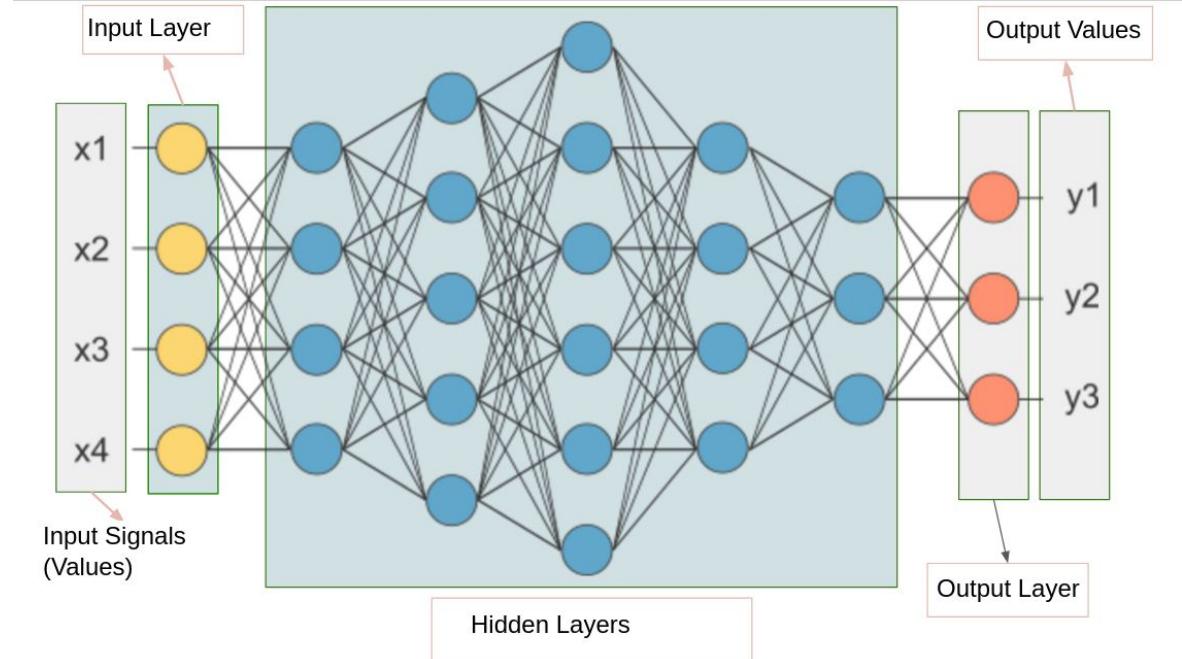


Image Source: Google Search Engine

Convolutional Neural Network (CNN)

- CNN mainly consists of a series of convolutional layers and downsampling layers.
- For classification, fully connected layers are added after convolutional layers.

Popular CNNs:

- * VGG
- * InceptionNet
- * ResNet
- * EfficientNet
- * MobileNet
- * XceptionNet
- * DenseNet

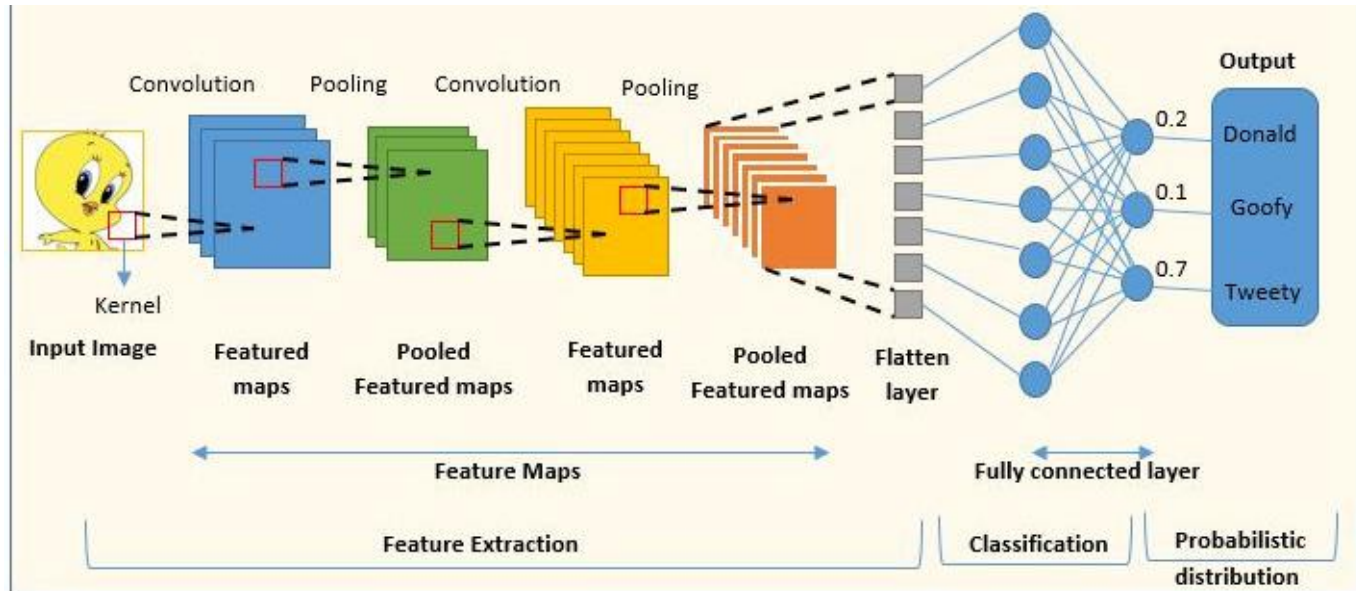


Image Source: Google Search Engine

Pre-trained Classifiers Provided by Keras API Team

- Models were trained by ImageNet 1K dataset for 1000 classes

<https://keras.io/api/applications/>

- We mainly depend on these pre-trained models

Model	Size (MB)	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth	Time (ms) per inference step (CPU)	Time (ms) per inference step (GPU)
Xception	88	79.0%	94.5%	22.9M	81	109.4	8.1
VGG16	528	71.3%	90.1%	138.4M	16	69.5	4.2
VGG19	549	71.3%	90.0%	143.7M	19	84.8	4.4
ResNet50	98	74.9%	92.1%	25.6M	107	58.2	4.6
ResNet50V2	98	76.0%	93.0%	25.6M	103	45.6	4.4
ResNet101	171	76.4%	92.8%	44.7M	209	89.6	5.2
ResNet101V2	171	77.2%	93.8%	44.7M	205	72.7	5.4
ResNet152	232	76.6%	93.1%	60.4M	311	127.4	6.5
ResNet152V2	232	78.0%	94.2%	60.4M	307	107.5	6.6
InceptionV3	92	77.9%	93.7%	23.9M	189	42.2	6.9
InceptionResNetV2	215	80.3%	95.3%	55.9M	449	130.2	10.0
MobileNet	16	70.4%	89.5%	4.3M	55	22.6	3.4
MobileNetV2	14	71.3%	90.1%	3.5M	105	25.9	3.8
DenseNet121	33	75.0%	92.3%	8.1M	242	77.1	5.4
DenseNet169	57	76.2%	93.2%	14.3M	338	96.4	6.3

Plot Model Architecture

```
from tensorflow.keras.applications import vgg16
from tensorflow.keras.utils import plot_model
vgg16_model = vgg16.VGG16()
plot_model(
    vgg16_model, to_file = "VGG16.png", show_shapes = False,
    show_dtype = False, show_layer_names = False,
    show_layer_activations = False, show_trainable = False
)
```

VGG16

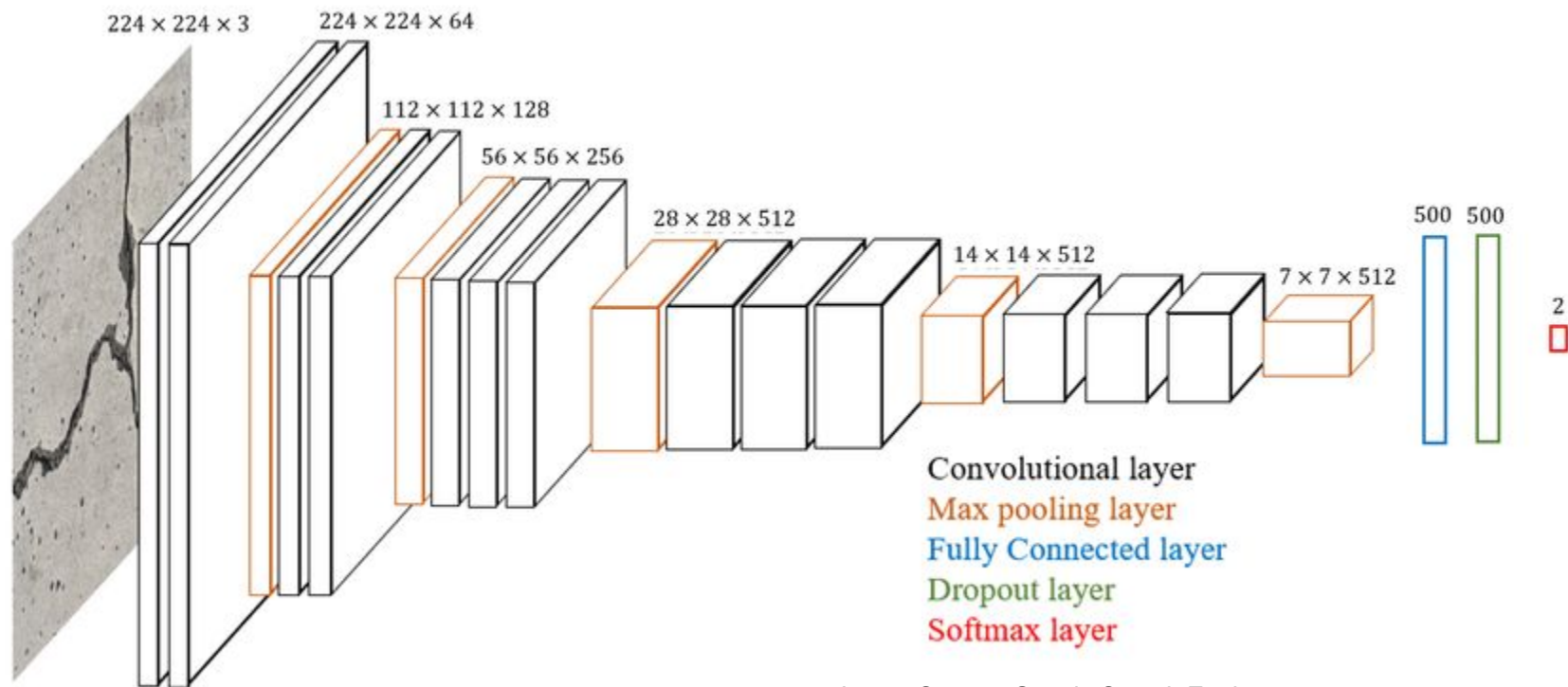


Image Source: Google Search Engine

InceptionNet

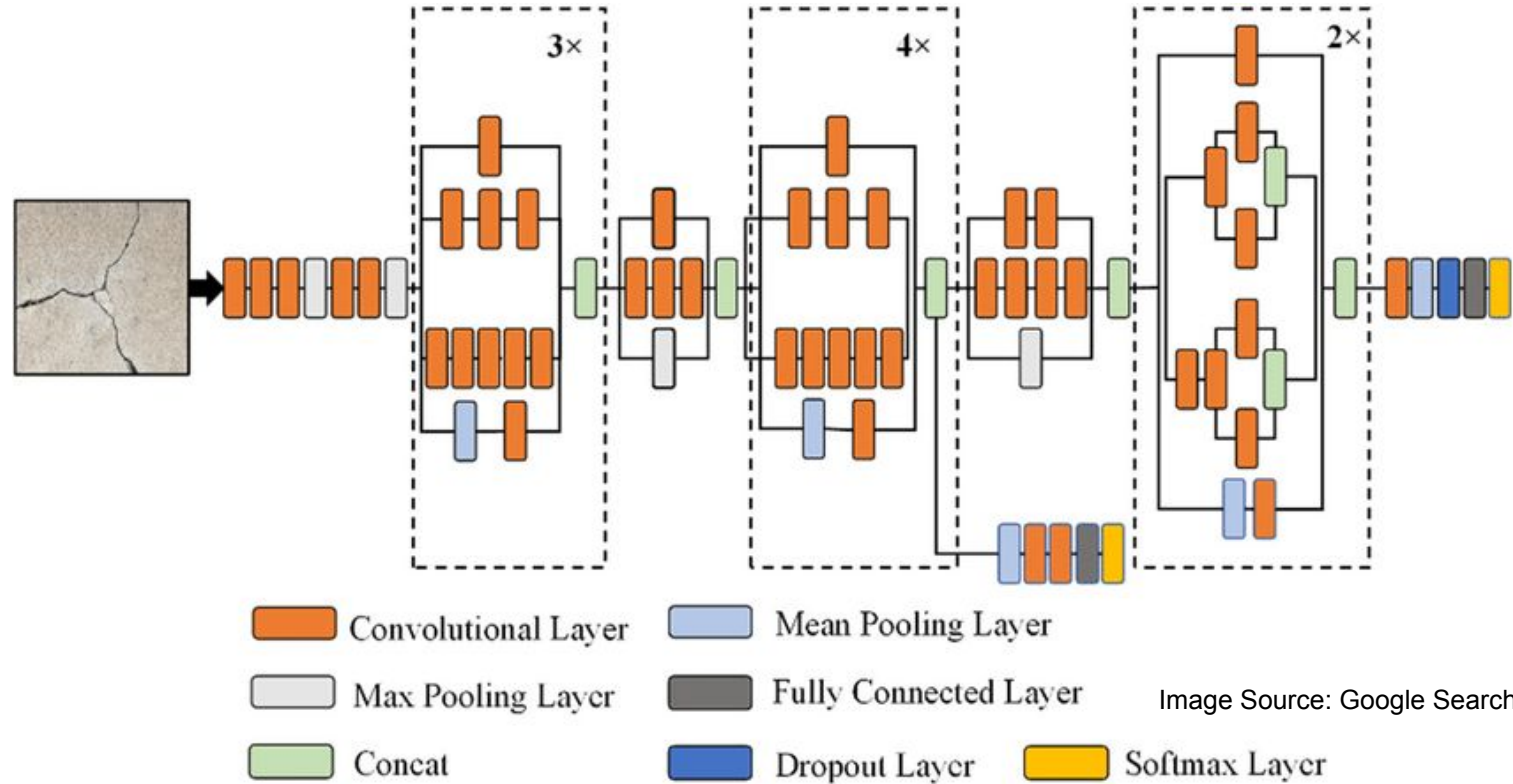


Image Source: Google Search Engine

Inception Modules

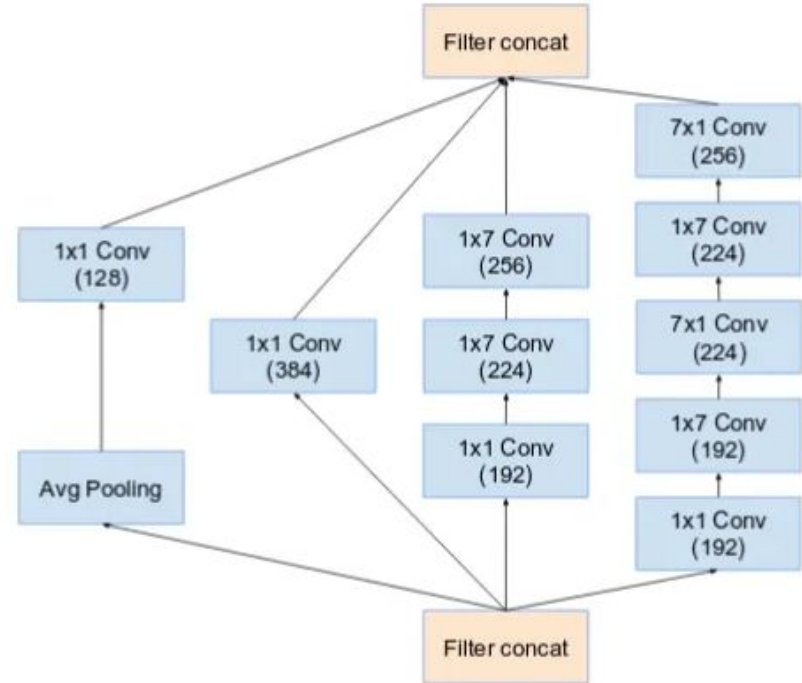
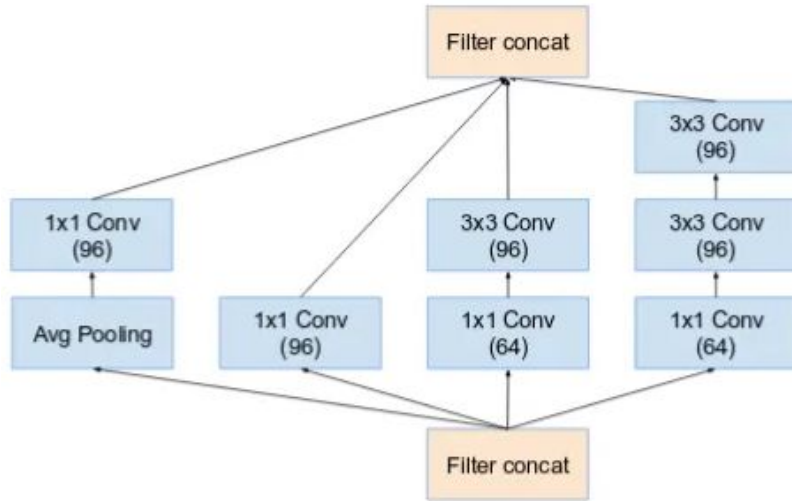
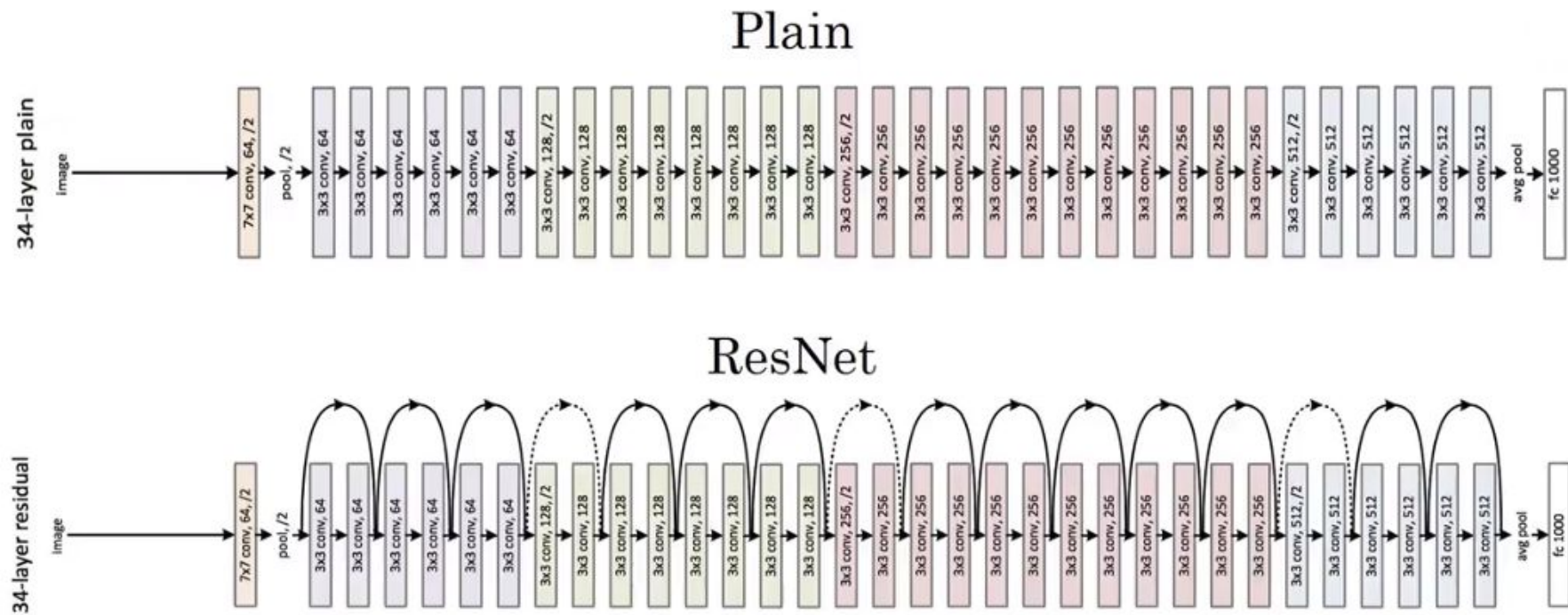


Image Source: Google Search Engine

ResNet



ResNet Block

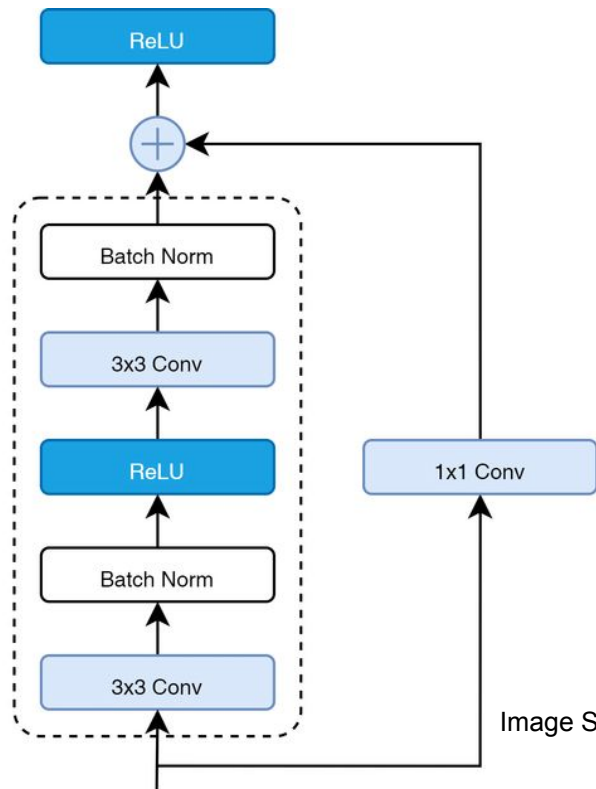
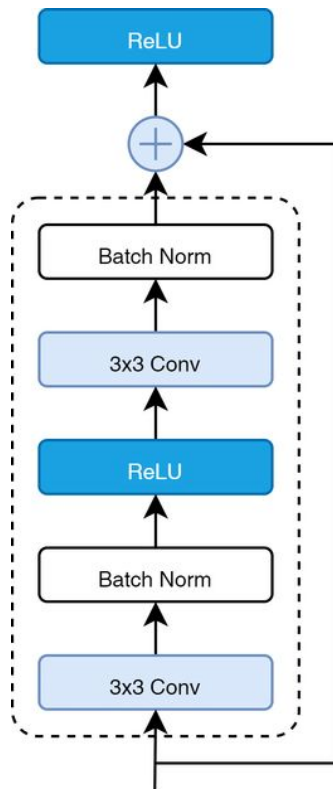
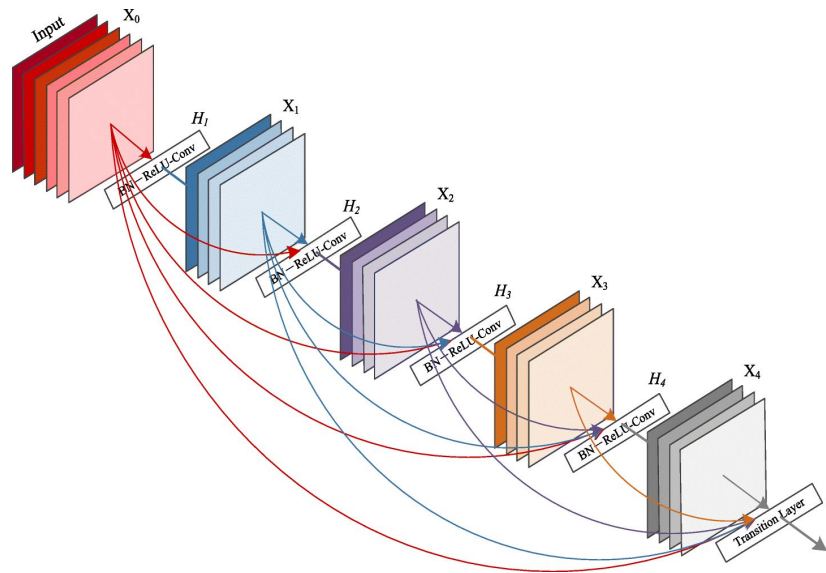
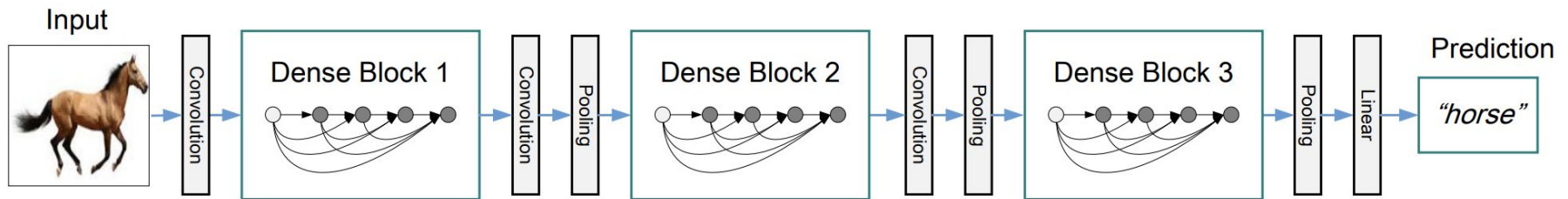


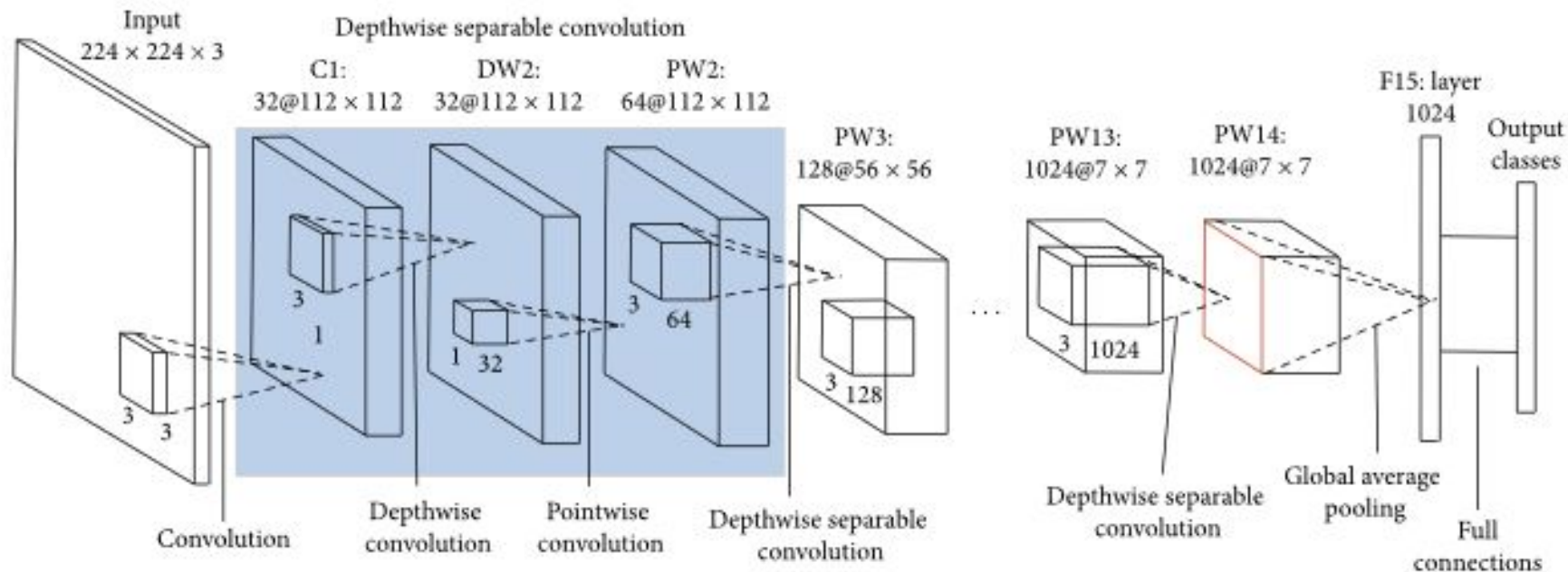
Image Source: Google Search Engine

DenseNet

Skip connections are used for having feature reusability.



MobileNet V2



Xception (E**x**treme version of In**ce**ption)

