PRIOR KNOWLEDGE

TEAM ID:PNT2022TMID50773

PROJECT NAME:DEEP LEARNING FUNDUS IMAGE

ANALYSIS FOR EARLY DETECTION OF DIABETIC RETINOPATHY

DEEP LEARNING CONCEPTS:

1.Convolutional Neural Network (CNN) :

- A convolutional neural network, or CNN, is a deep learning neural network sketched for processing structured arrays of data such as portrayals.
- CNN are very satisfactory at picking up on design in the input image, such as lines, gradients, circles, or even eyes and faces.
- This characteristic that makes convolutional neural network so robust for computer vision.
- CNN can run directly on a underdone image and do not need any preprocessing.

2.VGG16:

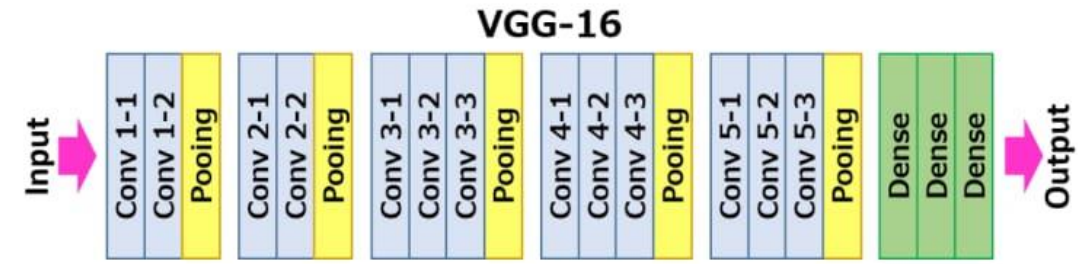
VGG 16 Easiest Explanation VGG16:

It is a **convolutional neural network model** proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper “Very Deep Convolutional Networks for the Large-Scale Image Recognition”.

ARCHITECTURE:

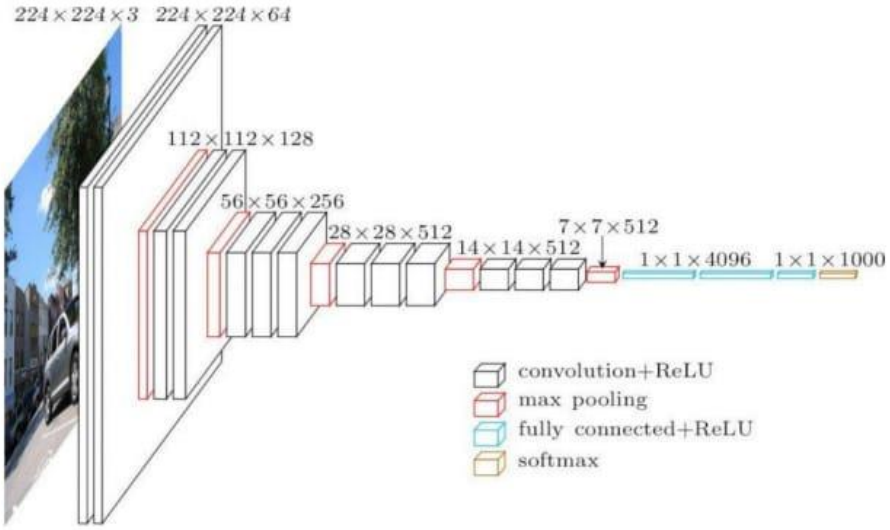
The “deep” refers to the number of layers with VGG-16 or VGG-19 consisting of 16 and 19 convolutional layers.

The VGG architecture is the basis of ground-breaking object recognition models. Developed as a deep neural network, the VGGNet also surpasses baselines on many tasks and datasets beyond ImageNet.



The Architecture

The architecture depicted below is VGG16.



ARCHITECTURE OF VGG16

ResNet50:

ResNet-50 is a convolutional neural network that is 50 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224. For

more pre trained networks in MATLAB, see **Pretrained Deep Neural Networks.**

ARCHITECTURE:

Now we are going to discuss about Resnet 50 and also the architecture for the above talked 18 and 34 layer ResNet is also given residual mapping and not shown for simplicity.

There was a small change that was made for the ResNet 50 and above that before this the shortcut connections skipped two layers but now they skip three layers and also there was 1 * 1 convolution layers added that we are going to see in detail with the ResNet 50 Architecture.

TABLE:

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ARCHITECTURE OF ResNet50

So as we can see in the table 1 the resnet 50 architecture contains the following element:

- A convoultion with a kernel size of 7 * 7 and 64 different kernels all with a stride of size 2 giving us **1 layer**.
- Next we see max pooling with also a stride size of 2.

- In the next convolution there is a 1 * 1,64 kernel following this a 3 * 3,64 kernel and at last a 1 * 1,256 kernel, These three layers are repeated in total 3 time so giving us **9 layers** in this step.
- Next we see kernel of 1 * 1,128 after that a kernel of 3 * 3,128 and at last a kernel of 1 * 1,512 this step was repeated 4 time so giving us **12 layers** in this step.
- After that there is a kernal of 1 * 1,256 and two more kernels with 3 * 3,256 and 1 * 1,1024 and this is repeated 6 time giving us a total of **18 layers**.
- And then again a 1 * 1,512 kernel with two more of 3 * 3,512 and 1 * 1,2048 and this was repeated 3 times giving us a total of **9 layers**.
- After that we do a average pool and end it with a fully connected layer containing 1000 nodes and at the end a softmax function so this gives us **1 layer**.

INCEPTIONV3:

Inception v is a convolutional neural network for assisting in image analysis and object detection, and got its start as a module for Googlenet.

It is the third edition of Google's Inception Convolutional Neural Network, originally introduced during the ImageNet Recognition Challenge. The design of Inceptionv3 was intended to allow deeper networks while also keeping the number of parameters from growing too large: it has "under 25 million parameters", compared against 60 million for AlexNet.

Just as ImageNet can be thought of as a database of classified visual objects, Inception helps classification of objects in the world of computer vision.

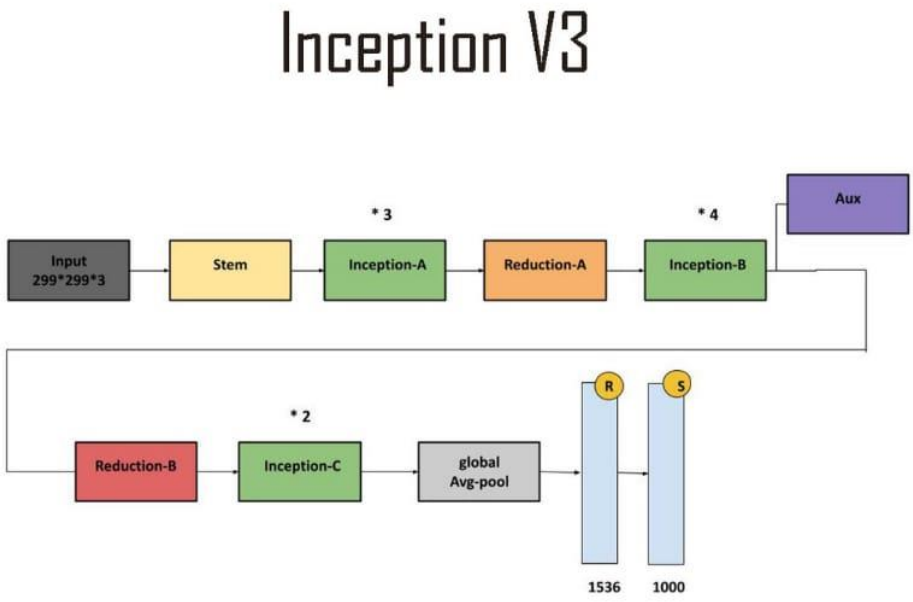
The Inceptionv3 architecture has been reused in many different applications, often used "pre-trained" from ImageNet. One such use is in life sciences, where it aids in the research of leukemia.

The original name (Inception) was codenamed this way after a popular "'we need to go deeper' internet meme" went viral, quoting a phrase from Inception film of Christopher Nolan.

ARCHITECTURE:

The inception v3 model was released in the year 2015, it has a total of 42 layers and a lower error rate than its predecessors. Let's look at what are the different optimizations that make the inception V3 model better.The major modifications done on the Inception V3 model are

1. Factorization into Smaller Convolutions
2. Spatial Factorization into Asymmetric Convolutions
3. Utility of Auxiliary Classifiers
4. Efficient Grid Size Reduction

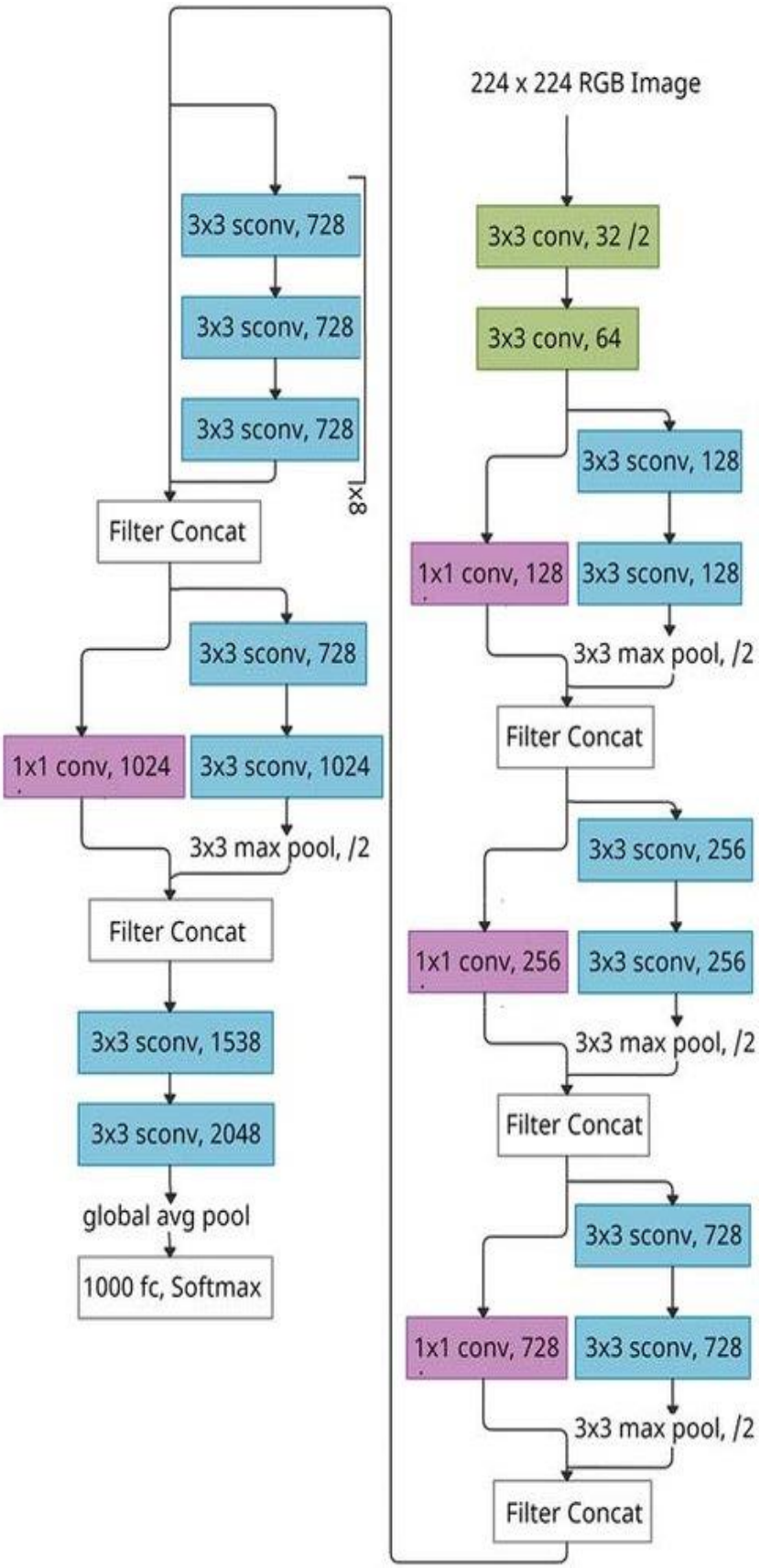


Xception:

Xception is a convolutional neural network that is 71 layers deep. It can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of

images. The network has an image input size of 299-by-299. For more pretrained networks in MATLAB, see Pretrained Deep Neural Networks.

ARCHITECTURE:


224 x 224 RGB Image
3x3 conv, 32 /2
3x3 conv, 64
3x3 sconv, 128
1x1 conv, 128
3x3 sconv, 128
3x3 max pool, /2
Filter Concat
3x3 sconv, 256
1x1 conv, 256
3x3 sconv, 256
3x3 max pool, /2
Filter Concat
3x3 sconv, 728
1x1 conv, 728
3x3 sconv, 728
3x3 max pool, /2
Filter Concat
3x3 sconv, 728
3x3 sconv, 728
3x3 sconv, 728
x8
Filter Concat
3x3 sconv, 728
1x1 conv, 1024
3x3 sconv, 1024
3x3 max pool, /2
Filter Concat
3x3 sconv, 1538
3x3 sconv, 2048
global avg pool
1000 fc, Softmax

FLASK:

Flask is a **micro web framework** written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, and various open in web application.