

CNN on CIFR Assignment: ¶

1. Please visit this link to access the state-of-art DenseNet code for reference - DenseNet - cifar10 notebook link
2. You need to create a copy of this and "retrain" this model to achieve 90+ test accuracy.
3. You cannot use Dense Layers (also called fully connected layers), or DropOut.
4. You MUST use Image Augmentation Techniques.
5. You cannot use an already trained model as a beginning points, you have to initialize as your own
6. You cannot run the program for more than 300 Epochs, and it should be clear from your log, that you have only used 300 Epochs
7. You cannot use test images for training the model.
8. You cannot change the general architecture of DenseNet (which means you must use Dense Block, Transition and Output blocks as mentioned in the code)
9. You are free to change Convolution types (e.g. from 3x3 normal convolution to Depthwise Separable, etc)
10. You cannot have more than 1 Million parameters in total
11. You are free to move the code from Keras to Tensorflow, Pytorch, MXNET etc.
12. You can use any optimization algorithm you need.
13. You can checkpoint your model and retrain the model from that checkpoint so that no need of training the model from first if you lost at any epoch while training. You can directly load that model and Train from that epoch.

```
In [0]: # ! pip install tensorflow==1.13.1
```

```
In [0]: import warnings
warnings.filterwarnings("ignore")
import tensorflow as tf
from tensorflow.keras.regularizers import l2
from tensorflow.keras.layers import Dense, AveragePooling2D, BatchNormalization,
Conv2D, Input, Flatten, Activation, concatenate, Dropout, GlobalAveragePooling2D
from tensorflow.keras.models import Model
```

```
In [0]: # Hyperparameters
batch_size = 64
num_classes = 10
num_filter = 12
compression = 0.5
```

```
In [4]: # Load CIFAR10 Data
(X_train, y_train), (X_test, y_test) = tf.keras.datasets.cifar10.load_data()
img_height, img_width, channel = X_train.shape[1], X_train.shape[2], X_train.shape[3]
```

Downloading data from <https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz>
170500096/170498071 [=====] - 3s 0us/step

```
In [0]: # from sklearn.model_selection import train_test_split
# from sklearn.utils import resample
# X_train,y_train = resample(X_train,y_train,n_samples = 30000)
# X_train, X_cv, y_train, y_cv = train_test_split(X_train, y_train, test_size=
5/30, random_state=42)
# X_train, X_test, y_train, y_test = train_test_split(X_train, y_train, test_s
ize=5/25, random_state=42)
```

```
In [0]: from sklearn.model_selection import train_test_split
from sklearn.utils import resample
X_train, X_cv, y_train, y_cv = train_test_split(X_train, y_train, test_size=10
/50, random_state=42)
```

```
In [0]: # convert to one hot encoing
y_train = tf.keras.utils.to_categorical(y_train, num_classes)
y_test = tf.keras.utils.to_categorical(y_test, num_classes)
y_cv = tf.keras.utils.to_categorical(y_cv, num_classes)
```

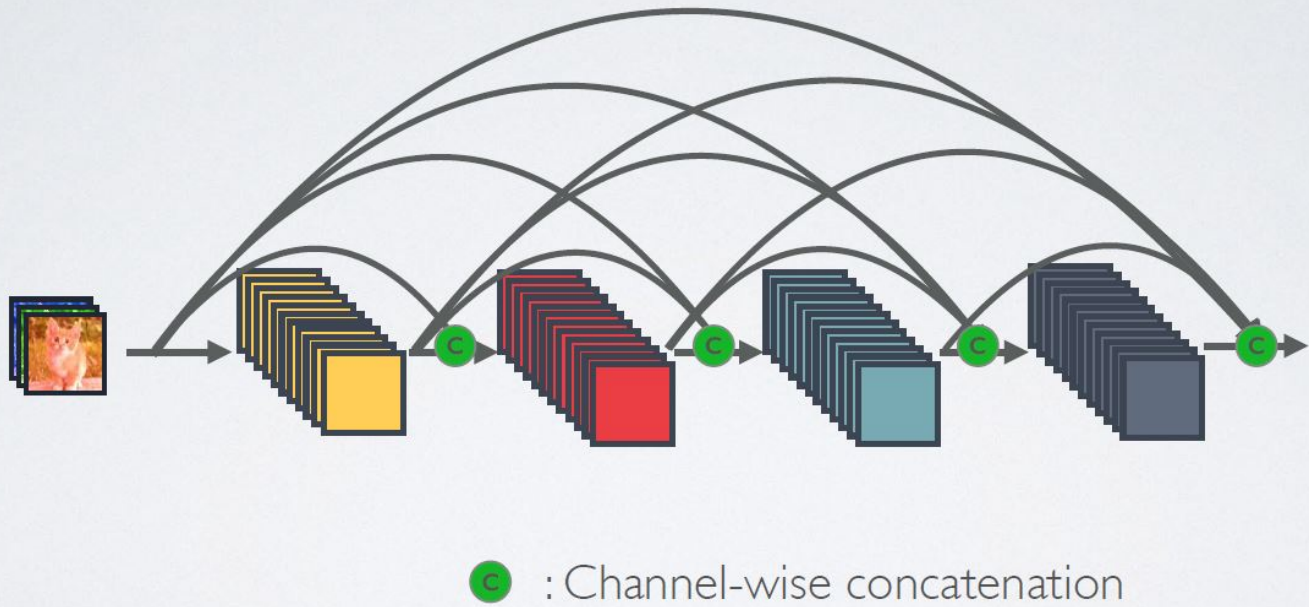
```
In [7]: print(X_train.shape,y_train.shape)
print(X_test.shape,y_test.shape)
print(X_cv.shape,y_cv.shape)
```

```
(40000, 32, 32, 3) (40000, 10)
(10000, 32, 32, 3) (10000, 10)
(10000, 32, 32, 3) (10000, 10)
```

Model Architechture

Implemented **DenseNet-BC(k= 12, L = 100)** with **0.76M parameters** with cifar-10 using augmentation.

DENSE CONNECTIVITY



```
In [0]: class Convolution():
    """
    Before entering the first dense vlock, a convolution with 16 (or twice the
    growth rate for BC type) output channels is performed on the input images
    """
    def __init__(self, growthRate = 12, weight_decay=1E-4):
        self.growthRate = growthRate
        self.weight_decay = weight_decay

    def __call__(self, x):

        #x = Activation('relu')(x)
        op = Conv2D(int(2*self.growthRate),3,padding="same",use_bias=False,bias_initializer='zeros',kernel_regularizer=l2(self.weight_decay))(x)

        return op
```

```
In [9]: # test code
in_ = Input(shape=(32,32,3))
op = Convolution()(in_)
print(op.shape)
```

WARNING: Logging before flag parsing goes to stderr.

W0813 16:36:52.858003 140186611292032 deprecation.py:506] From /usr/local/lib/python3.6/dist-packages/tensorflow/python/ops/init_ops.py:1251: calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be removed in a future version.

Instructions for updating:

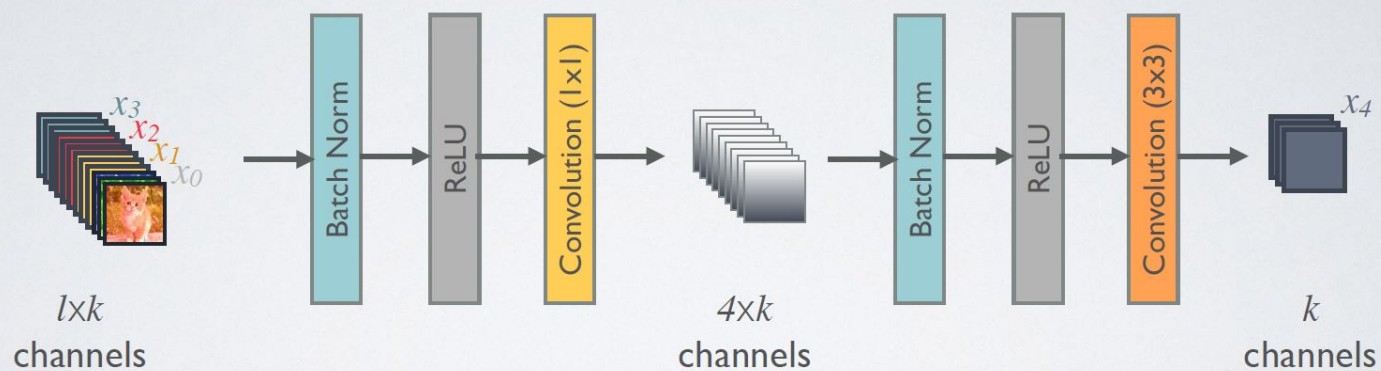
Call initializer instance with the dtype argument instead of passing it to the constructor

(?, 32, 32, 24)

Class BottleNeck

COMPOSITE LAYER IN DENSENET

WITH BOTTLENECK LAYER



Higher parameter and computational efficiency

```
In [0]: class Bottleneck():
        """
        This class implements H(L) as mentioned in paper.
        BN-ReLU-Conv(1x1)-BN-ReLU-Conv(3x3) version of H.
        """

        def __init__(self, growthrate = 12, weight_decay=1E-4):

            self.growthrate = growthrate
            self.weight_decay = weight_decay

        def __call__(self, x):
            """
            This is where Logic Lives.
            """
            x = BatchNormalization(beta_regularizer=l2(self.weight_decay), gamma_regularizer=l2(self.weight_decay))(x)
            x = Activation("relu")(x)
            x = Conv2D(int(4*self.growthrate), 1, padding="same", use_bias=False, kernel_initializer="he_uniform", kernel_regularizer=l2(self.weight_decay))(x)

            x = BatchNormalization(beta_regularizer=l2(self.weight_decay), gamma_regularizer=l2(self.weight_decay))(x)
            x = Activation("relu")(x)
            x = Conv2D(int(self.growthrate), 3, padding="same", use_bias=False, kernel_initializer="he_uniform", kernel_regularizer=l2(self.weight_decay))(x)

            return x
```

Class Transition

```
In [0]: class Transition():
        """
        This class implements transition layer as per paper.
        BN-Con-pooling
        """
        def __init__(self,compression = 0.5,growthrate = 12,weight_decay=1E-4):

            self.compression = compression
            self.growthrate = growthrate
            self.weight_decay = weight_decay

        def __call__(self,x):

            nChannels = x.shape.as_list()[-1]

            x = BatchNormalization(beta_regularizer=l2(self.weight_decay),gamma_regularizer=l2(self.weight_decay))(x)
            x = Activation("relu")(x)
            x = Conv2D(int(self.compression * nChannels),1,padding="same",use_bias=False,kernel_regularizer=l2(self.weight_decay))(x)
            x = AveragePooling2D(pool_size=(2,2))(x)

            return x
```

Class DenseNet

```
In [0]: class Densenet():
        """
        This layer implements densenet.
        """
        def __init__(self, depth, growthrate = 12, bottleneck = True):

            nDenseLayers = (depth-4) // 3
            if bottleneck:
                nDenseLayers //= 2

            self.nDenseLayers = nDenseLayers
            self.growthrate = growthrate

        def __call__(self, x):

            # Dense Block Logic

            # convolution block
            x = Convolution()(x)

            # 1st dense block
            layers = []
            for i in range(int(self.nDenseLayers)):
                bt = Bottleneck()(x)
                x = concatenate([x, bt])

            # transition layer
            x = Transition()(x)

            # 2nd dense block
            layers = []
            for i in range(int(self.nDenseLayers)):
                bt = Bottleneck()(x)
                x = concatenate([x, bt])

            # transition layer
            x = Transition()(x)

            # 3rd dense block
            layers = []
            for i in range(int(self.nDenseLayers)):
                bt = Bottleneck()(x)
                x = concatenate([x, bt])

            return x
```

Class Output


```
In [0]: class Output():
        """
        This layer implements output layer.
        """

        def __init__(self,num_classes = 10,weight_decay=1E-4):

            self.num_classes = num_classes
            self.weight_decay = weight_decay

        def __call__(self,x):

            x = BatchNormalization(beta_regularizer=l2(self.weight_decay),gamma_regularizer=l2(self.weight_decay))(x)
            x = GlobalAveragePooling2D()(x)
            #avg = AveragePooling2D(pool_size=(8,8))(x)
            #flt = Flatten()(avg)
            op = Dense(self.num_classes,activation="softmax",kernel_regularizer=l2(self.weight_decay), bias_regularizer=l2(self.weight_decay))(x)

            return op
```

Model Building

```
In [0]: ## The DenseNet used in our experiments has three dense blocks that each has
        an equal number of layers.

        #

        ## First convolution
        # cnv = Convolution()(in_)

        ## For dense block 1
        # op = Densenet(depth=100)(cnv)
        # op = Transition()(op)

        ## For dense block 2
        # op = Densenet(depth=100)(op)
        # op = Transition()(op)

        in_ = Input(shape=(32,32,3))

        # densenet
        op = Densenet(100)(in_)

        # output
        op = Output()(op)
```

```
In [0]: model = Model(inputs = in_,outputs = op)
        sgd = tf.keras.optimizers.SGD(lr = 0.1,momentum = 0.9,nesterov = True)
        model.compile(sgd,loss="categorical_crossentropy",metrics=["acc"])
```

In [16]: `model.summary()`

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
=====			
input_2 (InputLayer)	[(None, 32, 32, 3)]	0	
conv2d_1 (Conv2D)	(None, 32, 32, 24)	648	input_2[0]
batch_normalization (BatchNormaliza	(None, 32, 32, 24)	96	conv2d_1[0]
activation (Activation)	(None, 32, 32, 24)	0	batch_normalization[0][0]
conv2d_2 (Conv2D)	(None, 32, 32, 48)	1152	activation[0][0]
batch_normalization_1 (BatchNormal	(None, 32, 32, 48)	192	conv2d_2[0]
activation_1 (Activation)	(None, 32, 32, 48)	0	batch_normalization_1[0][0]
conv2d_3 (Conv2D)	(None, 32, 32, 12)	5184	activation_1[0][0]
concatenate (Concatenate)	(None, 32, 32, 36)	0	conv2d_1[0] conv2d_3[0]
batch_normalization_2 (BatchNormal	(None, 32, 32, 36)	144	concatenate[0][0]
activation_2 (Activation)	(None, 32, 32, 36)	0	batch_normalization_2[0][0]
conv2d_4 (Conv2D)	(None, 32, 32, 48)	1728	activation_2[0][0]
batch_normalization_3 (BatchNormal	(None, 32, 32, 48)	192	conv2d_4[0]

activation_3 (Activation) ization_3[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_5 (Conv2D) [0][0]	(None, 32, 32, 12)	5184	activation_3
concatenate_1 (Concatenate) [0][0] [0]	(None, 32, 32, 48)	0	concatenate conv2d_5[0]
batch_normalization_4 (BatchNor 1[0][0]	(None, 32, 32, 48)	192	concatenate_
activation_4 (Activation) ization_4[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_6 (Conv2D) [0][0]	(None, 32, 32, 48)	2304	activation_4
batch_normalization_5 (BatchNor [0]	(None, 32, 32, 48)	192	conv2d_6[0]
activation_5 (Activation) ization_5[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_7 (Conv2D) [0][0]	(None, 32, 32, 12)	5184	activation_5
concatenate_2 (Concatenate) 1[0][0] [0]	(None, 32, 32, 60)	0	concatenate_ conv2d_7[0]
batch_normalization_6 (BatchNor 2[0][0]	(None, 32, 32, 60)	240	concatenate_
activation_6 (Activation) ization_6[0][0]	(None, 32, 32, 60)	0	batch_normal
conv2d_8 (Conv2D) [0][0]	(None, 32, 32, 48)	2880	activation_6

batch_normalization_7 (BatchNor	(None, 32, 32, 48)	192	conv2d_8[0]
[0]			
activation_7 (Activation)	(None, 32, 32, 48)	0	batch_normal
ization_7[0][0]			
conv2d_9 (Conv2D)	(None, 32, 32, 12)	5184	activation_7
[0][0]			
concatenate_3 (Concatenate)	(None, 32, 32, 72)	0	concatenate_
2[0][0]			
[0]			conv2d_9[0]
batch_normalization_8 (BatchNor	(None, 32, 32, 72)	288	concatenate_
3[0][0]			
activation_8 (Activation)	(None, 32, 32, 72)	0	batch_normal
ization_8[0][0]			
conv2d_10 (Conv2D)	(None, 32, 32, 48)	3456	activation_8
[0][0]			
batch_normalization_9 (BatchNor	(None, 32, 32, 48)	192	conv2d_10[0]
[0]			
activation_9 (Activation)	(None, 32, 32, 48)	0	batch_normal
ization_9[0][0]			
conv2d_11 (Conv2D)	(None, 32, 32, 12)	5184	activation_9
[0][0]			
concatenate_4 (Concatenate)	(None, 32, 32, 84)	0	concatenate_
3[0][0]			
[0]			conv2d_11[0]
batch_normalization_10 (BatchNo	(None, 32, 32, 84)	336	concatenate_
4[0][0]			
activation_10 (Activation)	(None, 32, 32, 84)	0	batch_normal
ization_10[0][0]			

conv2d_12 (Conv2D) 0[0][0]	(None, 32, 32, 48)	4032	activation_1
batch_normalization_11 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_12[0]
activation_11 (Activation) ization_11[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_13 (Conv2D) 1[0][0]	(None, 32, 32, 12)	5184	activation_1
concatenate_5 (Concatenate) 4[0][0]	(None, 32, 32, 96)	0	concatenate_ conv2d_13[0]
batch_normalization_12 (BatchNo 5[0][0])	(None, 32, 32, 96)	384	concatenate_
activation_12 (Activation) ization_12[0][0]	(None, 32, 32, 96)	0	batch_normal
conv2d_14 (Conv2D) 2[0][0]	(None, 32, 32, 48)	4608	activation_1
batch_normalization_13 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_14[0]
activation_13 (Activation) ization_13[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_15 (Conv2D) 3[0][0]	(None, 32, 32, 12)	5184	activation_1
concatenate_6 (Concatenate) 5[0][0]	(None, 32, 32, 108)	0	concatenate_ conv2d_15[0]
batch_normalization_14 (BatchNo 6[0][0])	(None, 32, 32, 108)	432	concatenate_
activation_14 (Activation)	(None, 32, 32, 108)	0	batch_normal

ization_14[0][0]

conv2d_16 (Conv2D) 4[0][0]	(None, 32, 32, 48)	5184	activation_14[0][0]
-------------------------------	--------------------	------	---------------------

batch_normalization_15 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_16[0] [0]
---	--------------------	-----	---------------------

activation_15 (Activation) ization_15[0][0]	(None, 32, 32, 48)	0	batch_normal ization_15[0][0]
--	--------------------	---	----------------------------------

conv2d_17 (Conv2D) 5[0][0]	(None, 32, 32, 12)	5184	activation_15[0][0]
-------------------------------	--------------------	------	---------------------

concatenate_7 (Concatenate) 6[0][0]	(None, 32, 32, 120)	0	concatenate_6[0][0]
--	---------------------	---	---------------------

batch_normalization_16 (BatchNo 7[0][0])	(None, 32, 32, 120)	480	concatenate_7[0][0]
---	---------------------	-----	---------------------

activation_16 (Activation) ization_16[0][0]	(None, 32, 32, 120)	0	batch_normal ization_16[0][0]
--	---------------------	---	----------------------------------

conv2d_18 (Conv2D) 6[0][0]	(None, 32, 32, 48)	5760	activation_16[0][0]
-------------------------------	--------------------	------	---------------------

batch_normalization_17 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_18[0] [0]
---	--------------------	-----	---------------------

activation_17 (Activation) ization_17[0][0]	(None, 32, 32, 48)	0	batch_normal ization_17[0][0]
--	--------------------	---	----------------------------------

conv2d_19 (Conv2D) 7[0][0]	(None, 32, 32, 12)	5184	activation_17[0][0]
-------------------------------	--------------------	------	---------------------

concatenate_8 (Concatenate) 7[0][0]	(None, 32, 32, 132)	0	concatenate_7[0][0]
--	---------------------	---	---------------------

batch_normalization_18 (BatchNo 8[0][0])	(None, 32, 32, 132)	528	concatenate_8[0][0]
---	---------------------	-----	---------------------

activation_18 (Activation) ization_18[0][0]	(None, 32, 32, 132)	0	batch_normal
conv2d_20 (Conv2D) 8[0][0]	(None, 32, 32, 48)	6336	activation_1
batch_normalization_19 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_20[0]
activation_19 (Activation) ization_19[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_21 (Conv2D) 9[0][0]	(None, 32, 32, 12)	5184	activation_1
concatenate_9 (Concatenate) 8[0][0]	(None, 32, 32, 144)	0	concatenate_ conv2d_21[0]
batch_normalization_20 (BatchNo 9[0][0])	(None, 32, 32, 144)	576	concatenate_
activation_20 (Activation) ization_20[0][0]	(None, 32, 32, 144)	0	batch_normal
conv2d_22 (Conv2D) 0[0][0]	(None, 32, 32, 48)	6912	activation_2
batch_normalization_21 (BatchNo [0])	(None, 32, 32, 48)	192	conv2d_22[0]
activation_21 (Activation) ization_21[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_23 (Conv2D) 1[0][0]	(None, 32, 32, 12)	5184	activation_2
concatenate_10 (Concatenate) 9[0][0]	(None, 32, 32, 156)	0	concatenate_ conv2d_23[0]
			[0]

batch_normalization_22 (BatchNo	(None, 32, 32, 156)	624	concatenate_10[0][0]
activation_22 (Activation)	(None, 32, 32, 156)	0	batch_normalization_22[0][0]
conv2d_24 (Conv2D)	(None, 32, 32, 48)	7488	activation_22[0][0]
batch_normalization_23 (BatchNo	(None, 32, 32, 48)	192	conv2d_24[0][0]
activation_23 (Activation)	(None, 32, 32, 48)	0	batch_normalization_23[0][0]
conv2d_25 (Conv2D)	(None, 32, 32, 12)	5184	activation_23[0][0]
concatenate_11 (Concatenate)	(None, 32, 32, 168)	0	concatenate_10[0][0]
			conv2d_25[0][0]
batch_normalization_24 (BatchNo	(None, 32, 32, 168)	672	concatenate_11[0][0]
activation_24 (Activation)	(None, 32, 32, 168)	0	batch_normalization_24[0][0]
conv2d_26 (Conv2D)	(None, 32, 32, 48)	8064	activation_24[0][0]
batch_normalization_25 (BatchNo	(None, 32, 32, 48)	192	conv2d_26[0][0]
activation_25 (Activation)	(None, 32, 32, 48)	0	batch_normalization_25[0][0]
conv2d_27 (Conv2D)	(None, 32, 32, 12)	5184	activation_25[0][0]
concatenate_12 (Concatenate)	(None, 32, 32, 180)	0	concatenate_11[0][0]

conv2d_27[0]

[0]

batch_normalization_26 (BatchNo	(None, 32, 32, 180)	720	concatenate_12[0][0]
activation_26 (Activation)	(None, 32, 32, 180)	0	batch_normalization_26[0][0]
conv2d_28 (Conv2D)	(None, 32, 32, 48)	8640	activation_26[0][0]
batch_normalization_27 (BatchNo	(None, 32, 32, 48)	192	conv2d_28[0][0]
activation_27 (Activation)	(None, 32, 32, 48)	0	batch_normalization_27[0][0]
conv2d_29 (Conv2D)	(None, 32, 32, 12)	5184	activation_27[0][0]
concatenate_13 (Concatenate)	(None, 32, 32, 192)	0	concatenate_12[0][0]
			conv2d_29[0][0]
batch_normalization_28 (BatchNo	(None, 32, 32, 192)	768	concatenate_13[0][0]
activation_28 (Activation)	(None, 32, 32, 192)	0	batch_normalization_28[0][0]
conv2d_30 (Conv2D)	(None, 32, 32, 48)	9216	activation_28[0][0]
batch_normalization_29 (BatchNo	(None, 32, 32, 48)	192	conv2d_30[0][0]
activation_29 (Activation)	(None, 32, 32, 48)	0	batch_normalization_29[0][0]
conv2d_31 (Conv2D)	(None, 32, 32, 12)	5184	activation_29[0][0]

concatenate_14 (Concatenate) 13[0][0]	(None, 32, 32, 204)	0	concatenate_ conv2d_31[0]
batch_normalization_30 (BatchNo 14[0][0]	(None, 32, 32, 204)	816	concatenate_
activation_30 (Activation) ization_30[0][0]	(None, 32, 32, 204)	0	batch_normal
conv2d_32 (Conv2D) 0[0][0]	(None, 32, 32, 48)	9792	activation_3
batch_normalization_31 (BatchNo [0]	(None, 32, 32, 48)	192	conv2d_32[0]
activation_31 (Activation) ization_31[0][0]	(None, 32, 32, 48)	0	batch_normal
conv2d_33 (Conv2D) 1[0][0]	(None, 32, 32, 12)	5184	activation_3
concatenate_15 (Concatenate) 14[0][0]	(None, 32, 32, 216)	0	concatenate_ conv2d_33[0]
batch_normalization_32 (BatchNo 15[0][0]	(None, 32, 32, 216)	864	concatenate_
activation_32 (Activation) ization_32[0][0]	(None, 32, 32, 216)	0	batch_normal
conv2d_34 (Conv2D) 2[0][0]	(None, 32, 32, 108)	23328	activation_3
average_pooling2d (AveragePooli [0]	(None, 16, 16, 108)	0	conv2d_34[0]
batch_normalization_33 (BatchNo ing2d[0][0]	(None, 16, 16, 108)	432	average_pool

activation_33 (Activation) ization_33[0][0]	(None, 16, 16, 108)	0	batch_normal
conv2d_35 (Conv2D) 3[0][0]	(None, 16, 16, 48)	5184	activation_3
batch_normalization_34 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_35[0]
activation_34 (Activation) ization_34[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_36 (Conv2D) 4[0][0]	(None, 16, 16, 12)	5184	activation_3
concatenate_16 (Concatenate) ing2d[0][0] [0]	(None, 16, 16, 120)	0	average_pool conv2d_36[0]
batch_normalization_35 (BatchNo 16[0][0]	(None, 16, 16, 120)	480	concatenate_
activation_35 (Activation) ization_35[0][0]	(None, 16, 16, 120)	0	batch_normal
conv2d_37 (Conv2D) 5[0][0]	(None, 16, 16, 48)	5760	activation_3
batch_normalization_36 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_37[0]
activation_36 (Activation) ization_36[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_38 (Conv2D) 6[0][0]	(None, 16, 16, 12)	5184	activation_3
concatenate_17 (Concatenate) 16[0][0] [0]	(None, 16, 16, 132)	0	concatenate_ conv2d_38[0]
batch_normalization_37 (BatchNo	(None, 16, 16, 132)	528	concatenate_

17[0][0]

activation_37 (Activation) ization_37[0][0]	(None, 16, 16, 132)	0	batch_normal
conv2d_39 (Conv2D) 7[0][0]	(None, 16, 16, 48)	6336	activation_3
batch_normalization_38 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_39[0]
activation_38 (Activation) ization_38[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_40 (Conv2D) 8[0][0]	(None, 16, 16, 12)	5184	activation_3
concatenate_18 (Concatenate) 17[0][0]	(None, 16, 16, 144)	0	concatenate_ conv2d_40[0]
batch_normalization_39 (BatchNo 18[0][0]	(None, 16, 16, 144)	576	concatenate_
activation_39 (Activation) ization_39[0][0]	(None, 16, 16, 144)	0	batch_normal
conv2d_41 (Conv2D) 9[0][0]	(None, 16, 16, 48)	6912	activation_3
batch_normalization_40 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_41[0]
activation_40 (Activation) ization_40[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_42 (Conv2D) 0[0][0]	(None, 16, 16, 12)	5184	activation_4
concatenate_19 (Concatenate) 18[0][0]	(None, 16, 16, 156)	0	concatenate_ conv2d_42[0]

[0]

batch_normalization_41 (BatchNo	(None, 16, 16, 156)	624	concatenate_19[0][0]
activation_41 (Activation)	(None, 16, 16, 156)	0	batch_normalization_41[0][0]
conv2d_43 (Conv2D)	(None, 16, 16, 48)	7488	activation_41[0][0]
batch_normalization_42 (BatchNo	(None, 16, 16, 48)	192	conv2d_43[0][0]
activation_42 (Activation)	(None, 16, 16, 48)	0	batch_normalization_42[0][0]
conv2d_44 (Conv2D)	(None, 16, 16, 12)	5184	activation_42[0][0]
concatenate_20 (Concatenate)	(None, 16, 16, 168)	0	concatenate_19[0][0]
			conv2d_44[0][0]
batch_normalization_43 (BatchNo	(None, 16, 16, 168)	672	concatenate_20[0][0]
activation_43 (Activation)	(None, 16, 16, 168)	0	batch_normalization_43[0][0]
conv2d_45 (Conv2D)	(None, 16, 16, 48)	8064	activation_43[0][0]
batch_normalization_44 (BatchNo	(None, 16, 16, 48)	192	conv2d_45[0][0]
activation_44 (Activation)	(None, 16, 16, 48)	0	batch_normalization_44[0][0]
conv2d_46 (Conv2D)	(None, 16, 16, 12)	5184	activation_44[0][0]
concatenate_21 (Concatenate)	(None, 16, 16, 180)	0	concatenate_20[0][0]

20[0][0]			conv2d_46[0]
[0]			
batch_normalization_45 (BatchNo	(None, 16, 16, 180)	720	concatenate_21[0][0]
activation_45 (Activation)	(None, 16, 16, 180)	0	batch_normalization_45[0][0]
conv2d_47 (Conv2D)	(None, 16, 16, 48)	8640	activation_45[0][0]
batch_normalization_46 (BatchNo	(None, 16, 16, 48)	192	conv2d_47[0][0]
activation_46 (Activation)	(None, 16, 16, 48)	0	batch_normalization_46[0][0]
conv2d_48 (Conv2D)	(None, 16, 16, 12)	5184	activation_46[0][0]
concatenate_22 (Concatenate)	(None, 16, 16, 192)	0	concatenate_21[0][0]
[0]			conv2d_48[0]
batch_normalization_47 (BatchNo	(None, 16, 16, 192)	768	concatenate_22[0][0]
activation_47 (Activation)	(None, 16, 16, 192)	0	batch_normalization_47[0][0]
conv2d_49 (Conv2D)	(None, 16, 16, 48)	9216	activation_47[0][0]
batch_normalization_48 (BatchNo	(None, 16, 16, 48)	192	conv2d_49[0][0]
activation_48 (Activation)	(None, 16, 16, 48)	0	batch_normalization_48[0][0]
conv2d_50 (Conv2D)	(None, 16, 16, 12)	5184	activation_48[0][0]

concatenate_23 (Concatenate) 22[0][0]	(None, 16, 16, 204)	0	concatenate_ conv2d_50[0]
			[0]
batch_normalization_49 (BatchNo 23[0][0]	(None, 16, 16, 204)	816	concatenate_
activation_49 (Activation) ization_49[0][0]	(None, 16, 16, 204)	0	batch_normal
conv2d_51 (Conv2D) 9[0][0]	(None, 16, 16, 48)	9792	activation_4
batch_normalization_50 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_51[0]
activation_50 (Activation) ization_50[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_52 (Conv2D) 0[0][0]	(None, 16, 16, 12)	5184	activation_5
concatenate_24 (Concatenate) 23[0][0]	(None, 16, 16, 216)	0	concatenate_
			conv2d_52[0]
			[0]
batch_normalization_51 (BatchNo 24[0][0]	(None, 16, 16, 216)	864	concatenate_
activation_51 (Activation) ization_51[0][0]	(None, 16, 16, 216)	0	batch_normal
conv2d_53 (Conv2D) 1[0][0]	(None, 16, 16, 48)	10368	activation_5
batch_normalization_52 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_53[0]
activation_52 (Activation) ization_52[0][0]	(None, 16, 16, 48)	0	batch_normal

conv2d_54 (Conv2D) 2[0][0]	(None, 16, 16, 12)	5184	activation_5
concatenate_25 (Concatenate) 24[0][0]	(None, 16, 16, 228)	0	concatenate_ conv2d_54[0]
batch_normalization_53 (BatchNo 25[0][0])	(None, 16, 16, 228)	912	concatenate_
activation_53 (Activation) ization_53[0][0]	(None, 16, 16, 228)	0	batch_normal
conv2d_55 (Conv2D) 3[0][0]	(None, 16, 16, 48)	10944	activation_5
batch_normalization_54 (BatchNo [0])	(None, 16, 16, 48)	192	conv2d_55[0]
activation_54 (Activation) ization_54[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_56 (Conv2D) 4[0][0]	(None, 16, 16, 12)	5184	activation_5
concatenate_26 (Concatenate) 25[0][0]	(None, 16, 16, 240)	0	concatenate_ conv2d_56[0]
batch_normalization_55 (BatchNo 26[0][0])	(None, 16, 16, 240)	960	concatenate_
activation_55 (Activation) ization_55[0][0]	(None, 16, 16, 240)	0	batch_normal
conv2d_57 (Conv2D) 5[0][0]	(None, 16, 16, 48)	11520	activation_5
batch_normalization_56 (BatchNo [0])	(None, 16, 16, 48)	192	conv2d_57[0]

activation_56 (Activation) ization_56[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_58 (Conv2D) 6[0][0]	(None, 16, 16, 12)	5184	activation_5
concatenate_27 (Concatenate) 26[0][0]	(None, 16, 16, 252)	0	concatenate_ conv2d_58[0]
batch_normalization_57 (BatchNo 27[0][0]	(None, 16, 16, 252)	1008	concatenate_
activation_57 (Activation) ization_57[0][0]	(None, 16, 16, 252)	0	batch_normal
conv2d_59 (Conv2D) 7[0][0]	(None, 16, 16, 48)	12096	activation_5
batch_normalization_58 (BatchNo [0]	(None, 16, 16, 48)	192	conv2d_59[0]
activation_58 (Activation) ization_58[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_60 (Conv2D) 8[0][0]	(None, 16, 16, 12)	5184	activation_5
concatenate_28 (Concatenate) 27[0][0]	(None, 16, 16, 264)	0	concatenate_ conv2d_60[0]
batch_normalization_59 (BatchNo 28[0][0]	(None, 16, 16, 264)	1056	concatenate_
activation_59 (Activation) ization_59[0][0]	(None, 16, 16, 264)	0	batch_normal
conv2d_61 (Conv2D) 9[0][0]	(None, 16, 16, 48)	12672	activation_5
batch_normalization_60 (BatchNo	(None, 16, 16, 48)	192	conv2d_61[0]

[0]

activation_60 (Activation) ization_60[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_62 (Conv2D) 0[0][0]	(None, 16, 16, 12)	5184	activation_6
concatenate_29 (Concatenate) 28[0][0]	(None, 16, 16, 276)	0	concatenate_ conv2d_62[0]
batch_normalization_61 (BatchNo 29[0][0])	(None, 16, 16, 276)	1104	concatenate_
activation_61 (Activation) ization_61[0][0]	(None, 16, 16, 276)	0	batch_normal
conv2d_63 (Conv2D) 1[0][0]	(None, 16, 16, 48)	13248	activation_6
batch_normalization_62 (BatchNo [0])	(None, 16, 16, 48)	192	conv2d_63[0]
activation_62 (Activation) ization_62[0][0]	(None, 16, 16, 48)	0	batch_normal
conv2d_64 (Conv2D) 2[0][0]	(None, 16, 16, 12)	5184	activation_6
concatenate_30 (Concatenate) 29[0][0]	(None, 16, 16, 288)	0	concatenate_ conv2d_64[0]
batch_normalization_63 (BatchNo 30[0][0])	(None, 16, 16, 288)	1152	concatenate_
activation_63 (Activation) ization_63[0][0]	(None, 16, 16, 288)	0	batch_normal
conv2d_65 (Conv2D) 3[0][0]	(None, 16, 16, 48)	13824	activation_6

batch_normalization_64 (BatchNo	(None, 16, 16, 48)	192	conv2d_65[0][0]
activation_64 (Activation)	(None, 16, 16, 48)	0	batch_normalization_64[0][0]
conv2d_66 (Conv2D)	(None, 16, 16, 12)	5184	activation_64[0][0]
concatenate_31 (Concatenate)	(None, 16, 16, 300)	0	concatenate_30[0][0]
			conv2d_66[0][0]
batch_normalization_65 (BatchNo	(None, 16, 16, 300)	1200	concatenate_31[0][0]
activation_65 (Activation)	(None, 16, 16, 300)	0	batch_normalization_65[0][0]
conv2d_67 (Conv2D)	(None, 16, 16, 150)	45000	activation_65[0][0]
average_pooling2d_1 (AveragePool)	(None, 8, 8, 150)	0	conv2d_67[0][0]
batch_normalization_66 (BatchNo	(None, 8, 8, 150)	600	average_pooling2d_1[0][0]
activation_66 (Activation)	(None, 8, 8, 150)	0	batch_normalization_66[0][0]
conv2d_68 (Conv2D)	(None, 8, 8, 48)	7200	activation_66[0][0]
batch_normalization_67 (BatchNo	(None, 8, 8, 48)	192	conv2d_68[0][0]
activation_67 (Activation)	(None, 8, 8, 48)	0	batch_normalization_67[0][0]
conv2d_69 (Conv2D)	(None, 8, 8, 12)	5184	activation_67[0][0]

7[0][0]

concatenate_32 (Concatenate) ing2d_1[0][0] [0]	(None, 8, 8, 162)	0	average_pool conv2d_69[0]
batch_normalization_68 (BatchNo 32[0][0]	(None, 8, 8, 162)	648	concatenate_
activation_68 (Activation) ization_68[0][0]	(None, 8, 8, 162)	0	batch_normal
conv2d_70 (Conv2D) 8[0][0]	(None, 8, 8, 48)	7776	activation_6
batch_normalization_69 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_70[0]
activation_69 (Activation) ization_69[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_71 (Conv2D) 9[0][0]	(None, 8, 8, 12)	5184	activation_6
concatenate_33 (Concatenate) 32[0][0] [0]	(None, 8, 8, 174)	0	concatenate_ conv2d_71[0]
batch_normalization_70 (BatchNo 33[0][0]	(None, 8, 8, 174)	696	concatenate_
activation_70 (Activation) ization_70[0][0]	(None, 8, 8, 174)	0	batch_normal
conv2d_72 (Conv2D) 0[0][0]	(None, 8, 8, 48)	8352	activation_7
batch_normalization_71 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_72[0]
activation_71 (Activation) ization_71[0][0]	(None, 8, 8, 48)	0	batch_normal

conv2d_73 (Conv2D) 1[0][0]	(None, 8, 8, 12)	5184	activation_7
concatenate_34 (Concatenate) 33[0][0] [0]	(None, 8, 8, 186)	0	concatenate_ conv2d_73[0]
batch_normalization_72 (BatchNo 34[0][0]	(None, 8, 8, 186)	744	concatenate_
activation_72 (Activation) ization_72[0][0]	(None, 8, 8, 186)	0	batch_normal
conv2d_74 (Conv2D) 2[0][0]	(None, 8, 8, 48)	8928	activation_7
batch_normalization_73 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_74[0]
activation_73 (Activation) ization_73[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_75 (Conv2D) 3[0][0]	(None, 8, 8, 12)	5184	activation_7
concatenate_35 (Concatenate) 34[0][0] [0]	(None, 8, 8, 198)	0	concatenate_ conv2d_75[0]
batch_normalization_74 (BatchNo 35[0][0]	(None, 8, 8, 198)	792	concatenate_
activation_74 (Activation) ization_74[0][0]	(None, 8, 8, 198)	0	batch_normal
conv2d_76 (Conv2D) 4[0][0]	(None, 8, 8, 48)	9504	activation_7
batch_normalization_75 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_76[0]

activation_75 (Activation) ization_75[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_77 (Conv2D) 5[0][0]	(None, 8, 8, 12)	5184	activation_7
concatenate_36 (Concatenate) 35[0][0]	(None, 8, 8, 210)	0	concatenate_ conv2d_77[0]
batch_normalization_76 (BatchNo 36[0][0]	(None, 8, 8, 210)	840	concatenate_
activation_76 (Activation) ization_76[0][0]	(None, 8, 8, 210)	0	batch_normal
conv2d_78 (Conv2D) 6[0][0]	(None, 8, 8, 48)	10080	activation_7
batch_normalization_77 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_78[0]
activation_77 (Activation) ization_77[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_79 (Conv2D) 7[0][0]	(None, 8, 8, 12)	5184	activation_7
concatenate_37 (Concatenate) 36[0][0]	(None, 8, 8, 222)	0	concatenate_ conv2d_79[0]
batch_normalization_78 (BatchNo 37[0][0]	(None, 8, 8, 222)	888	concatenate_
activation_78 (Activation) ization_78[0][0]	(None, 8, 8, 222)	0	batch_normal
conv2d_80 (Conv2D) 8[0][0]	(None, 8, 8, 48)	10656	activation_7

batch_normalization_79 (BatchNo	(None, 8, 8, 48)	192	conv2d_80[0][0]
activation_79 (Activation)	(None, 8, 8, 48)	0	batch_normalization_79[0][0]
conv2d_81 (Conv2D)	(None, 8, 8, 12)	5184	activation_79[0][0]
concatenate_38 (Concatenate)	(None, 8, 8, 234)	0	concatenate_37[0][0]
			conv2d_81[0][0]
batch_normalization_80 (BatchNo	(None, 8, 8, 234)	936	concatenate_38[0][0]
activation_80 (Activation)	(None, 8, 8, 234)	0	batch_normalization_80[0][0]
conv2d_82 (Conv2D)	(None, 8, 8, 48)	11232	activation_80[0][0]
batch_normalization_81 (BatchNo	(None, 8, 8, 48)	192	conv2d_82[0][0]
activation_81 (Activation)	(None, 8, 8, 48)	0	batch_normalization_81[0][0]
conv2d_83 (Conv2D)	(None, 8, 8, 12)	5184	activation_81[0][0]
concatenate_39 (Concatenate)	(None, 8, 8, 246)	0	concatenate_38[0][0]
			conv2d_83[0][0]
batch_normalization_82 (BatchNo	(None, 8, 8, 246)	984	concatenate_39[0][0]
activation_82 (Activation)	(None, 8, 8, 246)	0	batch_normalization_82[0][0]
conv2d_84 (Conv2D)	(None, 8, 8, 48)	11808	activation_82[0][0]

2[0][0]

batch_normalization_83 (BatchNo	(None, 8, 8, 48)	192	conv2d_84[0][0]
activation_83 (Activation)	(None, 8, 8, 48)	0	batch_normalization_83[0][0]
conv2d_85 (Conv2D)	(None, 8, 8, 12)	5184	activation_83[0][0]
concatenate_40 (Concatenate)	(None, 8, 8, 258)	0	concatenate_39[0][0]
			conv2d_85[0][0]
batch_normalization_84 (BatchNo	(None, 8, 8, 258)	1032	concatenate_40[0][0]
activation_84 (Activation)	(None, 8, 8, 258)	0	batch_normalization_84[0][0]
conv2d_86 (Conv2D)	(None, 8, 8, 48)	12384	activation_84[0][0]
batch_normalization_85 (BatchNo	(None, 8, 8, 48)	192	conv2d_86[0][0]
activation_85 (Activation)	(None, 8, 8, 48)	0	batch_normalization_85[0][0]
conv2d_87 (Conv2D)	(None, 8, 8, 12)	5184	activation_85[0][0]
concatenate_41 (Concatenate)	(None, 8, 8, 270)	0	concatenate_40[0][0]
			conv2d_87[0][0]
batch_normalization_86 (BatchNo	(None, 8, 8, 270)	1080	concatenate_41[0][0]
activation_86 (Activation)	(None, 8, 8, 270)	0	batch_normalization_86[0][0]

conv2d_88 (Conv2D)	(None, 8, 8, 48)	12960	activation_86[0][0]
batch_normalization_87 (Batch Normalization)	(None, 8, 8, 48)	192	conv2d_88[0][0]
activation_87 (Activation)	(None, 8, 8, 48)	0	batch_normalization_87[0][0]
conv2d_89 (Conv2D)	(None, 8, 8, 12)	5184	activation_87[0][0]
concatenate_42 (Concatenate)	(None, 8, 8, 282)	0	concatenate_41[0][0]
batch_normalization_88 (Batch Normalization)	(None, 8, 8, 282)	1128	concatenate_42[0][0]
activation_88 (Activation)	(None, 8, 8, 282)	0	batch_normalization_88[0][0]
conv2d_90 (Conv2D)	(None, 8, 8, 48)	13536	activation_88[0][0]
batch_normalization_89 (Batch Normalization)	(None, 8, 8, 48)	192	conv2d_90[0][0]
activation_89 (Activation)	(None, 8, 8, 48)	0	batch_normalization_89[0][0]
conv2d_91 (Conv2D)	(None, 8, 8, 12)	5184	activation_89[0][0]
concatenate_43 (Concatenate)	(None, 8, 8, 294)	0	concatenate_42[0][0]
batch_normalization_90 (Batch Normalization)	(None, 8, 8, 294)	1176	concatenate_43[0][0]

activation_90 (Activation) ization_90[0][0]	(None, 8, 8, 294)	0	batch_normal
conv2d_92 (Conv2D) 0[0][0]	(None, 8, 8, 48)	14112	activation_9
batch_normalization_91 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_92[0]
activation_91 (Activation) ization_91[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_93 (Conv2D) 1[0][0]	(None, 8, 8, 12)	5184	activation_9
concatenate_44 (Concatenate) 43[0][0]	(None, 8, 8, 306)	0	concatenate_ conv2d_93[0]
batch_normalization_92 (BatchNo 44[0][0]	(None, 8, 8, 306)	1224	concatenate_
activation_92 (Activation) ization_92[0][0]	(None, 8, 8, 306)	0	batch_normal
conv2d_94 (Conv2D) 2[0][0]	(None, 8, 8, 48)	14688	activation_9
batch_normalization_93 (BatchNo [0]	(None, 8, 8, 48)	192	conv2d_94[0]
activation_93 (Activation) ization_93[0][0]	(None, 8, 8, 48)	0	batch_normal
conv2d_95 (Conv2D) 3[0][0]	(None, 8, 8, 12)	5184	activation_9
concatenate_45 (Concatenate) 44[0][0]	(None, 8, 8, 318)	0	concatenate_ conv2d_95[0]
[0]			

batch_normalization_94 (BatchNo	(None, 8, 8, 318)	1272	concatenate_45[0][0]
activation_94 (Activation)	(None, 8, 8, 318)	0	batch_normalization_94[0][0]
conv2d_96 (Conv2D)	(None, 8, 8, 48)	15264	activation_94[0][0]
batch_normalization_95 (BatchNo	(None, 8, 8, 48)	192	conv2d_96[0][0]
activation_95 (Activation)	(None, 8, 8, 48)	0	batch_normalization_95[0][0]
conv2d_97 (Conv2D)	(None, 8, 8, 12)	5184	activation_95[0][0]
concatenate_46 (Concatenate)	(None, 8, 8, 330)	0	concatenate_45[0][0]
			conv2d_97[0][0]
batch_normalization_96 (BatchNo	(None, 8, 8, 330)	1320	concatenate_46[0][0]
activation_96 (Activation)	(None, 8, 8, 330)	0	batch_normalization_96[0][0]
conv2d_98 (Conv2D)	(None, 8, 8, 48)	15840	activation_96[0][0]
batch_normalization_97 (BatchNo	(None, 8, 8, 48)	192	conv2d_98[0][0]
activation_97 (Activation)	(None, 8, 8, 48)	0	batch_normalization_97[0][0]
conv2d_99 (Conv2D)	(None, 8, 8, 12)	5184	activation_97[0][0]
concatenate_47 (Concatenate)	(None, 8, 8, 342)	0	concatenate_46[0][0]
			conv2d_99[0][0]

[0]

batch_normalization_98 (BatchNo (None, 8, 8, 342)	1368	concatenate_47[0][0]
---	------	----------------------

global_average_pooling2d (Globa (None, 342)	0	batch_normalization_98[0][0]
---	---	------------------------------

dense (Dense)	(None, 10)	3430	global_average_pooling2d[0][0]
---------------	------------	------	--------------------------------

=====
 Total params: 793,150
 Trainable params: 769,162
 Non-trainable params: 23,988



```
In [0]: # sgd = tf.keras.optimizers.SGD(momentum=0.9,nesterov=True)

# from sklearn.metrics import roc_auc_score
# # https://stackoverflow.com/questions/41032551/how-to-compute-receiving-operating-characteristic-roc-and-auc-in-keras

# def auroc(y_true, y_pred):
#     return tf.py_func(roc_auc_score, (y_true, y_pred), tf.double)
```

```
In [0]: #callbacks
from time import time
from datetime import datetime

from tensorflow.python.keras.callbacks import TensorBoard

filepath = "weights.{epoch:02d}-{val_loss:.2f}.hdf5"
history = tf.keras.callbacks.History()

# tensorboard
tensorboard = TensorBoard(log_dir="model_logs/{}".format(time()))

filepath = "weights.{epoch:02d}-{val_loss:.2f}.hdf5"
learning_rate_reduction = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_acc',
                                                                patience=3,
                                                                verbose=1,
                                                                factor=0.5,
                                                                min_lr=0.0001)

checkpoint_save = tf.keras.callbacks.ModelCheckpoint(filepath, monitor='val_acc', verbose=1, save_best_only=True, mode='max')

callbacks_list = [checkpoint_save, learning_rate_reduction, history, tensorboard]
```

In [19]: `from keras.preprocessing.image import ImageDataGenerator`

```
train_datagen = ImageDataGenerator(
    zoom_range=0.3,
    rotation_range=15,
    horizontal_flip=True,
    rescale=1./255,
    fill_mode='nearest')

cv_datagen = ImageDataGenerator(rescale=1./255)
test_datagen = ImageDataGenerator(rescale=1./255)
```

Using TensorFlow backend.

In [0]: `train_datagen.fit(X_train)`
`cv_datagen.fit(X_cv)`
`test_datagen.fit(X_test)`

In [0]: *## using tensorboard instance for callbacks*
from time import time
from datetime import datetime
from tensorflow.python.keras.callbacks import TensorBoard

filepath = "weights.{epoch:02d}-{val_loss:.2f}.hdf5"
history = tf.keras.callbacks.History()

tensorboard
tensorboard = TensorBoard(log_dir="model_logs/{}".format(time()))

best model saving
model_check = tf.keras.callbacks.ModelCheckpoint(filepath, monitor='val_loss', verbose=1, save_best_only=True, save_weights_only=False, mode='auto', period=1)

reduce learning rate
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.1, patience=3, min_lr=0.0001)

W0813 09:44:20.662464 139758047876992 callbacks.py:875] `period` argument is deprecated. Please use `save_freq` to specify the frequency in number of samples seen.

In [0]: `epochs = 100`
`batch_size = 64`
`val_batch_size = 64`
`steps = len(y_train)//batch_size`
`val_steps = len(y_cv)//val_batch_size`

```
In [23]: model.fit_generator(train_datagen.flow(X_train, y_train, batch_size=64), steps_
per_epoch=steps,
                                epochs=100, callbacks=callbacks_list,
                                validation_data=cv_datagen.flow(X_cv, y_cv, batch_size=64), v
alidation_steps = val_steps)
```

```
Epoch 1/100
624/625 [=====>.] - ETA: 0s - loss: 3.2604 - acc: 0.41
24
Epoch 00001: val_acc improved from -inf to 0.38642, saving model to weights.0
1-3.60.hdf5
625/625 [=====] - 226s 362ms/step - loss: 3.2598 - a
cc: 0.4125 - val_loss: 3.6044 - val_acc: 0.3864
Epoch 2/100
624/625 [=====>.] - ETA: 0s - loss: 2.4873 - acc: 0.58
39
Epoch 00002: val_acc improved from 0.38642 to 0.48528, saving model to weight
s.02-2.66.hdf5
625/625 [=====] - 159s 255ms/step - loss: 2.4870 - a
cc: 0.5839 - val_loss: 2.6554 - val_acc: 0.4853
Epoch 3/100
624/625 [=====>.] - ETA: 0s - loss: 2.0091 - acc: 0.67
60
Epoch 00003: val_acc improved from 0.48528 to 0.57592, saving model to weight
s.03-2.20.hdf5
625/625 [=====] - 160s 256ms/step - loss: 2.0088 - a
cc: 0.6762 - val_loss: 2.1968 - val_acc: 0.5759
Epoch 4/100
624/625 [=====>.] - ETA: 0s - loss: 1.7002 - acc: 0.72
03
Epoch 00004: val_acc improved from 0.57592 to 0.68239, saving model to weight
s.04-1.78.hdf5
625/625 [=====] - 161s 257ms/step - loss: 1.6999 - a
cc: 0.7203 - val_loss: 1.7836 - val_acc: 0.6824
Epoch 5/100
624/625 [=====>.] - ETA: 0s - loss: 1.4817 - acc: 0.75
33
Epoch 00005: val_acc improved from 0.68239 to 0.74399, saving model to weight
s.05-1.45.hdf5
625/625 [=====] - 161s 257ms/step - loss: 1.4818 - a
cc: 0.7533 - val_loss: 1.4527 - val_acc: 0.7440
Epoch 6/100
624/625 [=====>.] - ETA: 0s - loss: 1.3299 - acc: 0.76
57
Epoch 00006: val_acc did not improve from 0.74399
625/625 [=====] - 160s 256ms/step - loss: 1.3300 - a
cc: 0.7656 - val_loss: 1.8016 - val_acc: 0.6072
Epoch 7/100
624/625 [=====>.] - ETA: 0s - loss: 1.2035 - acc: 0.78
33
Epoch 00007: val_acc did not improve from 0.74399
625/625 [=====] - 160s 256ms/step - loss: 1.2036 - a
cc: 0.7833 - val_loss: 1.3199 - val_acc: 0.7306
Epoch 8/100
624/625 [=====>.] - ETA: 0s - loss: 1.1205 - acc: 0.78
85
Epoch 00008: val_acc did not improve from 0.74399

Epoch 00008: ReduceLROnPlateau reducing learning rate to 0.05000000074505806.
625/625 [=====] - 161s 257ms/step - loss: 1.1204 - a
cc: 0.7885 - val_loss: 1.4653 - val_acc: 0.6929
Epoch 9/100
624/625 [=====>.] - ETA: 0s - loss: 0.9555 - acc: 0.83
```



```
18
Epoch 00009: val_acc improved from 0.74399 to 0.79928, saving model to weight
s.09-1.03.hdf5
625/625 [=====] - 160s 257ms/step - loss: 0.9554 - a
cc: 0.8318 - val_loss: 1.0333 - val_acc: 0.7993
Epoch 10/100
624/625 [=====>.] - ETA: 0s - loss: 0.8989 - acc: 0.83
86
Epoch 00010: val_acc did not improve from 0.79928
625/625 [=====] - 160s 256ms/step - loss: 0.8990 - a
cc: 0.8386 - val_loss: 1.3260 - val_acc: 0.7337
Epoch 11/100
624/625 [=====>.] - ETA: 0s - loss: 0.8688 - acc: 0.83
98
Epoch 00011: val_acc did not improve from 0.79928
625/625 [=====] - 160s 256ms/step - loss: 0.8691 - a
cc: 0.8396 - val_loss: 0.9800 - val_acc: 0.7991
Epoch 12/100
624/625 [=====>.] - ETA: 0s - loss: 0.8360 - acc: 0.84
46
Epoch 00012: val_acc improved from 0.79928 to 0.79998, saving model to weight
s.12-1.00.hdf5
625/625 [=====] - 161s 257ms/step - loss: 0.8360 - a
cc: 0.8446 - val_loss: 1.0038 - val_acc: 0.8000
Epoch 13/100
624/625 [=====>.] - ETA: 0s - loss: 0.8073 - acc: 0.84
87
Epoch 00013: val_acc did not improve from 0.79998
625/625 [=====] - 160s 256ms/step - loss: 0.8073 - a
cc: 0.8487 - val_loss: 1.1439 - val_acc: 0.7578
Epoch 14/100
624/625 [=====>.] - ETA: 0s - loss: 0.7871 - acc: 0.85
14
Epoch 00014: val_acc improved from 0.79998 to 0.80970, saving model to weight
s.14-0.94.hdf5
625/625 [=====] - 161s 257ms/step - loss: 0.7871 - a
cc: 0.8514 - val_loss: 0.9397 - val_acc: 0.8097
Epoch 15/100
624/625 [=====>.] - ETA: 0s - loss: 0.7736 - acc: 0.85
08
Epoch 00015: val_acc did not improve from 0.80970
625/625 [=====] - 160s 255ms/step - loss: 0.7736 - a
cc: 0.8507 - val_loss: 0.9367 - val_acc: 0.8097
Epoch 16/100
624/625 [=====>.] - ETA: 0s - loss: 0.7573 - acc: 0.85
35
Epoch 00016: val_acc did not improve from 0.80970
625/625 [=====] - 159s 255ms/step - loss: 0.7572 - a
cc: 0.8534 - val_loss: 0.9727 - val_acc: 0.7969
Epoch 17/100
624/625 [=====>.] - ETA: 0s - loss: 0.7424 - acc: 0.85
48
Epoch 00017: val_acc did not improve from 0.80970

Epoch 00017: ReduceLROnPlateau reducing learning rate to 0.02500000037252903.
625/625 [=====] - 159s 255ms/step - loss: 0.7422 - a
cc: 0.8549 - val_loss: 0.9129 - val_acc: 0.7983
```

Epoch 18/100
624/625 [=====>.] - ETA: 0s - loss: 0.6451 - acc: 0.8870
Epoch 00018: val_acc improved from 0.80970 to 0.86008, saving model to weights.18-0.73.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.6453 - acc: 0.8869 - val_loss: 0.7349 - val_acc: 0.8601
Epoch 19/100
624/625 [=====>.] - ETA: 0s - loss: 0.6214 - acc: 0.8913
Epoch 00019: val_acc improved from 0.86008 to 0.87230, saving model to weights.19-0.68.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.6215 - acc: 0.8912 - val_loss: 0.6831 - val_acc: 0.8723
Epoch 20/100
624/625 [=====>.] - ETA: 0s - loss: 0.6040 - acc: 0.8932
Epoch 00020: val_acc did not improve from 0.87230
625/625 [=====] - 159s 254ms/step - loss: 0.6042 - acc: 0.8932 - val_loss: 0.7740 - val_acc: 0.8456
Epoch 21/100
624/625 [=====>.] - ETA: 0s - loss: 0.5881 - acc: 0.8952
Epoch 00021: val_acc did not improve from 0.87230
625/625 [=====] - 159s 254ms/step - loss: 0.5879 - acc: 0.8953 - val_loss: 0.8960 - val_acc: 0.8137
Epoch 22/100
624/625 [=====>.] - ETA: 0s - loss: 0.5812 - acc: 0.8980
Epoch 00022: val_acc improved from 0.87230 to 0.87350, saving model to weights.22-0.67.hdf5
625/625 [=====] - 160s 255ms/step - loss: 0.5813 - acc: 0.8980 - val_loss: 0.6677 - val_acc: 0.8735
Epoch 23/100
624/625 [=====>.] - ETA: 0s - loss: 0.5720 - acc: 0.8986
Epoch 00023: val_acc did not improve from 0.87350
625/625 [=====] - 159s 254ms/step - loss: 0.5718 - acc: 0.8987 - val_loss: 0.6725 - val_acc: 0.8632
Epoch 24/100
624/625 [=====>.] - ETA: 0s - loss: 0.5679 - acc: 0.8977
Epoch 00024: val_acc did not improve from 0.87350
625/625 [=====] - 159s 254ms/step - loss: 0.5679 - acc: 0.8978 - val_loss: 0.7070 - val_acc: 0.8615
Epoch 25/100
624/625 [=====>.] - ETA: 0s - loss: 0.5623 - acc: 0.8991
Epoch 00025: val_acc did not improve from 0.87350

Epoch 00025: ReduceLROnPlateau reducing learning rate to 0.012500000186264515.
625/625 [=====] - 159s 255ms/step - loss: 0.5621 - acc: 0.8991 - val_loss: 0.7965 - val_acc: 0.8350
Epoch 26/100
624/625 [=====>.] - ETA: 0s - loss: 0.4951 - acc: 0.9204

Epoch 00026: val_acc improved from 0.87350 to 0.88822, saving model to weights.26-0.60.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.4951 - acc: 0.9204 - val_loss: 0.6030 - val_acc: 0.8882
Epoch 27/100
624/625 [=====>.] - ETA: 0s - loss: 0.4688 - acc: 0.9282
Epoch 00027: val_acc improved from 0.88822 to 0.89032, saving model to weights.27-0.62.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.4688 - acc: 0.9282 - val_loss: 0.6220 - val_acc: 0.8903
Epoch 28/100
624/625 [=====>.] - ETA: 0s - loss: 0.4652 - acc: 0.9285
Epoch 00028: val_acc did not improve from 0.89032
625/625 [=====] - 159s 255ms/step - loss: 0.4652 - acc: 0.9286 - val_loss: 0.6632 - val_acc: 0.8759
Epoch 29/100
624/625 [=====>.] - ETA: 0s - loss: 0.4564 - acc: 0.9291
Epoch 00029: val_acc did not improve from 0.89032
625/625 [=====] - 159s 254ms/step - loss: 0.4565 - acc: 0.9291 - val_loss: 0.6051 - val_acc: 0.8899
Epoch 30/100
624/625 [=====>.] - ETA: 0s - loss: 0.4526 - acc: 0.9296
Epoch 00030: val_acc improved from 0.89032 to 0.89944, saving model to weights.30-0.55.hdf5
625/625 [=====] - 159s 255ms/step - loss: 0.4526 - acc: 0.9295 - val_loss: 0.5517 - val_acc: 0.8994
Epoch 31/100
624/625 [=====>.] - ETA: 0s - loss: 0.4445 - acc: 0.9303
Epoch 00031: val_acc improved from 0.89944 to 0.91086, saving model to weights.31-0.54.hdf5
625/625 [=====] - 159s 255ms/step - loss: 0.4446 - acc: 0.9303 - val_loss: 0.5391 - val_acc: 0.9109
Epoch 32/100
624/625 [=====>.] - ETA: 0s - loss: 0.4431 - acc: 0.9315
Epoch 00032: val_acc did not improve from 0.91086
625/625 [=====] - 159s 254ms/step - loss: 0.4430 - acc: 0.9315 - val_loss: 0.6024 - val_acc: 0.8887
Epoch 33/100
624/625 [=====>.] - ETA: 0s - loss: 0.4406 - acc: 0.9301
Epoch 00033: val_acc did not improve from 0.91086
625/625 [=====] - 159s 254ms/step - loss: 0.4407 - acc: 0.9300 - val_loss: 0.6307 - val_acc: 0.8817
Epoch 34/100
624/625 [=====>.] - ETA: 0s - loss: 0.4316 - acc: 0.9332
Epoch 00034: val_acc did not improve from 0.91086

Epoch 00034: ReduceLROnPlateau reducing learning rate to 0.0062500000931322575.
625/625 [=====] - 159s 254ms/step - loss: 0.4317 - a

```
cc: 0.9332 - val_loss: 0.6349 - val_acc: 0.8808
Epoch 35/100
624/625 [=====>.] - ETA: 0s - loss: 0.3892 - acc: 0.9489
Epoch 00035: val_acc improved from 0.91086 to 0.91346, saving model to weight
s.35-0.51.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.3890 - a
cc: 0.9490 - val_loss: 0.5112 - val_acc: 0.9135
Epoch 36/100
624/625 [=====>.] - ETA: 0s - loss: 0.3741 - acc: 0.9522
Epoch 00036: val_acc improved from 0.91346 to 0.91556, saving model to weight
s.36-0.51.hdf5
625/625 [=====] - 159s 255ms/step - loss: 0.3740 - a
cc: 0.9522 - val_loss: 0.5088 - val_acc: 0.9156
Epoch 37/100
624/625 [=====>.] - ETA: 0s - loss: 0.3655 - acc: 0.9536
Epoch 00037: val_acc improved from 0.91556 to 0.92578, saving model to weight
s.37-0.47.hdf5
625/625 [=====] - 160s 255ms/step - loss: 0.3655 - a
cc: 0.9535 - val_loss: 0.4690 - val_acc: 0.9258
Epoch 38/100
624/625 [=====>.] - ETA: 0s - loss: 0.3687 - acc: 0.9518
Epoch 00038: val_acc did not improve from 0.92578
625/625 [=====] - 159s 254ms/step - loss: 0.3687 - a
cc: 0.9519 - val_loss: 0.4885 - val_acc: 0.9195
Epoch 39/100
624/625 [=====>.] - ETA: 0s - loss: 0.3539 - acc: 0.9551
Epoch 00039: val_acc did not improve from 0.92578
625/625 [=====] - 159s 255ms/step - loss: 0.3538 - a
cc: 0.9551 - val_loss: 0.5563 - val_acc: 0.8995
Epoch 40/100
624/625 [=====>.] - ETA: 0s - loss: 0.3540 - acc: 0.9558
Epoch 00040: val_acc did not improve from 0.92578

Epoch 00040: ReduceLROnPlateau reducing learning rate to 0.003125000046566128
7.
625/625 [=====] - 159s 254ms/step - loss: 0.3541 - a
cc: 0.9558 - val_loss: 0.4972 - val_acc: 0.9164
Epoch 41/100
624/625 [=====>.] - ETA: 0s - loss: 0.3330 - acc: 0.9620
Epoch 00041: val_acc improved from 0.92578 to 0.92738, saving model to weight
s.41-0.47.hdf5
625/625 [=====] - 160s 255ms/step - loss: 0.3331 - a
cc: 0.9620 - val_loss: 0.4677 - val_acc: 0.9274
Epoch 42/100
624/625 [=====>.] - ETA: 0s - loss: 0.3233 - acc: 0.9670
Epoch 00042: val_acc improved from 0.92738 to 0.92748, saving model to weight
s.42-0.46.hdf5
625/625 [=====] - 160s 256ms/step - loss: 0.3234 - a
cc: 0.9670 - val_loss: 0.4644 - val_acc: 0.9275
```

Epoch 43/100
624/625 [=====>.] - ETA: 0s - loss: 0.3186 - acc: 0.9681
Epoch 00043: val_acc did not improve from 0.92748
625/625 [=====] - 159s 254ms/step - loss: 0.3186 - acc: 0.9681 - val_loss: 0.4770 - val_acc: 0.9222
Epoch 44/100
624/625 [=====>.] - ETA: 0s - loss: 0.3136 - acc: 0.9691
Epoch 00044: val_acc did not improve from 0.92748
625/625 [=====] - 159s 255ms/step - loss: 0.3139 - acc: 0.9689 - val_loss: 0.4715 - val_acc: 0.9267
Epoch 45/100
624/625 [=====>.] - ETA: 0s - loss: 0.3106 - acc: 0.9690
Epoch 00045: val_acc improved from 0.92748 to 0.92929, saving model to weights.45-0.46.hdf5
625/625 [=====] - 161s 257ms/step - loss: 0.3105 - acc: 0.9689 - val_loss: 0.4577 - val_acc: 0.9293
Epoch 46/100
624/625 [=====>.] - ETA: 0s - loss: 0.3056 - acc: 0.9703
Epoch 00046: val_acc did not improve from 0.92929
625/625 [=====] - 160s 256ms/step - loss: 0.3056 - acc: 0.9703 - val_loss: 0.4708 - val_acc: 0.9260
Epoch 47/100
624/625 [=====>.] - ETA: 0s - loss: 0.3038 - acc: 0.9715
Epoch 00047: val_acc did not improve from 0.92929
625/625 [=====] - 161s 257ms/step - loss: 0.3039 - acc: 0.9714 - val_loss: 0.4649 - val_acc: 0.9287
Epoch 48/100
624/625 [=====>.] - ETA: 0s - loss: 0.3028 - acc: 0.9706
Epoch 00048: val_acc did not improve from 0.92929

Epoch 00048: ReduceLROnPlateau reducing learning rate to 0.0015625000232830644.
625/625 [=====] - 162s 258ms/step - loss: 0.3027 - acc: 0.9706 - val_loss: 0.4879 - val_acc: 0.9237
Epoch 49/100
624/625 [=====>.] - ETA: 0s - loss: 0.2926 - acc: 0.9745
Epoch 00049: val_acc improved from 0.92929 to 0.93239, saving model to weights.49-0.45.hdf5
625/625 [=====] - 162s 259ms/step - loss: 0.2926 - acc: 0.9745 - val_loss: 0.4547 - val_acc: 0.9324
Epoch 50/100
624/625 [=====>.] - ETA: 0s - loss: 0.2860 - acc: 0.9757
Epoch 00050: val_acc did not improve from 0.93239
625/625 [=====] - 161s 258ms/step - loss: 0.2860 - acc: 0.9757 - val_loss: 0.4502 - val_acc: 0.9321
Epoch 51/100
624/625 [=====>.] - ETA: 0s - loss: 0.2845 - acc: 0.9767
Epoch 00051: val_acc did not improve from 0.93239

```
625/625 [=====] - 161s 258ms/step - loss: 0.2845 - a
cc: 0.9767 - val_loss: 0.4542 - val_acc: 0.9304
Epoch 52/100
624/625 [=====>.] - ETA: 0s - loss: 0.2836 - acc: 0.97
59
Epoch 00052: val_acc improved from 0.93239 to 0.93319, saving model to weight
s.52-0.45.hdf5
625/625 [=====] - 162s 259ms/step - loss: 0.2836 - a
cc: 0.9759 - val_loss: 0.4459 - val_acc: 0.9332
Epoch 53/100
624/625 [=====>.] - ETA: 0s - loss: 0.2817 - acc: 0.97
70
Epoch 00053: val_acc did not improve from 0.93319
625/625 [=====] - 162s 259ms/step - loss: 0.2816 - a
cc: 0.9771 - val_loss: 0.4455 - val_acc: 0.9323
Epoch 54/100
624/625 [=====>.] - ETA: 0s - loss: 0.2796 - acc: 0.97
70
Epoch 00054: val_acc did not improve from 0.93319
625/625 [=====] - 162s 260ms/step - loss: 0.2796 - a
cc: 0.9770 - val_loss: 0.4443 - val_acc: 0.9328
Epoch 55/100
624/625 [=====>.] - ETA: 0s - loss: 0.2788 - acc: 0.97
71
Epoch 00055: val_acc did not improve from 0.93319

Epoch 00055: ReduceLROnPlateau reducing learning rate to 0.000781250011641532
2.
625/625 [=====] - 162s 259ms/step - loss: 0.2788 - a
cc: 0.9771 - val_loss: 0.4601 - val_acc: 0.9291
Epoch 56/100
624/625 [=====>.] - ETA: 0s - loss: 0.2705 - acc: 0.98
08
Epoch 00056: val_acc did not improve from 0.93319
625/625 [=====] - 162s 260ms/step - loss: 0.2706 - a
cc: 0.9808 - val_loss: 0.4507 - val_acc: 0.9327
Epoch 57/100
624/625 [=====>.] - ETA: 0s - loss: 0.2692 - acc: 0.98
06
Epoch 00057: val_acc improved from 0.93319 to 0.93389, saving model to weight
s.57-0.44.hdf5
625/625 [=====] - 163s 261ms/step - loss: 0.2692 - a
cc: 0.9806 - val_loss: 0.4450 - val_acc: 0.9339
Epoch 58/100
624/625 [=====>.] - ETA: 0s - loss: 0.2687 - acc: 0.98
10
Epoch 00058: val_acc improved from 0.93389 to 0.93440, saving model to weight
s.58-0.44.hdf5
625/625 [=====] - 163s 261ms/step - loss: 0.2688 - a
cc: 0.9810 - val_loss: 0.4426 - val_acc: 0.9344
Epoch 59/100
624/625 [=====>.] - ETA: 0s - loss: 0.2679 - acc: 0.98
16
Epoch 00059: val_acc did not improve from 0.93440
625/625 [=====] - 163s 261ms/step - loss: 0.2680 - a
cc: 0.9816 - val_loss: 0.4425 - val_acc: 0.9343
Epoch 60/100
```

```
624/625 [=====>.] - ETA: 0s - loss: 0.2657 - acc: 0.98
19
Epoch 00060: val_acc did not improve from 0.93440
625/625 [=====] - 163s 261ms/step - loss: 0.2657 - a
cc: 0.9819 - val_loss: 0.4443 - val_acc: 0.9334
Epoch 61/100
624/625 [=====>.] - ETA: 0s - loss: 0.2647 - acc: 0.98
22
Epoch 00061: val_acc improved from 0.93440 to 0.93530, saving model to weight
s.61-0.44.hdf5
625/625 [=====] - 164s 262ms/step - loss: 0.2649 - a
cc: 0.9821 - val_loss: 0.4443 - val_acc: 0.9353
Epoch 62/100
624/625 [=====>.] - ETA: 0s - loss: 0.2647 - acc: 0.98
19
Epoch 00062: val_acc did not improve from 0.93530
625/625 [=====] - 163s 260ms/step - loss: 0.2646 - a
cc: 0.9819 - val_loss: 0.4464 - val_acc: 0.9337
Epoch 63/100
624/625 [=====>.] - ETA: 0s - loss: 0.2637 - acc: 0.98
21
Epoch 00063: val_acc did not improve from 0.93530
625/625 [=====] - 163s 261ms/step - loss: 0.2637 - a
cc: 0.9821 - val_loss: 0.4413 - val_acc: 0.9349
Epoch 64/100
624/625 [=====>.] - ETA: 0s - loss: 0.2628 - acc: 0.98
22
Epoch 00064: val_acc did not improve from 0.93530

Epoch 00064: ReduceLROnPlateau reducing learning rate to 0.000390625005820766
1.
625/625 [=====] - 163s 262ms/step - loss: 0.2628 - a
cc: 0.9822 - val_loss: 0.4404 - val_acc: 0.9350
Epoch 65/100
624/625 [=====>.] - ETA: 0s - loss: 0.2587 - acc: 0.98
40
Epoch 00065: val_acc improved from 0.93530 to 0.93560, saving model to weight
s.65-0.44.hdf5
625/625 [=====] - 164s 263ms/step - loss: 0.2588 - a
cc: 0.9840 - val_loss: 0.4414 - val_acc: 0.9356
Epoch 66/100
624/625 [=====>.] - ETA: 0s - loss: 0.2619 - acc: 0.98
28
Epoch 00066: val_acc did not improve from 0.93560
625/625 [=====] - 164s 262ms/step - loss: 0.2619 - a
cc: 0.9828 - val_loss: 0.4400 - val_acc: 0.9356
Epoch 67/100
624/625 [=====>.] - ETA: 0s - loss: 0.2617 - acc: 0.98
29
Epoch 00067: val_acc improved from 0.93560 to 0.93570, saving model to weight
s.67-0.44.hdf5
625/625 [=====] - 164s 262ms/step - loss: 0.2618 - a
cc: 0.9829 - val_loss: 0.4406 - val_acc: 0.9357
Epoch 68/100
624/625 [=====>.] - ETA: 0s - loss: 0.2569 - acc: 0.98
45
Epoch 00068: val_acc did not improve from 0.93570
```

```
625/625 [=====] - 163s 261ms/step - loss: 0.2568 - a
cc: 0.9845 - val_loss: 0.4432 - val_acc: 0.9346
Epoch 69/100
624/625 [=====>.] - ETA: 0s - loss: 0.2575 - acc: 0.98
43
Epoch 00069: val_acc did not improve from 0.93570
625/625 [=====] - 164s 262ms/step - loss: 0.2575 - a
cc: 0.9843 - val_loss: 0.4443 - val_acc: 0.9350
Epoch 70/100
624/625 [=====>.] - ETA: 0s - loss: 0.2590 - acc: 0.98
31
Epoch 00070: val_acc did not improve from 0.93570

Epoch 00070: ReduceLROnPlateau reducing learning rate to 0.000195312502910383
05.
625/625 [=====] - 163s 261ms/step - loss: 0.2590 - a
cc: 0.9831 - val_loss: 0.4449 - val_acc: 0.9347
Epoch 71/100
624/625 [=====>.] - ETA: 0s - loss: 0.2572 - acc: 0.98
38
Epoch 00071: val_acc did not improve from 0.93570
625/625 [=====] - 163s 261ms/step - loss: 0.2571 - a
cc: 0.9839 - val_loss: 0.4426 - val_acc: 0.9345
Epoch 72/100
624/625 [=====>.] - ETA: 0s - loss: 0.2568 - acc: 0.98
38
Epoch 00072: val_acc did not improve from 0.93570
625/625 [=====] - 163s 261ms/step - loss: 0.2568 - a
cc: 0.9838 - val_loss: 0.4415 - val_acc: 0.9351
Epoch 73/100
624/625 [=====>.] - ETA: 0s - loss: 0.2574 - acc: 0.98
38
Epoch 00073: val_acc did not improve from 0.93570

Epoch 00073: ReduceLROnPlateau reducing learning rate to 0.0001.
625/625 [=====] - 163s 262ms/step - loss: 0.2574 - a
cc: 0.9838 - val_loss: 0.4427 - val_acc: 0.9347
Epoch 74/100
624/625 [=====>.] - ETA: 0s - loss: 0.2582 - acc: 0.98
35
Epoch 00074: val_acc did not improve from 0.93570
625/625 [=====] - 163s 261ms/step - loss: 0.2581 - a
cc: 0.9836 - val_loss: 0.4429 - val_acc: 0.9350
Epoch 75/100
624/625 [=====>.] - ETA: 0s - loss: 0.2553 - acc: 0.98
44
Epoch 00075: val_acc did not improve from 0.93570
625/625 [=====] - 163s 261ms/step - loss: 0.2553 - a
cc: 0.9844 - val_loss: 0.4435 - val_acc: 0.9350
Epoch 76/100
624/625 [=====>.] - ETA: 0s - loss: 0.2547 - acc: 0.98
53
Epoch 00076: val_acc did not improve from 0.93570
625/625 [=====] - 163s 261ms/step - loss: 0.2548 - a
cc: 0.9852 - val_loss: 0.4422 - val_acc: 0.9354
Epoch 77/100
```


62/625 [=>.....] - ETA: 2:18 - loss: 0.2558 - acc: 0.
9846

```

-----
KeyboardInterrupt                                Traceback (most recent call last)
<ipython-input-23-75f3d03121f9> in <module>()
      1 model.fit_generator(train_datagen.flow(X_train, y_train, batch_size=6
4), steps_per_epoch=steps,
      2                                epochs=100, callbacks=callbacks_list,
----> 3                                validation_data=cv_datagen.flow(X_cv, y_cv, batch_s
ize=64), validation_steps = val_steps)

/usr/local/lib/python3.6/dist-packages/tensorflow/python/keras/engine/trainin
g.py in fit_generator(self, generator, steps_per_epoch, epochs, verbose, call
backs, validation_data, validation_steps, validation_freq, class_weight, max_
queue_size, workers, use_multiprocessing, shuffle, initial_epoch)
    1431         shuffle=shuffle,
    1432         initial_epoch=initial_epoch,
-> 1433         steps_name='steps_per_epoch')
    1434
    1435     def evaluate_generator(self,

/usr/local/lib/python3.6/dist-packages/tensorflow/python/keras/engine/trainin
g_generator.py in model_iteration(model, data, steps_per_epoch, epochs, verbo
se, callbacks, validation_data, validation_steps, validation_freq, class_weig
ht, max_queue_size, workers, use_multiprocessing, shuffle, initial_epoch, mod
e, batch_size, steps_name, **kwargs)
    262
    263         is_deferred = not model._is_compiled
-> 264         batch_outs = batch_function(*batch_data)
    265         if not isinstance(batch_outs, list):
    266             batch_outs = [batch_outs]

/usr/local/lib/python3.6/dist-packages/tensorflow/python/keras/engine/trainin
g.py in train_on_batch(self, x, y, sample_weight, class_weight, reset_metric
s)
    1173         self._update_sample_weight_modes(sample_weights=sample_weights)
    1174         self._make_train_function()
-> 1175         outputs = self.train_function(ins) # pylint: disable=not-calla
ble
    1176
    1177         if reset_metrics:

/usr/local/lib/python3.6/dist-packages/tensorflow/python/keras/backend.py in
__call__(self, inputs)
    3290
    3291         fetched = self._callable_fn(*array_vals,
-> 3292                                     run_metadata=self.run_metadata)
    3293         self._call_fetch_callbacks(fetched[-len(self._fetches):])
    3294         output_structure = nest.pack_sequence_as(

/usr/local/lib/python3.6/dist-packages/tensorflow/python/client/session.py in
__call__(self, *args, **kwargs)
    1456         ret = tf_session.TF_SessionRunCallable(self._session._sessio
n,
    1457                                                     self._handle, args,
-> 1458                                                     run_metadata_ptr)
    1459         if run_metadata:
    1460             proto_data = tf_session.TF_GetBuffer(run_metadata_ptr)

```

KeyboardInterrupt:

```
In [0]: # https://stackoverflow.com/questions/50127257/is-there-any-way-to-stop-training-a-model-in-keras-after-a-certain-accuracy-has

# callbacks = [
#     EarlyStoppingByAccuracy(monitor='accuracy', value=0.98, verbose=1),
#     ModelCheckpoint(kfold_weights_path, monitor='val_loss', save_best_only=True, verbose=0),
# ]
# model.fit(X_train.astype('float32'), Y_train, batch_size=batch_size, nb_epoch=nb_epoch,
#           shuffle=True, verbose=1, validation_data=(X_valid, Y_valid),
#           callbacks=callbacks)
```

Loss And Accuracy Plots

Download tensorboard logs

```
In [36]: !zip -r /content/logs.zip /content/model_logs

adding: content/model_logs/ (stored 0%)
adding: content/model_logs/1565714363.7803748/ (stored 0%)
adding: content/model_logs/1565714363.7803748/train/ (stored 0%)
adding: content/model_logs/1565714363.7803748/train/events.out.tfevents.1565714551.4683528c7227.137.14507.v2 (deflated 95%)
adding: content/model_logs/1565714363.7803748/validation/ (stored 0%)
adding: content/model_logs/1565714363.7803748/validation/events.out.tfevents.1565714781.4683528c7227.137.28198.v2 (deflated 65%)
```

In [42]: `!wget https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip`
`!unzip ngrok-stable-linux-amd64.zip`

```
--2019-08-13 20:42:35-- https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip
Resolving bin.equinox.io (bin.equinox.io)... 34.196.237.103, 52.45.111.123, 52.201.75.180, ...
Connecting to bin.equinox.io (bin.equinox.io)|34.196.237.103|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13607069 (13M) [application/octet-stream]
Saving to: 'ngrok-stable-linux-amd64.zip'
```

```
ngrok-stable-linux- 100%[=====>] 12.98M 31.6MB/s in 0.4s
```

```
2019-08-13 20:42:35 (31.6 MB/s) - 'ngrok-stable-linux-amd64.zip' saved [13607069/13607069]
```

```
Archive: ngrok-stable-linux-amd64.zip
  inflating: ngrok
```

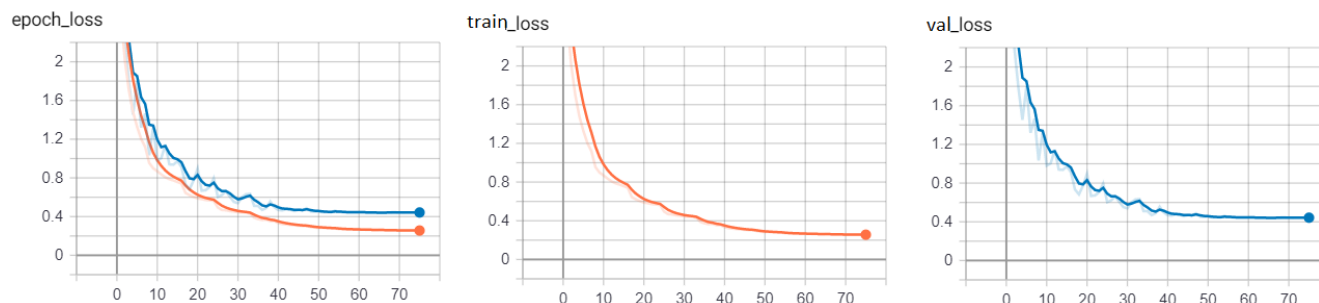


In [0]: `LOG_DIR = '/content/model_logs'`
`get_ipython().system_raw(`
 `'tensorboard --logdir {} --host 0.0.0.0 --port 6006 &'`
 `.format(LOG_DIR)`
`)`
`get_ipython().system_raw('./ngrok http 6006 &')`

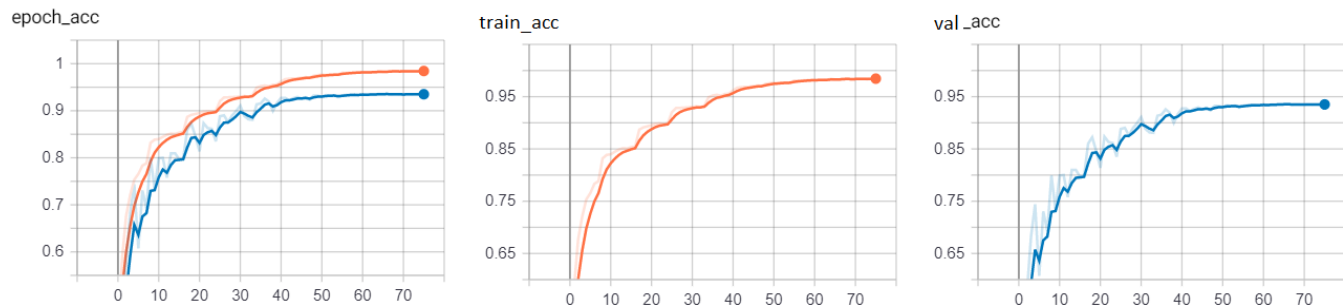
In [45]: `!curl -s http://localhost:4040/api/tunnels | python3 -c "`
`import sys, json; print(json.load(sys.stdin)['tunnels'][0]['public_url'])"`

<https://aacd44cd.ngrok.io>

Train and Test Loss



Train and Test Accuracy



Test Accuracy

```
In [24]: # evaluate on test data
score = model.evaluate_generator(test_datagen.flow(X_test, y_test, batch_size=
64), verbose=1)
```

```
157/157 [=====] - 10s 64ms/step - loss: 0.4492 - ac
c: 0.9269
```

```
In [25]: print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

```
Test loss: 0.4492105321519694
Test accuracy: 0.9269
```

Confusion Matrix

```

In [0]: import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
import itertools
import seaborn as sns
%matplotlib inline
# Look at confusion matrix
#Note, this code is taken straight from the SKLEARN website, an nice way of vi
ewing confusion matrix.
def plot_confusion_matrix(cm, classes,
                           normalize=False,
                           title='Confusion matrix',
                           cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]

    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, cm[i, j],
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')

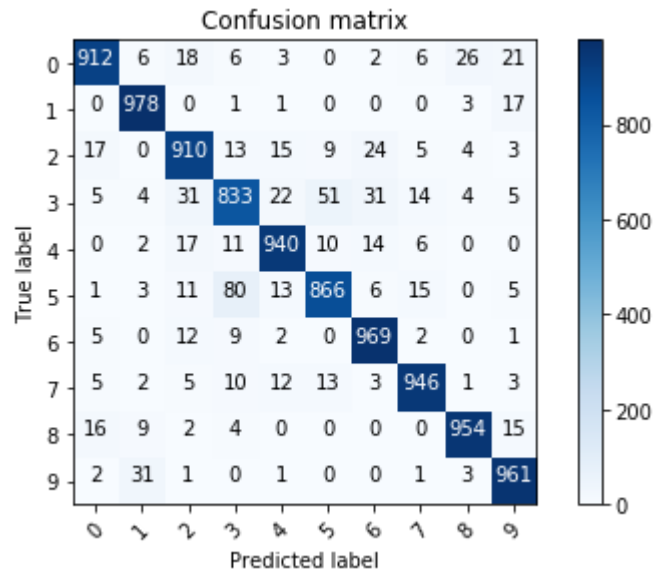
```

```

In [0]: import numpy as np
# Predict the values from the validation dataset
X_test = X_test.astype('float32')
X_test /= 255

```

```
In [34]: Y_pred = model.predict(X_test)
# Convert predictions classes to one hot vectors
Y_pred_classes = np.argmax(Y_pred, axis = 1)
# Convert validation observations to one hot vectors
Y_true = np.argmax(y_test, axis = 1)
# compute the confusion matrix
confusion_mtx = confusion_matrix(Y_true, Y_pred_classes)
# plot the confusion matrix
plot_confusion_matrix(confusion_mtx, classes = range(10))
```



Heat Map

```

In [35]: from sklearn.metrics import classification_report, confusion_matrix
import matplotlib.pyplot as plt
%matplotlib inline
Y_pred = model.predict(X_test, verbose=2)
y_pred = np.argmax(Y_pred, axis=1)

cm = confusion_matrix(np.argmax(y_test,axis=1),
                      y_pred)

import seaborn as sn
import pandas as pd

df_cm = pd.DataFrame(cm, range(10),
                     range(10))
plt.figure(figsize = (10,7))
sn.set(font_scale=1.4)#for label size
sn.heatmap(df_cm, annot=True,annot_kws={"size": 12})# font size

```

10000/10000 - 8s

Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7fa9532fd0>

