part_1

Model building: Brain MRI Segmentation

```
so in model building stage we will use UNET models as they are popular for the sematic segmantation for the medical problem.

here we mainly focus on UNET and try to use different different variation of  UNET for example - Unet with resnet34, etc.

so instead of creating archictecture for all and train from scratch which is too much time consuming

we will use consept of transfer learning .
for the context - transfer learning : is a concept where previously trained model is reused to solve the similar type of problem. h


so there is libary "segmantation model" which provides multiple pretrained model with different-2 variation of UNET .
to install this library and see how it work - click me

there are multiple backbone(variation) and with some research
we have listed our priorities as follows which we train our tasks:

1. UNET with resnet34  ( basemodel )
2. UNET with inceptionv3
3. UNet with EfficientNet
4. UNet with senet154
5. UNet with senet

apart from using pretrained model, we will also train some model from scrach so
```

```
that we can compare what is diffence among them.(in next upcoming notebook)
```

table of content

Table of contents

1. Dependencies

```
# import pickle

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from os import path
import cv2
import os
import re
import random
import pdb
import seaborn as sns

import tensorflow as tf
import keras

!pip install -U segmentation-models==1.0.1
from keras.utils.layer_utils import get_source_inputs
import segmentation_models as sm
sm.set_framework('tf.keras')
tf.keras.backend.set_image_data_format('channels_last')
from segmentation_models import Unet

from segmentation_models.metrics import iou_score
from segmentation_models.losses import DiceLoss

import pickle
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting segmentation-models==1.0.1
  Downloading segmentation_models-1.0.1-py3-none-any.whl (33 kB)
Collecting keras-applications<=1.0.8,>=1.0.7
  Downloading Keras_Applications-1.0.8-py3-none-any.whl (50 kB)
     |████████████████████████████████| 50 kB 5.5 MB/s
Collecting image-classifiers==1.0.0
  Downloading image_classifiers-1.0.0-py3-none-any.whl (19 kB)
Collecting efficientnet==1.0.0
  Downloading efficientnet-1.0.0-py3-none-any.whl (17 kB)
```

```
Requirement already satisfied: scikit-image in /usr/local/lib/python3.7/dist-packages (from efficientnet==1.0.0->segmentation-m
Requirement already satisfied: h5py in /usr/local/lib/python3.7/dist-packages (from keras-applications<=1.0.8,>=1.0.7->segmenta
Requirement already satisfied: numpy>=1.9.1 in /usr/local/lib/python3.7/dist-packages (from keras-applications<=1.0.8,>=1.0.7->
Requirement already satisfied: cached-property in /usr/local/lib/python3.7/dist-packages (from h5py->keras-applications<=1.0.8,
Requirement already satisfied: scipy>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficientnet==1.0.0->
Requirement already satisfied: imageio>=2.3.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficientnet==1.0.0
Requirement already satisfied: tifffile>=2019.7.26 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficientnet==
Requirement already satisfied: networkx>=2.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficientnet==1.0.0-
Requirement already satisfied: pillow!=7.1.0,!=7.1.1,>=4.3.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->effi
Requirement already satisfied: matplotlib!=3.0.0,>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficien
Requirement already satisfied: PyWavelets>=1.1.1 in /usr/local/lib/python3.7/dist-packages (from scikit-image->efficientnet==1.
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib!=3.0.0,>=2.0.0->sci
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages (from matplot
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib!=3.0.0,>=2.0.0->
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib!=3.0.0,>=2.0.0->scikit-i
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packages (from kiwisolver>=1.0.1->matplotlib!
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.1->matplotlib!=3.0.0
Installing collected packages: keras-applications, image-classifiers, efficientnet, segmentation-models
Successfully installed efficientnet-1.0.0 image-classifiers-1.0.0 keras-applications-1.0.8 segmentation-models-1.0.1
Segmentation Models: using `keras` framework.
```

```
# https://www.kaggle.com/datasets/mateuszbuda/lgg-mri-segmentation
!wget --header="Host: storage.googleapis.com" --header="User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT
```

```
--2022-10-18 12:37:38--  https://storage.googleapis.com/kaggle-data-sets/181273/407317/bundle/archive.zip?X-Goog-Algorithm=GOOG
Resolving storage.googleapis.com (storage.googleapis.com)... 74.125.203.128, 74.125.204.128, 64.233.188.128, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|74.125.203.128|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 748584920 (714M) [application/zip]
Saving to: ‘archive.zip’

archive.zip         100%[===================>] 713.91M   215MB/s    in 3.4s

2022-10-18 12:37:41 (212 MB/s) - ‘archive.zip’ saved [748584920/748584920]
```

```
!unzip "/content/archive.zip"
```

Streaming output truncated to the last 5000 lines.

```
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7294_19890104/TCGA_DU_7294_19890104_9_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_1.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_10.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_10_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_11.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_11_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_12.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_12_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_13.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_13_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_14.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_14_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_15.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_15_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_16.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_16_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_17.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_17_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_18.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_18_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_19.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_19_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_1_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_2.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_20.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_20_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_21.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_21_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_22.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_22_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_23.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_23_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_24.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_24_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_25.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_25_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_26.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_26_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_27.tif
```

```
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_27_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_28.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_28_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_29.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_29_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_2_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_3.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_30.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_30_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_31.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_31_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_32.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_32_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_3_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_4.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_4_mask.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_5.tif
  inflating: lgg-mri-segmentation/kaggle_3m/TCGA_DU_7298_19910324/TCGA_DU_7298_19910324_5_mask.tif
```

2. data loading

```python
# main_dir = "/content/kaggle_3m/"
# for i in os.listdir(main_dir)[:-1]:
#     sub_dir = main_dir+i+"/"
#     if path.isdir(i+"/"+sub_dir):
#         for j in os.listdir(sub_dir):
#             print("\'{}\'".format(sub_dir+j))
```

```python
def return_file_name(root_dir):
    img_url = []
    mask_img_url = []
    df = pd.DataFrame([])
    for i in os.listdir(root_dir):
        sub_dir = root_dir+i+"/"
```

```
        # pdb.set_trace()
        if path.isdir(sub_dir):
            for j in os.listdir(sub_dir):
    #             img_dir.append(str(sub_dir+j))

                if "mask" in j :
                    mask_img_url.append(str(sub_dir+j))

    # pdb.set_trace()
    img_url = [re.sub("_mask","", i) for i in mask_img_url]

    df["image"] = img_url
    df["mask"] = mask_img_url

    return  df

main_dir = "/content/kaggle_3m/"
data = return_file_name(main_dir)


"""
    checking all file if path is correct
"""
b = True
for i in range(data.shape[0]):
    if not path.isfile(data["image"][i]):
        b = False
    elif not path.isfile(data["mask"][i]):
        b = False

print(b)
```

```
True
```

```
data.head()
```

	image	mask
0	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...
1	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...
2	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...
3	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...
4	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...	/content/kaggle_3m/TCGA_HT_7855_19951020/TCGA_...

```
data.shape
```

```
(3929, 2)
```

4. Data Preprocessing

```
In data this stage: we will analyse, filter and transforming data so that
algorithm can understand and work with the preprocessed data.

So, as the popular saying goes, "if garbage goes in, garbage comes out".
model will only work successfully if data going into machine is high quality.
```

```
when we get data from real world scenarios, these is high chances that it contains noice
data and missing value.
```

```
in this case, we are using the LGG Segmentation Dataset.
```

4.1 Data cleaning and filtering

while doing EDA we observe there are some images which doesn't contain any information(black image) so we will remove those images

```
data.shape
```

```
(3929, 2)
```

```
def garbage_img_prepross(df):
    """
        finding image which doesn't have much infomation
        here we choose 30 as pixel value threshhold  all img which maximum pixel value
        is less that 30 considered to be garbage image
    """
    thres = 30
    temp_img = []

    for i in df["image"]:
        val = np.max(cv2.imread(i))
        if val <thres:
            temp_img.append(i)

    temp_img = np.array(temp_img)
    df = df[~df["image"].isin(temp_img)]

    return df, temp_img
```

```
data, temp_img = garbage_img_prepross(data)
```

```
print("we got total {} image which doesn't contains information, so in order to prepross, we removed it from data".format(len(temp_im
```

```
we got total 89 image which doesn't contains information, so in order to prepross, we removed it from data
```

```
data.shape
```

```
(3840, 2)
```

```
# # visulization garbase images which having less information (max picxel val < 30)

# r = 3
# c = 4
# fig,axis = plt.subplots(r, c, figsize=(16,10))
# p = 0
# for j in range(r):
#     for  i in range(c):
#         axis[j,i].imshow(cv2.imread(temp_img[63-p]))
#         p +=1
```

these image in our train data doesn't have any information so as next move we will remove it from our train data so that our model learn better

NOW, going ahead, we resizes of all images(will done in data pipeline), remove garbage images, now our data is ready for next task - (data pipeline, augmentation , modeling)

```
# data.to_pickle("preprocessed_data.pkl")
# data = pd.read_pickle("/content/preprocessed_data.pkl")
```

Data Pipeline

```
from sklearn.model_selection import train_test_split

X_train, X_test = train_test_split(data, test_size=0.20, random_state=42)
X_train = X_train.reset_index(drop=True)
X_test = X_test.reset_index(drop=True)


# def preprocess_test_img(test):
#     te_img = []
#     for i in range(test.shape[0]):
#         image = cv2.imread(test[i], cv2.IMREAD_UNCHANGED)
#         image = cv2.resize(image, (256,256))
#         image = preprocess_input(image)
#         pdb.set_trace()
#         te_img.append(image)

#     return np.array(te_img)

# X_test1 = preprocess_test_img(X_test["image"])
```

image generator

```
# 1. creating image generator - through url
from tensorflow.keras.preprocessing.image import ImageDataGenerator


datagen=ImageDataGenerator(rescale=1./255.,
                           rotation_range=0.2, zoom_range=0.1, horizontal_flip=True,
                                width_shift_range=0.05,
                                height_shift_range=0.05,
                                shear_range=0.05, fill_mode='nearest')
```

```
mask_datagen=ImageDataGenerator(rescale=1./255.,
                           rotation_range=0.2, zoom_range=0.1, horizontal_flip=True,
                                width_shift_range=0.05,
                                height_shift_range=0.05,
                                shear_range=0.05, fill_mode='nearest')

val_datagen=ImageDataGenerator(rescale=1./255. )

val_mask_datagen=ImageDataGenerator(rescale=1./255. )
```

```
BATCH_SIZE = 16
train_generator=datagen.flow_from_dataframe(dataframe=X_train, x_col='image',
                                            color_mode = 'rgb', class_mode=None,
                                            target_size=(256,256),batch_size=BATCH_SIZE,
                                            seed=42, shuffle=True)

train_mask_generator = mask_datagen.flow_from_dataframe(dataframe=X_train,
                                                        x_col='mask',
                                                        batch_size=BATCH_SIZE,
                                                        class_mode=None,
                                                        target_size=(256, 256),
                                                        seed=42,
                                                        color_mode='grayscale')

val_image_generator = val_datagen.flow_from_dataframe(dataframe=X_test, x_col='image',
                                                batch_size=BATCH_SIZE, seed=42,
                                                shuffle=True,  color_mode='rgb',
                                                class_mode=None,target_size=(256,256))

val_mask_generator = val_mask_datagen.flow_from_dataframe(dataframe=X_test, x_col='mask',
                                                batch_size=BATCH_SIZE, seed=42,
                                                shuffle=True,  color_mode='grayscale',
```

```
class_mode=None,target_size=(256,256))
```

```
def data_iterator(image_generator, mask_generator):
    while True:
        X, Y = next(image_generator), next(mask_generator)
        yield X, Y

def data_generator(train_image_generator, train_mask_generator, val_image_generator, val_mask_generator):
    return data_iterator(train_image_generator, train_mask_generator), data_iterator(val_image_generator, val_mask_generator)

train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
```

```
Found 3072 validated image filenames.
Found 3072 validated image filenames.
Found 768 validated image filenames.
Found 768 validated image filenames.
```

```
train_generator.n//BATCH_SIZE, val_image_generator.n//BATCH_SIZE
```

```
(192, 48)
```

```
# !rm -rf ./logs/
```

callbacks

```
%load_ext tensorboard
```

```
# define callbacks for learning rate scheduling and best checkpoints saving
import datetime
from tensorflow.keras.callbacks import ModelCheckpoint
```

```
from tensorflow.keras.callbacks import ModelCheckpoint

def create_callback_lists(name = ""):
    filepath='best_model_with_{}.h5'.format(name)
    checkpoint = ModelCheckpoint(filepath=filepath, monitor='val_iou_score', verbose=1, save_best_only=True, mode='max')

    learning_rt = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_iou_score', min_lr=0.0001,patience=1, verbose=1)
    # !rm -rf ./logs/
    log_dir="logs/fit/" + datetime.datetime.now().strftime("%Y-%m-%d-%H_%M")

    early_stop_callback = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=3)

    tensorboard_callback = tf.keras.callbacks.TensorBoard(log_dir=log_dir, histogram_freq=1, write_graph=True)

    return [early_stop_callback, tensorboard_callback, checkpoint]

callback_list = create_callback_lists()
```

drive connect

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

5. Modeling

Base line model

```
# loading unet model with backbone - resnet34

model = sm.Unet('resnet34', encoder_weights="imagenet", classes=1,
```

```
                  activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3))



optim = tf.keras.optimizers.Adam(learning_rate = .006)
callback_list = create_callback_lists(name = "resnet34_UNET")
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model.compile(optim, loss, metrics=[iou_score])


# with different augmentation techinique
# train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
callback_list = create_callback_lists("baseline_unet")

history = model.fit(train_data_loader, steps_per_epoch=192, epochs=10,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/10
192/192 [==============================] - 103s 444ms/step - loss: 0.4907 - iou_score: 0.3781 - val_loss: 0.9984 - val_iou_scor
Epoch 2/10
192/192 [==============================] - 80s 420ms/step - loss: 0.2970 - iou_score: 0.5588 - val_loss: 1.0000 - val_iou_score
Epoch 3/10
192/192 [==============================] - 83s 432ms/step - loss: 0.2626 - iou_score: 0.5992 - val_loss: 1.0000 - val_iou_score
Epoch 4/10
192/192 [==============================] - 84s 438ms/step - loss: 0.2463 - iou_score: 0.6150 - val_loss: 1.0000 - val_iou_score
Epoch 5/10
192/192 [==============================] - 81s 420ms/step - loss: 0.2167 - iou_score: 0.6539 - val_loss: 0.9994 - val_iou_score
Epoch 6/10
192/192 [==============================] - 82s 426ms/step - loss: 0.2149 - iou_score: 0.6599 - val_loss: 0.2782 - val_iou_score
Epoch 7/10
192/192 [==============================] - 81s 424ms/step - loss: 0.1930 - iou_score: 0.6861 - val_loss: 0.4557 - val_iou_score
Epoch 8/10
192/192 [==============================] - 81s 421ms/step - loss: 0.1709 - iou_score: 0.7147 - val_loss: 0.2033 - val_iou_score
Epoch 9/10
192/192 [==============================] - 82s 430ms/step - loss: 0.1623 - iou_score: 0.7297 - val_loss: 0.2393 - val_iou_score
Epoch 10/10
192/192 [==============================] - 81s 425ms/step - loss: 0.1681 - iou_score: 0.7257 - val_loss: 0.2126 - val_iou_score
```

```
history = model.fit(train_data_loader, steps_per_epoch=192, epochs=5,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/5
192/192 [==============================] - 92s 481ms/step - loss: 0.1871 - iou_score: 0.6998 - val_loss: 0.2552 - val_iou_score
Epoch 2/5
192/192 [==============================] - 86s 450ms/step - loss: 0.1626 - iou_score: 0.7270 - val_loss: 0.1853 - val_iou_score
Epoch 3/5
192/192 [==============================] - 83s 434ms/step - loss: 0.1570 - iou_score: 0.7361 - val_loss: 0.1571 - val_iou_score
Epoch 4/5
192/192 [==============================] - 87s 455ms/step - loss: 0.1494 - iou_score: 0.7438 - val_loss: 0.2192 - val_iou_score
Epoch 5/5
192/192 [==============================] - 84s 440ms/step - loss: 0.1408 - iou_score: 0.7583 - val_loss: 0.1548 - val_iou_score
```

```
history = model.fit(train_data_loader, steps_per_epoch=192, epochs=5,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/5
192/192 [==============================] - 88s 460ms/step - loss: 0.1679 - iou_score: 0.7263 - val_loss: 0.2436 - val_iou_score
Epoch 2/5
192/192 [==============================] - 83s 434ms/step - loss: 0.1449 - iou_score: 0.7512 - val_loss: 0.1548 - val_iou_score
Epoch 3/5
192/192 [==============================] - 84s 436ms/step - loss: 0.1529 - iou_score: 0.7430 - val_loss: 0.1571 - val_iou_score
Epoch 4/5
192/192 [==============================] - 87s 455ms/step - loss: 0.1450 - iou_score: 0.7508 - val_loss: 0.2499 - val_iou_score
Epoch 5/5
192/192 [==============================] - 90s 472ms/step - loss: 0.1449 - iou_score: 0.7531 - val_loss: 0.2054 - val_iou_score
```

```
history = model.fit(train_data_loader, steps_per_epoch=192, epochs=15,
                    validation_data = val_data_loader,  validation_steps = 48,
```

```
                    initial_epoch = 10, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 11/15
192/192 [==============================] - ETA: 0s - loss: 0.1868 - iou_score: 0.6988
Epoch 11: val_iou_score did not improve from 0.71187
192/192 [==============================] - 116s 602ms/step - loss: 0.1868 - iou_score: 0.6988 - val_loss: 0.1805 - val_iou_scor
Epoch 12/15
192/192 [==============================] - ETA: 0s - loss: 0.1581 - iou_score: 0.7330
Epoch 12: val_iou_score improved from 0.71187 to 0.74025, saving model to best_model_with_baseline_unet.hdf5
192/192 [==============================] - 95s 498ms/step - loss: 0.1581 - iou_score: 0.7330 - val_loss: 0.1550 - val_iou_score
Epoch 13/15
192/192 [==============================] - ETA: 0s - loss: 0.1475 - iou_score: 0.7491
Epoch 13: val_iou_score improved from 0.74025 to 0.74189, saving model to best_model_with_baseline_unet.hdf5
192/192 [==============================] - 96s 500ms/step - loss: 0.1475 - iou_score: 0.7491 - val_loss: 0.1554 - val_iou_score
Epoch 14/15
192/192 [==============================] - ETA: 0s - loss: 0.1450 - iou_score: 0.7502
Epoch 14: val_iou_score improved from 0.74189 to 0.74702, saving model to best_model_with_baseline_unet.hdf5
192/192 [==============================] - 96s 500ms/step - loss: 0.1450 - iou_score: 0.7502 - val_loss: 0.1514 - val_iou_score
Epoch 15/15
192/192 [==============================] - ETA: 0s - loss: 0.1347 - iou_score: 0.7662
Epoch 15: val_iou_score improved from 0.74702 to 0.75386, saving model to best_model_with_baseline_unet.hdf5
192/192 [==============================] - 96s 499ms/step - loss: 0.1347 - iou_score: 0.7662 - val_loss: 0.1460 - val_iou_score
```

saving and loading model in drive

```
# from google.colab import drive
# drive.mount('/content/drive')
```

```
# model.save_weights("/content/drive/MyDrive/imp document/unet_res_model.hdf5")
# model.load_weights("/content/drive/MyDrive/imp document/unet_res_model.hdf5")
```

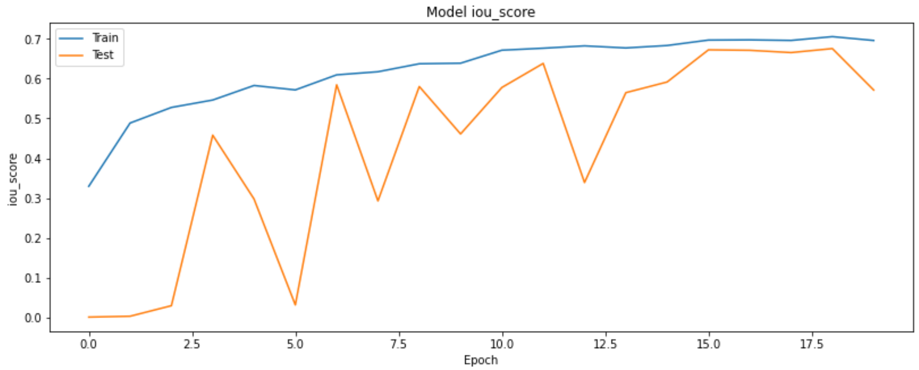
```
# with different augmentation techinique - 20 epochs
train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
callback_list = create_callback_lists("baseline_unet")

history = model.fit(train_data_loader, steps_per_epoch=384, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 96,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

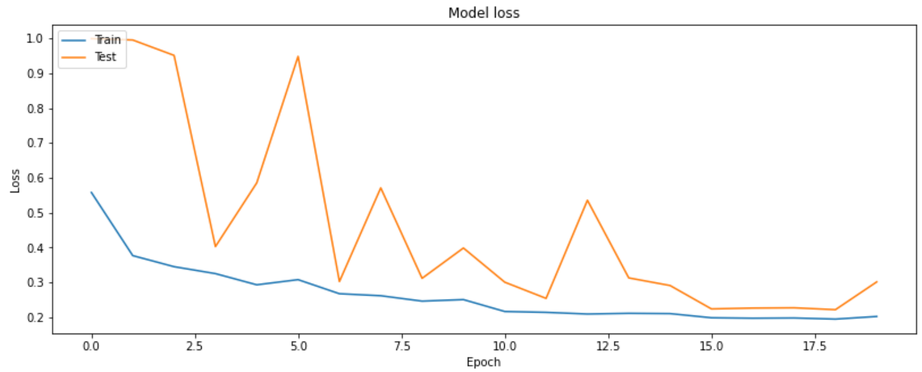
```
Epoch 1/20
384/384 [==============================] - ETA: 0s - loss: 0.5580 - iou_score: 0.3295
Epoch 1: val_iou_score improved from -inf to 0.00048, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 113s 247ms/step - loss: 0.5580 - iou_score: 0.3295 - val_loss: 0.9990 - val_iou_sc
Epoch 2/20
384/384 [==============================] - ETA: 0s - loss: 0.3771 - iou_score: 0.4886
Epoch 2: val_iou_score improved from 0.00048 to 0.00269, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 103s 268ms/step - loss: 0.3771 - iou_score: 0.4886 - val_loss: 0.9951 - val_iou_sc
Epoch 3/20
384/384 [==============================] - ETA: 0s - loss: 0.3454 - iou_score: 0.5278
Epoch 3: val_iou_score improved from 0.00269 to 0.02921, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 93s 243ms/step - loss: 0.3454 - iou_score: 0.5278 - val_loss: 0.9510 - val_iou_sco
Epoch 4/20
384/384 [==============================] - ETA: 0s - loss: 0.3255 - iou_score: 0.5465
Epoch 4: val_iou_score improved from 0.02921 to 0.45811, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 92s 240ms/step - loss: 0.3255 - iou_score: 0.5465 - val_loss: 0.4029 - val_iou_sco
Epoch 5/20
384/384 [==============================] - ETA: 0s - loss: 0.2934 - iou_score: 0.5830
Epoch 5: val_iou_score did not improve from 0.45811
384/384 [==============================] - 93s 243ms/step - loss: 0.2934 - iou_score: 0.5830 - val_loss: 0.5856 - val_iou_sco
Epoch 6/20
384/384 [==============================] - ETA: 0s - loss: 0.3082 - iou_score: 0.5719
Epoch 6: val_iou_score did not improve from 0.45811
384/384 [==============================] - 92s 239ms/step - loss: 0.3082 - iou_score: 0.5719 - val_loss: 0.9481 - val_iou_sco
Epoch 7/20
384/384 [==============================] - ETA: 0s - loss: 0.2678 - iou_score: 0.6098
Epoch 7: val_iou_score improved from 0.45811 to 0.58481, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 103s 267ms/step - loss: 0.2678 - iou_score: 0.6098 - val_loss: 0.3028 - val_iou_sc
Epoch 8/20
384/384 [==============================] - ETA: 0s - loss: 0.2621 - iou_score: 0.6176
Epoch 8: val_iou_score did not improve from 0.58481
```

```
384/384 [==============================] - 93s 241ms/step - loss: 0.2621 - iou_score: 0.6176 - val_loss: 0.5712 - val_iou_sco
Epoch 9/20
384/384 [==============================] - ETA: 0s - loss: 0.2466 - iou_score: 0.6377
Epoch 9: val_iou_score did not improve from 0.58481
384/384 [==============================] - 91s 237ms/step - loss: 0.2466 - iou_score: 0.6377 - val_loss: 0.3120 - val_iou_sco
Epoch 10/20
384/384 [==============================] - ETA: 0s - loss: 0.2510 - iou_score: 0.6391
Epoch 10: val_iou_score did not improve from 0.58481
384/384 [==============================] - 93s 241ms/step - loss: 0.2510 - iou_score: 0.6391 - val_loss: 0.3989 - val_iou_sco
Epoch 11/20
384/384 [==============================] - ETA: 0s - loss: 0.2169 - iou_score: 0.6718
Epoch 11: val_iou_score did not improve from 0.58481
384/384 [==============================] - 90s 234ms/step - loss: 0.2169 - iou_score: 0.6718 - val_loss: 0.3009 - val_iou_sco
Epoch 12/20
384/384 [==============================] - ETA: 0s - loss: 0.2144 - iou_score: 0.6768
Epoch 12: val_iou_score improved from 0.58481 to 0.63858, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 94s 246ms/step - loss: 0.2144 - iou_score: 0.6768 - val_loss: 0.2545 - val_iou_sco
Epoch 13/20
384/384 [==============================] - ETA: 0s - loss: 0.2097 - iou_score: 0.6827
Epoch 13: val_iou_score did not improve from 0.63858
384/384 [==============================] - 91s 238ms/step - loss: 0.2097 - iou_score: 0.6827 - val_loss: 0.5358 - val_iou_sco
Epoch 14/20
384/384 [==============================] - ETA: 0s - loss: 0.2117 - iou_score: 0.6775
Epoch 14: val_iou_score did not improve from 0.63858
384/384 [==============================] - 114s 297ms/step - loss: 0.2117 - iou_score: 0.6775 - val_loss: 0.3132 - val_iou_sc
Epoch 15/20
```

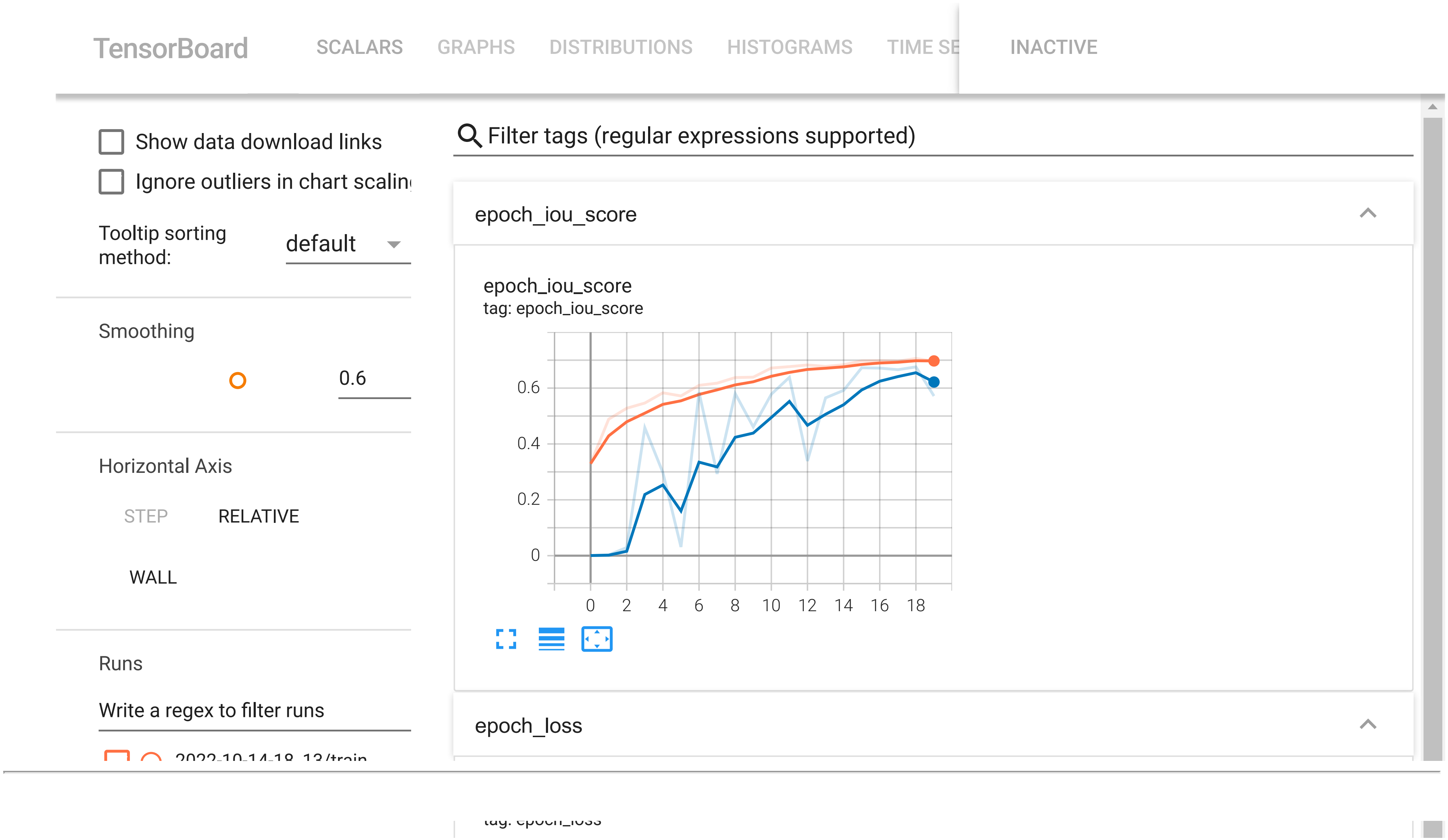
```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
```

```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```






```
# loading tensorboard
%tensorboard --logdir logs/fit
```




```
# with different augmentation techinique
train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
callback_list = create_callback_lists("baseline_unet")

history = model.fit(train_data_loader, steps_per_epoch=384, epochs=40,
```

```
              validation_data = val_data_loader,  validation_steps = 96,
              initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/40
384/384 [==============================] - ETA: 0s - loss: 0.5389 - iou_score: 0.3446
Epoch 1: val_iou_score improved from -inf to 0.00004, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 114s 262ms/step - loss: 0.5389 - iou_score: 0.3446 - val_loss: 0.9999 - val_iou_sc
Epoch 2/40
384/384 [==============================] - ETA: 0s - loss: 0.3693 - iou_score: 0.4970
Epoch 2: val_iou_score did not improve from 0.00004
384/384 [==============================] - 101s 262ms/step - loss: 0.3693 - iou_score: 0.4970 - val_loss: 1.0000 - val_iou_sc
Epoch 3/40
384/384 [==============================] - ETA: 0s - loss: 0.3413 - iou_score: 0.5344
Epoch 3: val_iou_score improved from 0.00004 to 0.32058, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 107s 279ms/step - loss: 0.3413 - iou_score: 0.5344 - val_loss: 0.5604 - val_iou_sc
Epoch 4/40
384/384 [==============================] - ETA: 0s - loss: 0.3134 - iou_score: 0.5590
Epoch 4: val_iou_score improved from 0.32058 to 0.54698, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 97s 252ms/step - loss: 0.3134 - iou_score: 0.5590 - val_loss: 0.3372 - val_iou_sco
Epoch 5/40
384/384 [==============================] - ETA: 0s - loss: 0.2858 - iou_score: 0.5907
Epoch 5: val_iou_score did not improve from 0.54698
384/384 [==============================] - 96s 249ms/step - loss: 0.2858 - iou_score: 0.5907 - val_loss: 0.5975 - val_iou_sco
Epoch 6/40
384/384 [==============================] - ETA: 0s - loss: 0.3001 - iou_score: 0.5775
Epoch 6: val_iou_score improved from 0.54698 to 0.61231, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 109s 284ms/step - loss: 0.3001 - iou_score: 0.5775 - val_loss: 0.2757 - val_iou_sc
Epoch 7/40
384/384 [==============================] - ETA: 0s - loss: 0.2653 - iou_score: 0.6125
Epoch 7: val_iou_score improved from 0.61231 to 0.61754, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 128s 334ms/step - loss: 0.2653 - iou_score: 0.6125 - val_loss: 0.2679 - val_iou_sc
Epoch 8/40
384/384 [==============================] - ETA: 0s - loss: 0.2485 - iou_score: 0.6325
Epoch 8: val_iou_score did not improve from 0.61754
384/384 [==============================] - 114s 296ms/step - loss: 0.2485 - iou_score: 0.6325 - val_loss: 0.3949 - val_iou_sc
Epoch 9/40
384/384 [==============================] - ETA: 0s - loss: 0.2357 - iou_score: 0.6516
Epoch 9: val_iou_score did not improve from 0.61754
384/384 [==============================] - 103s 269ms/step - loss: 0.2357 - iou_score: 0.6516 - val_loss: 0.3018 - val_iou_sc
```

```
Epoch 10/40
384/384 [==============================] - ETA: 0s - loss: 0.2447 - iou_score: 0.6447
Epoch 10: val_iou_score did not improve from 0.61754
384/384 [==============================] - 106s 277ms/step - loss: 0.2447 - iou_score: 0.6447 - val_loss: 0.3395 - val_iou_sc
Epoch 11/40
384/384 [==============================] - ETA: 0s - loss: 0.2135 - iou_score: 0.6742
Epoch 11: val_iou_score improved from 0.61754 to 0.64517, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 96s 250ms/step - loss: 0.2135 - iou_score: 0.6742 - val_loss: 0.2496 - val_iou_sco
Epoch 12/40
384/384 [==============================] - ETA: 0s - loss: 0.2184 - iou_score: 0.6680
Epoch 12: val_iou_score did not improve from 0.64517
384/384 [==============================] - 129s 335ms/step - loss: 0.2184 - iou_score: 0.6680 - val_loss: 0.6545 - val_iou_sc
Epoch 13/40
384/384 [==============================] - ETA: 0s - loss: 0.2222 - iou_score: 0.6686
Epoch 13: val_iou_score did not improve from 0.64517
384/384 [==============================] - 100s 260ms/step - loss: 0.2222 - iou_score: 0.6686 - val_loss: 0.2534 - val_iou_sc
Epoch 14/40
384/384 [==============================] - ETA: 0s - loss: 0.2089 - iou_score: 0.6815
Epoch 14: val_iou_score improved from 0.64517 to 0.66384, saving model to best_model_with_baseline_unet.hdf5
384/384 [==============================] - 101s 262ms/step - loss: 0.2089 - iou_score: 0.6815 - val_loss: 0.2334 - val_iou_sc
```

```
history = model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=20, validation_data=test_dataloader, callbacks =
```

```
Epoch 1/20
384/384 [==============================] - 72s 164ms/step - loss: 0.6686 - iou_score: 0.2306 - val_loss: 0.5283 - val_iou_score
Epoch 2/20
384/384 [==============================] - 62s 161ms/step - loss: 0.5944 - iou_score: 0.3008 - val_loss: 0.9969 - val_iou_score
Epoch 3/20
384/384 [==============================] - 62s 161ms/step - loss: 0.5466 - iou_score: 0.3337 - val_loss: 0.4612 - val_iou_score
Epoch 4/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4589 - iou_score: 0.4068 - val_loss: 0.4442 - val_iou_score
Epoch 5/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4380 - iou_score: 0.4241 - val_loss: 0.3299 - val_iou_score
Epoch 6/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4111 - iou_score: 0.4527 - val_loss: 0.4616 - val_iou_score
Epoch 7/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4036 - iou_score: 0.4609 - val_loss: 0.4492 - val_iou_score
Epoch 8/20
```

```
384/384 [==============================] - 62s 161ms/step - loss: 0.3950 - iou_score: 0.4658 - val_loss: 0.3112 - val_iou_score
Epoch 9/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3986 - iou_score: 0.4659 - val_loss: 0.3197 - val_iou_score
Epoch 10/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3652 - iou_score: 0.5026 - val_loss: 0.2799 - val_iou_score
Epoch 11/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3885 - iou_score: 0.4766 - val_loss: 0.2948 - val_iou_score
Epoch 12/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3594 - iou_score: 0.5059 - val_loss: 0.3294 - val_iou_score
Epoch 13/20
384/384 [==============================] - 62s 160ms/step - loss: 0.3759 - iou_score: 0.4887 - val_loss: 0.2754 - val_iou_score
Epoch 14/20
384/384 [==============================] - 62s 160ms/step - loss: 0.3616 - iou_score: 0.5055 - val_loss: 0.2620 - val_iou_score
Epoch 15/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3580 - iou_score: 0.5083 - val_loss: 0.3321 - val_iou_score
Epoch 16/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3570 - iou_score: 0.5115 - val_loss: 0.2949 - val_iou_score
Epoch 17/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3652 - iou_score: 0.5031 - val_loss: 0.2749 - val_iou_score
Epoch 18/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3560 - iou_score: 0.5099 - val_loss: 0.3168 - val_iou_score
Epoch 19/20
384/384 [==============================] - 62s 160ms/step - loss: 0.3479 - iou_score: 0.5201 - val_loss: 0.2595 - val_iou_score
Epoch 20/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3419 - iou_score: 0.5267 - val_loss: 0.2623 - val_iou_score
```

```
history = model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=20, validation_data=test_dataloader, callbacks =
```

```
Epoch 1/20
384/384 [==============================] - 61s 157ms/step - loss: 0.3358 - iou_score: 0.5344 - val_loss: 0.2844 - val_iou_score
Epoch 2/20
384/384 [==============================] - 61s 159ms/step - loss: 0.3532 - iou_score: 0.5162 - val_loss: 0.2511 - val_iou_score
Epoch 3/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3467 - iou_score: 0.5219 - val_loss: 0.2470 - val_iou_score
Epoch 4/20
384/384 [==============================] - 61s 159ms/step - loss: 0.3328 - iou_score: 0.5373 - val_loss: 0.3833 - val_iou_score
Epoch 5/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3389 - iou_score: 0.5297 - val_loss: 0.2699 - val_iou_score
```

```
Epoch 6/20
384/384 [==============================] - 61s 159ms/step - loss: 0.3217 - iou_score: 0.5497 - val_loss: 0.2346 - val_iou_score
Epoch 7/20
384/384 [==============================] - 61s 160ms/step - loss: 0.3272 - iou_score: 0.5449 - val_loss: 0.2353 - val_iou_score
Epoch 8/20
384/384 [==============================] - 62s 160ms/step - loss: 0.3421 - iou_score: 0.5269 - val_loss: 0.2441 - val_iou_score
Epoch 9/20
384/384 [==============================] - 61s 160ms/step - loss: 0.3551 - iou_score: 0.5147 - val_loss: 0.2281 - val_iou_score
```

```
model.save_weights("model_case_2.h5")
```

```
history = model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=20, validation_data=test_dataloader, callbacks =
```

```
Epoch 1/20
384/384 [==============================] - 68s 166ms/step - loss: 0.6254 - iou_score: 0.2597 - val_loss: 0.5792 - val_iou_score
Epoch 2/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4931 - iou_score: 0.3729 - val_loss: 0.4059 - val_iou_score
Epoch 3/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4611 - iou_score: 0.4028 - val_loss: 0.3932 - val_iou_score
Epoch 4/20
384/384 [==============================] - 62s 161ms/step - loss: 0.4453 - iou_score: 0.4150 - val_loss: 0.3533 - val_iou_score
Epoch 5/20
384/384 [==============================] - 62s 161ms/step - loss: 0.4257 - iou_score: 0.4365 - val_loss: 0.3290 - val_iou_score
Epoch 6/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4012 - iou_score: 0.4616 - val_loss: 0.3800 - val_iou_score
Epoch 7/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3994 - iou_score: 0.4644 - val_loss: 0.3500 - val_iou_score
Epoch 8/20
384/384 [==============================] - 62s 162ms/step - loss: 0.4143 - iou_score: 0.4485 - val_loss: 0.2865 - val_iou_score
Epoch 9/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3906 - iou_score: 0.4732 - val_loss: 0.2744 - val_iou_score
Epoch 10/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3835 - iou_score: 0.4797 - val_loss: 0.3165 - val_iou_score
Epoch 11/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3805 - iou_score: 0.4848 - val_loss: 0.2905 - val_iou_score
Epoch 12/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3711 - iou_score: 0.4924 - val_loss: 0.2946 - val_iou_score
```

```
Epoch 13/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3751 - iou_score: 0.4925 - val_loss: 0.2513 - val_iou_score
Epoch 14/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3744 - iou_score: 0.4896 - val_loss: 0.2777 - val_iou_score
Epoch 15/20
384/384 [==============================] - 62s 161ms/step - loss: 0.3649 - iou_score: 0.4993 - val_loss: 0.2778 - val_iou_score
Epoch 16/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3540 - iou_score: 0.5132 - val_loss: 0.3154 - val_iou_score
Epoch 17/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3396 - iou_score: 0.5300 - val_loss: 0.2876 - val_iou_score
Epoch 18/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3656 - iou_score: 0.5021 - val_loss: 0.3081 - val_iou_score
Epoch 19/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3687 - iou_score: 0.4998 - val_loss: 0.2631 - val_iou_score
Epoch 20/20
384/384 [==============================] - 62s 162ms/step - loss: 0.3346 - iou_score: 0.5354 - val_loss: 0.2549 - val_iou_score
```

hyperparameter tuning

for hyperparameter tuning we have choosen 100 image only so that we can better analysis which loss, matrix, optimiser is better.

```
from tensorflow.keras.models import clone_model
```

```
tf.random.set_seed(22)
tf.keras.backend.clear_session()


# model = clone_model(model_t)
model = sm.Unet('resnet34', encoder_weights="imagenet", classes=1, activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3)


optim = tf.keras.optimizers.Adam()
focal_loss =  DiceLoss()#sm.losses.bce_jaccard_loss
```

```
model.compile(optim, focal_loss, metrics=[iou_score])
```

```
# diceloss + ADAM
```

```
model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=50, validation_data=test_dataloader)

    Epoch 1/50
    12/12 [==============================] - 7s 259ms/step - loss: 0.9609 - iou_score: 0.0201 - val_loss: 0.9763 - val_iou_score:
    Epoch 2/50
    12/12 [==============================] - 2s 185ms/step - loss: 0.9337 - iou_score: 0.0346 - val_loss: 0.9780 - val_iou_score:
    Epoch 3/50
    12/12 [==============================] - 2s 186ms/step - loss: 0.8931 - iou_score: 0.0571 - val_loss: 0.9285 - val_iou_score:
    Epoch 4/50
    12/12 [==============================] - 2s 189ms/step - loss: 0.8589 - iou_score: 0.0781 - val_loss: 0.9820 - val_iou_score:
    Epoch 5/50
    12/12 [==============================] - 2s 188ms/step - loss: 0.7762 - iou_score: 0.1314 - val_loss: 0.9532 - val_iou_score:
    Epoch 6/50
    12/12 [==============================] - 2s 189ms/step - loss: 0.7429 - iou_score: 0.1539 - val_loss: 0.8610 - val_iou_score:
    Epoch 7/50
    12/12 [==============================] - 2s 188ms/step - loss: 0.6253 - iou_score: 0.2421 - val_loss: 0.9688 - val_iou_score:
    Epoch 8/50
    12/12 [==============================] - 2s 189ms/step - loss: 0.6034 - iou_score: 0.2587 - val_loss: 0.9545 - val_iou_score:
    Epoch 9/50
    12/12 [==============================] - 2s 187ms/step - loss: 0.5575 - iou_score: 0.3065 - val_loss: 0.9652 - val_iou_score:
    Epoch 10/50
    12/12 [==============================] - 2s 190ms/step - loss: 0.5545 - iou_score: 0.3090 - val_loss: 0.9327 - val_iou_score:
    Epoch 11/50
    12/12 [==============================] - 2s 188ms/step - loss: 0.5572 - iou_score: 0.3083 - val_loss: 0.8818 - val_iou_score:
    Epoch 12/50
    12/12 [==============================] - 2s 189ms/step - loss: 0.5007 - iou_score: 0.3541 - val_loss: 0.7997 - val_iou_score:
    Epoch 13/50
    12/12 [==============================] - 2s 192ms/step - loss: 0.4982 - iou_score: 0.3555 - val_loss: 0.7148 - val_iou_score:
    Epoch 14/50
    12/12 [==============================] - 2s 191ms/step - loss: 0.4139 - iou_score: 0.4276 - val_loss: 0.8090 - val_iou_score:
    Epoch 15/50
    12/12 [==============================] - 2s 191ms/step - loss: 0.4122 - iou_score: 0.4289 - val_loss: 0.7130 - val_iou_score:
    Epoch 16/50
    12/12 [==============================] - 2s 190ms/step - loss: 0.3978 - iou_score: 0.4449 - val_loss: 0.7182 - val_iou_score:
    Epoch 17/50
```

```
12/12 [==============================] - 2s 192ms/step - loss: 0.3841 - iou_score: 0.4600 - val_loss: 0.8105 - val_iou_score:
Epoch 18/50
12/12 [==============================] - 2s 193ms/step - loss: 0.4821 - iou_score: 0.3760 - val_loss: 0.7225 - val_iou_score:
Epoch 19/50
12/12 [==============================] - 2s 192ms/step - loss: 0.4598 - iou_score: 0.3876 - val_loss: 0.6739 - val_iou_score:
Epoch 20/50
12/12 [==============================] - 2s 201ms/step - loss: 0.4226 - iou_score: 0.4226 - val_loss: 0.5576 - val_iou_score:
Epoch 21/50
12/12 [==============================] - 2s 192ms/step - loss: 0.4007 - iou_score: 0.4474 - val_loss: 0.8538 - val_iou_score:
Epoch 22/50
12/12 [==============================] - 2s 194ms/step - loss: 0.4095 - iou_score: 0.4333 - val_loss: 0.7961 - val_iou_score:
Epoch 23/50
12/12 [==============================] - 2s 193ms/step - loss: 0.5225 - iou_score: 0.3381 - val_loss: 0.8650 - val_iou_score:
Epoch 24/50
12/12 [==============================] - 2s 191ms/step - loss: 0.5418 - iou_score: 0.3182 - val_loss: 0.6795 - val_iou_score:
Epoch 25/50
12/12 [==============================] - 2s 191ms/step - loss: 0.4789 - iou_score: 0.3690 - val_loss: 0.5979 - val_iou_score:
Epoch 26/50
12/12 [==============================] - 2s 192ms/step - loss: 0.4579 - iou_score: 0.3909 - val_loss: 0.5759 - val_iou_score:
Epoch 27/50
12/12 [==============================] - 2s 191ms/step - loss: 0.4155 - iou_score: 0.4264 - val_loss: 0.5782 - val_iou_score:
Epoch 28/50
12/12 [==============================] - 2s 191ms/step - loss: 0.4249 - iou_score: 0.4240 - val_loss: 0.5975 - val_iou_score:
```

got **val_iou score = .38** with 1000 images

```
# diceloss + ADAM

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=50, validation_data=test_dataloader)
```

```
Epoch 1/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3818 - iou_score: 0.4648 - val_loss: 0.6134 - val_iou_score:
Epoch 2/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3600 - iou_score: 0.4950 - val_loss: 0.5507 - val_iou_score:
Epoch 3/50
12/12 [==============================] - 2s 187ms/step - loss: 0.3170 - iou_score: 0.5326 - val_loss: 0.5205 - val_iou_score:
Epoch 4/50
12/12 [==============================] - 2s 187ms/step - loss: 0.3501 - iou_score: 0.4967 - val_loss: 0.6795 - val_iou_score:
```

```
Epoch 5/50
12/12 [==============================] - 2s 189ms/step - loss: 0.3595 - iou_score: 0.4806 - val_loss: 0.5698 - val_iou_score:
Epoch 6/50
12/12 [==============================] - 2s 189ms/step - loss: 0.3178 - iou_score: 0.5384 - val_loss: 0.5131 - val_iou_score:
Epoch 7/50
12/12 [==============================] - 2s 190ms/step - loss: 0.3330 - iou_score: 0.5115 - val_loss: 0.5398 - val_iou_score:
Epoch 8/50
12/12 [==============================] - 2s 190ms/step - loss: 0.3237 - iou_score: 0.5379 - val_loss: 0.5700 - val_iou_score:
Epoch 9/50
12/12 [==============================] - 2s 191ms/step - loss: 0.3059 - iou_score: 0.5513 - val_loss: 0.5719 - val_iou_score:
Epoch 10/50
12/12 [==============================] - 2s 190ms/step - loss: 0.3383 - iou_score: 0.5128 - val_loss: 0.4952 - val_iou_score:
Epoch 11/50
12/12 [==============================] - 2s 192ms/step - loss: 0.2605 - iou_score: 0.5930 - val_loss: 0.4956 - val_iou_score:
Epoch 12/50
12/12 [==============================] - 2s 190ms/step - loss: 0.3164 - iou_score: 0.5490 - val_loss: 0.6552 - val_iou_score:
Epoch 13/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3342 - iou_score: 0.5193 - val_loss: 0.6231 - val_iou_score:
Epoch 14/50
12/12 [==============================] - 2s 194ms/step - loss: 0.3858 - iou_score: 0.4680 - val_loss: 0.4958 - val_iou_score:
Epoch 15/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3888 - iou_score: 0.4704 - val_loss: 0.5011 - val_iou_score:
Epoch 16/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3306 - iou_score: 0.5200 - val_loss: 0.4848 - val_iou_score:
Epoch 17/50
12/12 [==============================] - 2s 193ms/step - loss: 0.2655 - iou_score: 0.5889 - val_loss: 0.5091 - val_iou_score:
Epoch 18/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3150 - iou_score: 0.5311 - val_loss: 0.5222 - val_iou_score:
Epoch 19/50
12/12 [==============================] - 2s 195ms/step - loss: 0.3239 - iou_score: 0.5290 - val_loss: 0.5069 - val_iou_score:
Epoch 20/50
12/12 [==============================] - 2s 192ms/step - loss: 0.3410 - iou_score: 0.5196 - val_loss: 0.5442 - val_iou_score:
Epoch 21/50
12/12 [==============================] - 2s 192ms/step - loss: 0.3229 - iou_score: 0.5323 - val_loss: 0.5483 - val_iou_score:
Epoch 22/50
12/12 [==============================] - 2s 191ms/step - loss: 0.3788 - iou_score: 0.4823 - val_loss: 0.5361 - val_iou_score:
Epoch 23/50
12/12 [==============================] - 2s 190ms/step - loss: 0.2782 - iou_score: 0.5799 - val_loss: 0.5108 - val_iou_score:
Epoch 24/50
12/12 [==============================] - 2s 189ms/step - loss: 0.2888 - iou_score: 0.5671 - val_loss: 0.5137 - val_iou_score:
Epoch 25/50
```

```
12/12 [==============================] - 2s 191ms/step - loss: 0.3185 - iou_score: 0.5446 - val_loss: 0.5071 - val_iou_score:
Epoch 26/50
12/12 [==============================] - 2s 193ms/step - loss: 0.3520 - iou_score: 0.5159 - val_loss: 0.5000 - val_iou_score:
Epoch 27/50
12/12 [==============================] - 2s 190ms/step - loss: 0.2243 - iou_score: 0.6434 - val_loss: 0.5340 - val_iou_score:
Epoch 28/50
12/12 [==============================] - 2s 191ms/step - loss: 0.3160 - iou_score: 0.5299 - val_loss: 0.5166 - val_iou_score:
Epoch 29/50
```

got **val_iou score = .41** with 100 images and epch = 100

performance is not increasing after some extent

```
# diceloss + ADAM

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader)
```

```
Epoch 1/10
12/12 [==============================] - 12s 259ms/step - loss: 0.9636 - iou_score: 0.0187 - val_loss: 0.9998 - val_iou_score:
Epoch 2/10
12/12 [==============================] - 2s 187ms/step - loss: 0.9396 - iou_score: 0.0314 - val_loss: 1.0000 - val_iou_score: 7
Epoch 3/10
12/12 [==============================] - 2s 187ms/step - loss: 0.9116 - iou_score: 0.0469 - val_loss: 0.9753 - val_iou_score: 0
Epoch 4/10
12/12 [==============================] - 2s 188ms/step - loss: 0.8793 - iou_score: 0.0653 - val_loss: 0.9299 - val_iou_score: 0
Epoch 5/10
12/12 [==============================] - 2s 187ms/step - loss: 0.8361 - iou_score: 0.0915 - val_loss: 0.9383 - val_iou_score: 0
Epoch 6/10
12/12 [==============================] - 2s 188ms/step - loss: 0.7702 - iou_score: 0.1388 - val_loss: 0.9255 - val_iou_score: 0
Epoch 7/10
12/12 [==============================] - 2s 188ms/step - loss: 0.6784 - iou_score: 0.2016 - val_loss: 0.9726 - val_iou_score: 0
Epoch 8/10
12/12 [==============================] - 2s 187ms/step - loss: 0.6260 - iou_score: 0.2384 - val_loss: 0.9887 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 187ms/step - loss: 0.5364 - iou_score: 0.3163 - val_loss: 0.9133 - val_iou_score: 0
Epoch 10/10
12/12 [==============================] - 2s 189ms/step - loss: 0.5499 - iou_score: 0.3019 - val_loss: 1.0000 - val_iou_score: 4
<keras.callbacks.History at 0x7f5090271fd0>
```

diceloss + ADAM

```
model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 2s 198ms/step - loss: 0.9109 - iou_score: 0.0473 - val_loss: 0.9543 - val_iou_score: 0
Epoch 2/10
12/12 [==============================] - 2s 193ms/step - loss: 0.9091 - iou_score: 0.0484 - val_loss: 0.9655 - val_iou_score: 0
Epoch 3/10
12/12 [==============================] - 2s 193ms/step - loss: 0.8998 - iou_score: 0.0535 - val_loss: 0.9749 - val_iou_score: 0
Epoch 4/10
12/12 [==============================] - 2s 192ms/step - loss: 0.8983 - iou_score: 0.0544 - val_loss: 0.9520 - val_iou_score: 0
Epoch 5/10
12/12 [==============================] - 2s 195ms/step - loss: 0.8854 - iou_score: 0.0617 - val_loss: 0.9504 - val_iou_score: 0
Epoch 6/10
12/12 [==============================] - 2s 194ms/step - loss: 0.8813 - iou_score: 0.0639 - val_loss: 0.9500 - val_iou_score: 0
Epoch 7/10
12/12 [==============================] - 2s 195ms/step - loss: 0.8673 - iou_score: 0.0723 - val_loss: 0.9496 - val_iou_score: 0
Epoch 8/10
12/12 [==============================] - 2s 197ms/step - loss: 0.8572 - iou_score: 0.0785 - val_loss: 0.9218 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 196ms/step - loss: 0.8625 - iou_score: 0.0750 - val_loss: 0.9485 - val_iou_score: 0
Epoch 10/10
12/12 [==============================] - 2s 197ms/step - loss: 0.8396 - iou_score: 0.0901 - val_loss: 0.9573 - val_iou_score: 0
<keras.callbacks.History at 0x7efdc7dbc950>
```

diceloss + ADAM + .0001

```
model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 7s 269ms/step - loss: 0.9791 - iou_score: 0.0106 - val_loss: 0.9843 - val_iou_score: 0
Epoch 2/10
12/12 [==============================] - 2s 192ms/step - loss: 0.9663 - iou_score: 0.0172 - val_loss: 0.9739 - val_iou_score: 0
Epoch 3/10
12/12 [==============================] - 2s 194ms/step - loss: 0.9611 - iou_score: 0.0200 - val_loss: 0.9745 - val_iou_score: 0
```

```
Epoch 4/10
12/12 [==============================] - 2s 194ms/step - loss: 0.9546 - iou_score: 0.0234 - val_loss: 0.9733 - val_iou_score: 0
Epoch 5/10
12/12 [==============================] - 2s 194ms/step - loss: 0.9496 - iou_score: 0.0261 - val_loss: 0.9692 - val_iou_score: 0
Epoch 6/10
12/12 [==============================] - 2s 195ms/step - loss: 0.9435 - iou_score: 0.0293 - val_loss: 0.9630 - val_iou_score: 0
Epoch 7/10
12/12 [==============================] - 2s 194ms/step - loss: 0.9398 - iou_score: 0.0313 - val_loss: 0.9608 - val_iou_score: 0
Epoch 8/10
12/12 [==============================] - 2s 194ms/step - loss: 0.9316 - iou_score: 0.0358 - val_loss: 0.9615 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 195ms/step - loss: 0.9249 - iou_score: 0.0394 - val_loss: 0.9529 - val_iou_score: 0
Epoch 10/10
12/12 [==============================] - 2s 195ms/step - loss: 0.9231 - iou_score: 0.0405 - val_loss: 0.9487 - val_iou_score: 0
<keras.callbacks.History at 0x7efdcd2fdad0>
```

```
# diceloss + ADAM + .1

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 8s 340ms/step - loss: 0.9369 - iou_score: 0.0333 - val_loss: 0.9824 - val_iou_score: 0
Epoch 2/10
12/12 [==============================] - 2s 190ms/step - loss: 0.9672 - iou_score: 0.0174 - val_loss: 0.9824 - val_iou_score: 0
Epoch 3/10
12/12 [==============================] - 2s 199ms/step - loss: 0.9959 - iou_score: 0.0021 - val_loss: 0.9824 - val_iou_score: 0
Epoch 4/10
12/12 [==============================] - 2s 199ms/step - loss: 0.7981 - iou_score: 0.1183 - val_loss: 0.9824 - val_iou_score: 0
Epoch 5/10
12/12 [==============================] - 2s 191ms/step - loss: 0.7637 - iou_score: 0.1538 - val_loss: 0.9824 - val_iou_score: 0
Epoch 6/10
12/12 [==============================] - 2s 191ms/step - loss: 0.6725 - iou_score: 0.2052 - val_loss: 0.9824 - val_iou_score: 0
Epoch 7/10
12/12 [==============================] - 2s 193ms/step - loss: 0.6071 - iou_score: 0.2648 - val_loss: 0.9824 - val_iou_score: 0
Epoch 8/10
12/12 [==============================] - 2s 194ms/step - loss: 0.5741 - iou_score: 0.2983 - val_loss: 0.9772 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 193ms/step - loss: 0.6228 - iou_score: 0.2525 - val_loss: 0.9481 - val_iou_score: 0
Epoch 10/10
```

```
12/12 [==============================] - 2s 193ms/step - loss: 0.5669 - iou_score: 0.3014 - val_loss: 1.0000 - val_iou_score: 4
<keras.callbacks.History at 0x7efdcdff0550>
```

```
# diceloss + ADAM + .01

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 7s 264ms/step - loss: 0.9539 - iou_score: 0.0239 - val_loss: 0.9823 - val_iou_score: 0
Epoch 2/10
12/12 [==============================] - 2s 191ms/step - loss: 0.8851 - iou_score: 0.0624 - val_loss: 1.0000 - val_iou_score: 4
Epoch 3/10
12/12 [==============================] - 2s 194ms/step - loss: 0.8309 - iou_score: 0.0960 - val_loss: 0.9791 - val_iou_score: 0
Epoch 4/10
12/12 [==============================] - 2s 193ms/step - loss: 0.6679 - iou_score: 0.2143 - val_loss: 1.0000 - val_iou_score: 4
Epoch 5/10
12/12 [==============================] - 2s 195ms/step - loss: 0.7005 - iou_score: 0.2004 - val_loss: 1.0000 - val_iou_score: 4
Epoch 6/10
12/12 [==============================] - 2s 194ms/step - loss: 0.8129 - iou_score: 0.1205 - val_loss: 1.0000 - val_iou_score: 4
Epoch 7/10
12/12 [==============================] - 2s 193ms/step - loss: 0.9517 - iou_score: 0.0288 - val_loss: 0.9997 - val_iou_score: 1
Epoch 8/10
12/12 [==============================] - 2s 194ms/step - loss: 0.6102 - iou_score: 0.2626 - val_loss: 0.9986 - val_iou_score: 7
Epoch 9/10
12/12 [==============================] - 2s 194ms/step - loss: 0.5454 - iou_score: 0.3157 - val_loss: 1.0000 - val_iou_score: 3
Epoch 10/10
12/12 [==============================] - 2s 194ms/step - loss: 0.5812 - iou_score: 0.2861 - val_loss: 0.7997 - val_iou_score: 0
<keras.callbacks.History at 0x7efdcfde7d90>
```

```
# jaccard + sgd

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 7s 263ms/step - loss: 1.4172 - iou_score: 0.0112 - val_loss: 1.2608 - val_iou_score: 1
Epoch 2/10
```

```
12/12 [==============================] - 2s 192ms/step - loss: 1.1716 - iou_score: 0.0153 - val_loss: 1.1959 - val_iou_score: 1
Epoch 3/10
12/12 [==============================] - 2s 190ms/step - loss: 1.0836 - iou_score: 0.0332 - val_loss: 1.1948 - val_iou_score: 1
Epoch 4/10
12/12 [==============================] - 2s 193ms/step - loss: 1.0363 - iou_score: 0.0504 - val_loss: 1.1927 - val_iou_score: 1
Epoch 5/10
12/12 [==============================] - 2s 193ms/step - loss: 0.9879 - iou_score: 0.0783 - val_loss: 1.1575 - val_iou_score: 1
Epoch 6/10
12/12 [==============================] - 2s 193ms/step - loss: 0.9485 - iou_score: 0.1054 - val_loss: 1.1415 - val_iou_score: 2
Epoch 7/10
12/12 [==============================] - 2s 193ms/step - loss: 0.9070 - iou_score: 0.1476 - val_loss: 1.1390 - val_iou_score: 3
Epoch 8/10
12/12 [==============================] - 2s 195ms/step - loss: 0.8461 - iou_score: 0.1908 - val_loss: 1.1417 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 195ms/step - loss: 0.8250 - iou_score: 0.2204 - val_loss: 1.1342 - val_iou_score: 9
Epoch 10/10
12/12 [==============================] - 2s 196ms/step - loss: 0.8279 - iou_score: 0.2234 - val_loss: 1.1402 - val_iou_score: 1
<keras.callbacks.History at 0x7f001b0e02d0>
```

```
# jaccard

model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader                    )
```

```
Epoch 1/10
12/12 [==============================] - 8s 272ms/step - loss: 1.4190 - iou_score: 0.0118 - val_loss: 1.1974 - val_iou_score: 1
Epoch 2/10
12/12 [==============================] - 2s 192ms/step - loss: 1.1683 - iou_score: 0.0165 - val_loss: 1.1642 - val_iou_score: 2
Epoch 3/10
12/12 [==============================] - 2s 193ms/step - loss: 1.0817 - iou_score: 0.0347 - val_loss: 1.1410 - val_iou_score: 3
Epoch 4/10
12/12 [==============================] - 2s 193ms/step - loss: 1.0270 - iou_score: 0.0618 - val_loss: 1.1885 - val_iou_score: 1
Epoch 5/10
12/12 [==============================] - 2s 200ms/step - loss: 0.9966 - iou_score: 0.0753 - val_loss: 1.1383 - val_iou_score: 3
Epoch 6/10
12/12 [==============================] - 2s 196ms/step - loss: 0.9403 - iou_score: 0.1097 - val_loss: 1.1421 - val_iou_score: 2
Epoch 7/10
12/12 [==============================] - 2s 195ms/step - loss: 0.9151 - iou_score: 0.1363 - val_loss: 1.1402 - val_iou_score: 2
Epoch 8/10
12/12 [==============================] - 2s 196ms/step - loss: 0.8723 - iou_score: 0.1671 - val_loss: 1.1414 - val_iou_score: 2
```

```
Epoch 9/10
12/12 [==============================] - 2s 196ms/step - loss: 0.9150 - iou_score: 0.1431 - val_loss: 1.1396 - val_iou_score: 2
Epoch 10/10
12/12 [==============================] - 2s 197ms/step - loss: 0.8258 - iou_score: 0.2132 - val_loss: 1.1409 - val_iou_score: 1
```

```
# diceloss
callback_list = [checkpoint, learning_rt]
# model.load_weights("/content/model_segmentation.h5")
history = model.fit(train_dataloader, steps_per_epoch=len(train_dataloader), epochs=10,validation_data=test_dataloader
                    )
```

```
Epoch 1/10
12/12 [==============================] - 7s 263ms/step - loss: 0.9604 - iou_score: 0.0204 - val_loss: 1.0000 - val_iou_score: 2
Epoch 2/10
12/12 [==============================] - 2s 189ms/step - loss: 0.9416 - iou_score: 0.0304 - val_loss: 1.0000 - val_iou_score: 4
Epoch 3/10
12/12 [==============================] - 2s 190ms/step - loss: 0.9281 - iou_score: 0.0375 - val_loss: 1.0000 - val_iou_score: 3
Epoch 4/10
12/12 [==============================] - 2s 192ms/step - loss: 0.9019 - iou_score: 0.0520 - val_loss: 1.0000 - val_iou_score: 3
Epoch 5/10
12/12 [==============================] - 2s 191ms/step - loss: 0.8329 - iou_score: 0.0934 - val_loss: 1.0000 - val_iou_score: 4
Epoch 6/10
12/12 [==============================] - 2s 192ms/step - loss: 0.7277 - iou_score: 0.1623 - val_loss: 1.0000 - val_iou_score: 4
Epoch 7/10
12/12 [==============================] - 2s 193ms/step - loss: 0.6447 - iou_score: 0.2260 - val_loss: 0.9531 - val_iou_score: 0
Epoch 8/10
12/12 [==============================] - 2s 194ms/step - loss: 0.6592 - iou_score: 0.2209 - val_loss: 0.9782 - val_iou_score: 0
Epoch 9/10
12/12 [==============================] - 2s 194ms/step - loss: 0.6178 - iou_score: 0.2580 - val_loss: 0.9782 - val_iou_score: 0
Epoch 10/10
12/12 [==============================] - 2s 193ms/step - loss: 0.5645 - iou_score: 0.3006 - val_loss: 0.9640 - val_iou_score: 0
```

5.2 model - inceptionv3_unet

```
# loading unet model with backbone - resnet34

model = sm.Unet('inceptionv3', encoder_weights="imagenet", classes=1,
                activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3))
```

```
Downloading data from https://github.com/fchollet/deep-learning-models/releases/download/v0.5/inception_v3_weights_tf_dim_order
87910968/87910968 [==============================] - 7s 0us/step
```

```
callback_list = create_callback_lists(name = "inceptionv3_UNET")
optim = tf.keras.optimizers.Adam()
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model.compile(optim, loss, metrics=[iou_score])
```

```
train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
callback_list = create_callback_lists("inceptionv3_unet")

history = model.fit(train_data_loader, steps_per_epoch=384, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 96,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/20
384/384 [==============================] - ETA: 0s - loss: 0.5105 - iou_score: 0.3743
Epoch 1: val_iou_score improved from -inf to 0.25847, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 93s 194ms/step - loss: 0.5105 - iou_score: 0.3743 - val_loss: 0.6090 - val_iou_sco
Epoch 2/20
384/384 [==============================] - ETA: 0s - loss: 0.3325 - iou_score: 0.5349
Epoch 2: val_iou_score improved from 0.25847 to 0.55838, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 92s 239ms/step - loss: 0.3325 - iou_score: 0.5349 - val_loss: 0.3183 - val_iou_sco
Epoch 3/20
```

```
384/384 [==============================] - ETA: 0s - loss: 0.2986 - iou_score: 0.5777
Epoch 3: val_iou_score improved from 0.55838 to 0.56096, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 81s 212ms/step - loss: 0.2986 - iou_score: 0.5777 - val_loss: 0.3203 - val_iou_sco
Epoch 4/20
384/384 [==============================] - ETA: 0s - loss: 0.2837 - iou_score: 0.5943
Epoch 4: val_iou_score improved from 0.56096 to 0.60582, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 130s 339ms/step - loss: 0.2837 - iou_score: 0.5943 - val_loss: 0.2772 - val_iou_sc
Epoch 5/20
384/384 [==============================] - ETA: 0s - loss: 0.2523 - iou_score: 0.6315
Epoch 5: val_iou_score did not improve from 0.60582
384/384 [==============================] - 88s 229ms/step - loss: 0.2523 - iou_score: 0.6315 - val_loss: 0.3111 - val_iou_sco
Epoch 6/20
384/384 [==============================] - ETA: 0s - loss: 0.2666 - iou_score: 0.6181
Epoch 6: val_iou_score did not improve from 0.60582
384/384 [==============================] - 81s 210ms/step - loss: 0.2666 - iou_score: 0.6181 - val_loss: 0.3341 - val_iou_sco
Epoch 7/20
384/384 [==============================] - ETA: 0s - loss: 0.2351 - iou_score: 0.6527
Epoch 7: val_iou_score did not improve from 0.60582
384/384 [==============================] - 89s 231ms/step - loss: 0.2351 - iou_score: 0.6527 - val_loss: 0.2866 - val_iou_sco
Epoch 8/20
384/384 [==============================] - ETA: 0s - loss: 0.2176 - iou_score: 0.6693
Epoch 8: val_iou_score improved from 0.60582 to 0.65729, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 91s 238ms/step - loss: 0.2176 - iou_score: 0.6693 - val_loss: 0.2373 - val_iou_sco
Epoch 9/20
384/384 [==============================] - ETA: 0s - loss: 0.2051 - iou_score: 0.6904
Epoch 9: val_iou_score did not improve from 0.65729
384/384 [==============================] - 96s 250ms/step - loss: 0.2051 - iou_score: 0.6904 - val_loss: 0.2959 - val_iou_sco
Epoch 10/20
384/384 [==============================] - ETA: 0s - loss: 0.2265 - iou_score: 0.6685
Epoch 10: val_iou_score did not improve from 0.65729
384/384 [==============================] - 81s 211ms/step - loss: 0.2265 - iou_score: 0.6685 - val_loss: 0.3183 - val_iou_sco
Epoch 11/20
384/384 [==============================] - ETA: 0s - loss: 0.1942 - iou_score: 0.7023
Epoch 11: val_iou_score improved from 0.65729 to 0.69145, saving model to best_model_with_inceptionv3_unet.hdf5
384/384 [==============================] - 90s 235ms/step - loss: 0.1942 - iou_score: 0.7023 - val_loss: 0.2105 - val_iou_sco
Epoch 12/20
384/384 [==============================] - ETA: 0s - loss: 0.1855 - iou_score: 0.7154
Epoch 12: val_iou_score did not improve from 0.69145
384/384 [==============================] - 76s 197ms/step - loss: 0.1855 - iou_score: 0.7154 - val_loss: 0.2209 - val_iou_sco
Epoch 13/20
384/384 [==============================] - ETA: 0s - loss: 0.1851 - iou_score: 0.7185
```

```
Epoch 13: val_iou_score did not improve from 0.69145
384/384 [==============================] - 82s 213ms/step - loss: 0.1851 - iou_score: 0.7185 - val_loss: 0.2163 - val_iou_sco
Epoch 14/20
384/384 [==============================] - ETA: 0s - loss: 0.1789 - iou_score: 0.7200
Epoch 14: val_iou_score did not improve from 0.69145
384/384 [==============================] - 81s 211ms/step - loss: 0.1789 - iou_score: 0.7200 - val_loss: 0.2246 - val_iou_sco
Epoch 15/20
```

```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```

Model iou_score

Model loss

```
# loading tensorboard
%tensorboard --logdir logs/fit
```

TensorBoard SCALARS GRAPHS DISTRIBUTIONS HISTOGRAMS TIME SE INACTIVE

Show data download links

Ignore outliers in chart scalin

Tooltip sorting method: default

Filter tags (regular expressions supported)

epoch_iou_score

5.3 model - efficientnetb1_unet

```
# loading unet model with backbone - resnet34

model3 = sm.Unet('efficientnetb1', encoder_weights="imagenet", classes=1,
                activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3))
```

```
Downloading data from https://github.com/Callidior/keras-applications/releases/download/efficientnet/efficientnet-b1_weights_tf
27164032/27164032 [==============================] - 1s 0us/step
```

```
callback_list = create_callback_lists(name = "efficientnetb1_UNET")
optim = tf.keras.optimizers.Adam()
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model3.compile(optim, loss, metrics=[iou_score])
# model3.load_weights("/content/best_model_with_efficientnetb1_UNET.hdf5")
```

```
# train_data_loader, val_data_loader = data_generator(train_generator, train_mask_generator, val_image_generator, val_mask_generator)
```

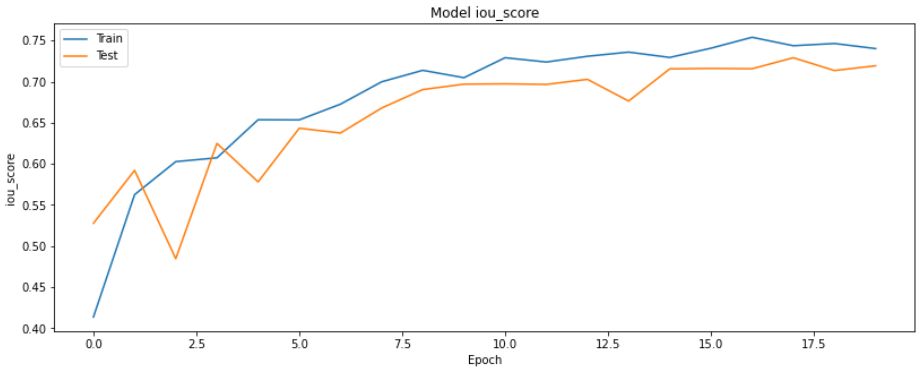
```
history = model3.fit(train_data_loader, steps_per_epoch=384, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 96,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

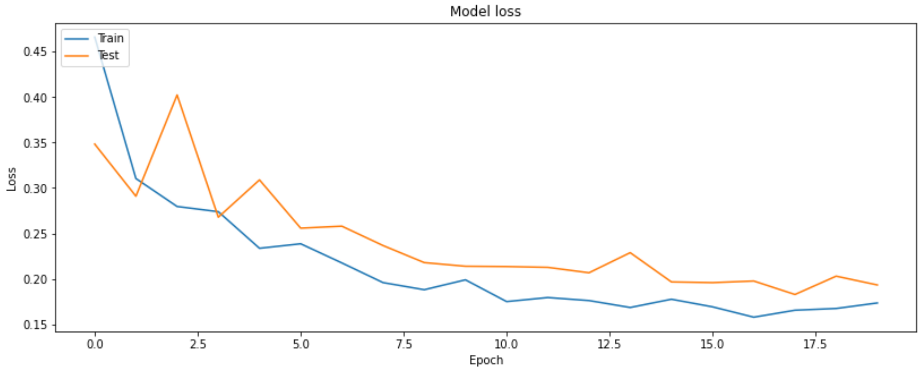
```
Epoch 1/20
384/384 [==============================] - ETA: 0s - loss: 0.4661 - iou_score: 0.4138
Epoch 1: val_iou_score improved from -inf to 0.52775, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 100s 219ms/step - loss: 0.4661 - iou_score: 0.4138 - val_loss: 0.3483 - val_iou_sc
Epoch 2/20
384/384 [==============================] - ETA: 0s - loss: 0.3104 - iou_score: 0.5626
Epoch 2: val_iou_score improved from 0.52775 to 0.59215, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 82s 213ms/step - loss: 0.3104 - iou_score: 0.5626 - val_loss: 0.2909 - val_iou_sco
Epoch 3/20
384/384 [==============================] - ETA: 0s - loss: 0.2796 - iou_score: 0.6027
Epoch 3: val_iou_score did not improve from 0.59215
384/384 [==============================] - 101s 263ms/step - loss: 0.2796 - iou_score: 0.6027 - val_loss: 0.4024 - val_iou_sc
Epoch 4/20
384/384 [==============================] - ETA: 0s - loss: 0.2739 - iou_score: 0.6074
Epoch 4: val_iou_score improved from 0.59215 to 0.62494, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 89s 231ms/step - loss: 0.2739 - iou_score: 0.6074 - val_loss: 0.2678 - val_iou_sco
Epoch 5/20
384/384 [==============================] - ETA: 0s - loss: 0.2336 - iou_score: 0.6536
Epoch 5: val_iou_score did not improve from 0.62494
384/384 [==============================] - 80s 209ms/step - loss: 0.2336 - iou_score: 0.6536 - val_loss: 0.3089 - val_iou_sco
Epoch 6/20
384/384 [==============================] - ETA: 0s - loss: 0.2386 - iou_score: 0.6535
Epoch 6: val_iou_score improved from 0.62494 to 0.64332, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 79s 205ms/step - loss: 0.2386 - iou_score: 0.6535 - val_loss: 0.2558 - val_iou_sco
Epoch 7/20
384/384 [==============================] - ETA: 0s - loss: 0.2175 - iou_score: 0.6726
Epoch 7: val_iou_score did not improve from 0.64332
384/384 [==============================] - 78s 204ms/step - loss: 0.2175 - iou_score: 0.6726 - val_loss: 0.2580 - val_iou_sco
Epoch 8/20
384/384 [==============================] - ETA: 0s - loss: 0.1958 - iou_score: 0.6998
Epoch 8: val_iou_score improved from 0.64332 to 0.66788, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 81s 210ms/step - loss: 0.1958 - iou_score: 0.6998 - val_loss: 0.2366 - val_iou_sco
Epoch 9/20
384/384 [==============================] - ETA: 0s - loss: 0.1880 - iou_score: 0.7138
Epoch 9: val_iou_score improved from 0.66788 to 0.69038, saving model to best_model_with_efficientnetb1_UNET.hdf5
```

```
384/384 [==============================] - 82s 212ms/step - loss: 0.1880 - iou_score: 0.7138 - val_loss: 0.2179 - val_iou_sco
Epoch 10/20
384/384 [==============================] - ETA: 0s - loss: 0.1989 - iou_score: 0.7048
Epoch 10: val_iou_score improved from 0.69038 to 0.69692, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 82s 214ms/step - loss: 0.1989 - iou_score: 0.7048 - val_loss: 0.2138 - val_iou_sco
Epoch 11/20
384/384 [==============================] - ETA: 0s - loss: 0.1749 - iou_score: 0.7291
Epoch 11: val_iou_score improved from 0.69692 to 0.69732, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 84s 218ms/step - loss: 0.1749 - iou_score: 0.7291 - val_loss: 0.2135 - val_iou_sco
Epoch 12/20
384/384 [==============================] - ETA: 0s - loss: 0.1794 - iou_score: 0.7238
Epoch 12: val_iou_score did not improve from 0.69732
384/384 [==============================] - 81s 210ms/step - loss: 0.1794 - iou_score: 0.7238 - val_loss: 0.2126 - val_iou_sco
Epoch 13/20
384/384 [==============================] - ETA: 0s - loss: 0.1761 - iou_score: 0.7308
Epoch 13: val_iou_score improved from 0.69732 to 0.70276, saving model to best_model_with_efficientnetb1_UNET.hdf5
384/384 [==============================] - 80s 207ms/step - loss: 0.1761 - iou_score: 0.7308 - val_loss: 0.2067 - val_iou_sco
Epoch 14/20
384/384 [==============================] - ETA: 0s - loss: 0.1684 - iou_score: 0.7359
Epoch 14: val_iou_score did not improve from 0.70276
384/384 [==============================] - 77s 200ms/step - loss: 0.1684 - iou_score: 0.7359 - val_loss: 0.2288 - val_iou_sco
```

```python
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
```

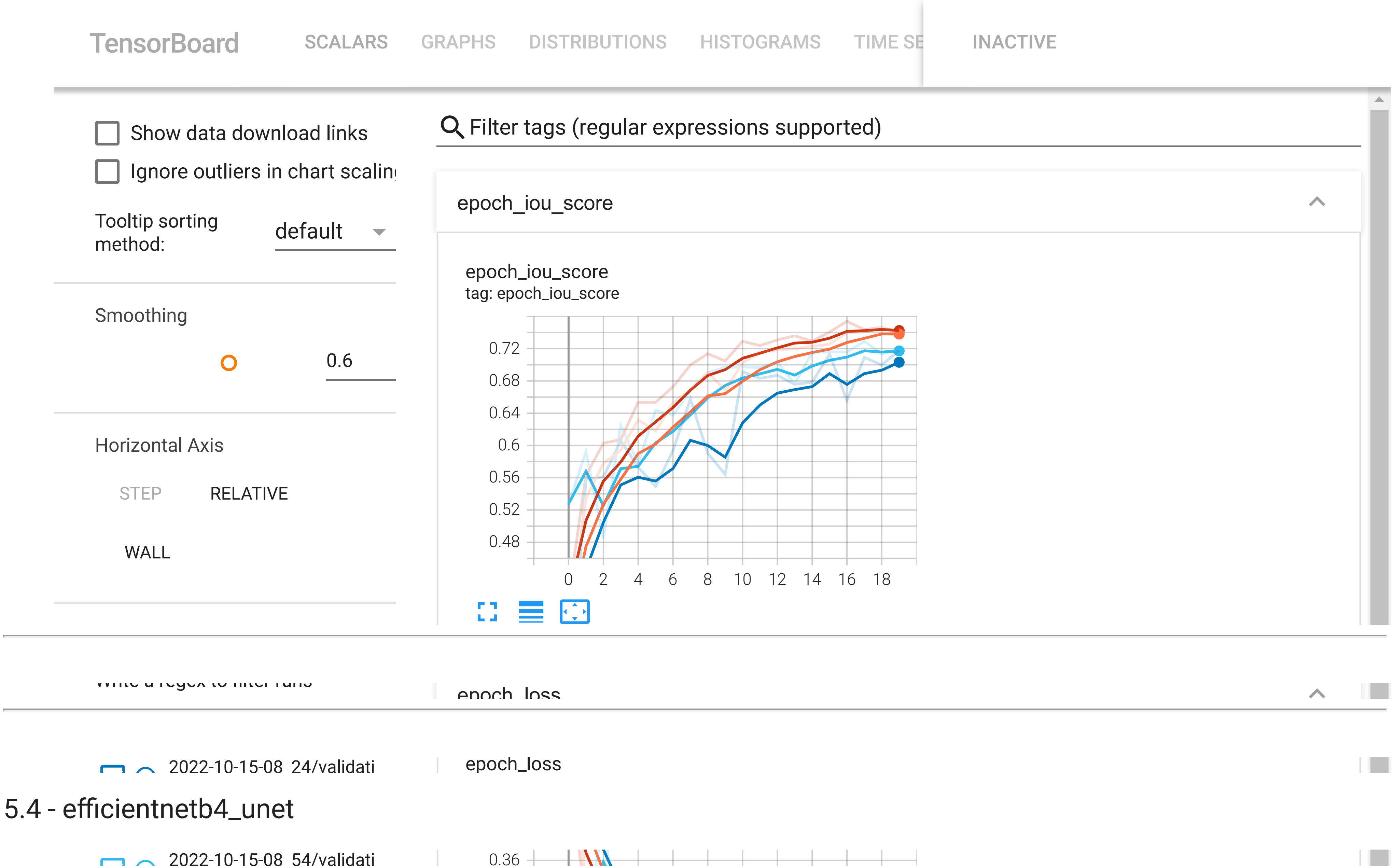
```
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```






```
# loading tensorboard
%tensorboard --logdir logs/fit
```

```
Reusing TensorBoard on port 6006 (pid 491), started 1:33:46 ago. (Use '!kill 491' to kill it.)
```



5.4 - efficientnetb4_unet

```
# loading unet model with backbone - resnet34
```

```
model4 = sm.Unet('efficientnetb4', encoder_weights="imagenet", classes=1,
                activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3))
```

```
Downloading data from https://github.com/Callidior/keras-applications/releases/download/efficientnet/efficientnet-b4_weights_tf
71892840/71892840 [==============================] - 0s 0us/step
```

```
callback_list = create_callback_lists(name = "efficientnetb4_unet")
optim = tf.keras.optimizers.Adam()
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model4.compile(optim, loss, metrics=[iou_score])
```

```
%%time
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=10,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/10
192/192 [==============================] - ETA: 0s - loss: 0.4951 - iou_score: 0.3818
Epoch 1: val_iou_score improved from -inf to 0.14672, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 130s 519ms/step - loss: 0.4951 - iou_score: 0.3818 - val_loss: 0.7477 - val_iou_scor
Epoch 2/10
192/192 [==============================] - ETA: 0s - loss: 0.2211 - iou_score: 0.6452
Epoch 2: val_iou_score improved from 0.14672 to 0.64450, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 103s 539ms/step - loss: 0.2211 - iou_score: 0.6452 - val_loss: 0.2221 - val_iou_scor
Epoch 3/10
192/192 [==============================] - ETA: 0s - loss: 0.2035 - iou_score: 0.6705
Epoch 3: val_iou_score improved from 0.64450 to 0.66512, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 145s 757ms/step - loss: 0.2035 - iou_score: 0.6705 - val_loss: 0.2069 - val_iou_scor
Epoch 4/10
192/192 [==============================] - ETA: 0s - loss: 0.1837 - iou_score: 0.6989
Epoch 4: val_iou_score improved from 0.66512 to 0.67102, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 116s 604ms/step - loss: 0.1837 - iou_score: 0.6989 - val_loss: 0.2047 - val_iou_scor
Epoch 5/10
192/192 [==============================] - ETA: 0s - loss: 0.1703 - iou_score: 0.7185
```

```
Epoch 5: val_iou_score improved from 0.67102 to 0.69105, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 124s 645ms/step - loss: 0.1703 - iou_score: 0.7185 - val_loss: 0.1880 - val_iou_scor
Epoch 6/10
192/192 [==============================] - ETA: 0s - loss: 0.1611 - iou_score: 0.7310
Epoch 6: val_iou_score improved from 0.69105 to 0.71631, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 107s 559ms/step - loss: 0.1611 - iou_score: 0.7310 - val_loss: 0.1694 - val_iou_scor
Epoch 7/10
192/192 [==============================] - ETA: 0s - loss: 0.1387 - iou_score: 0.7612
Epoch 7: val_iou_score improved from 0.71631 to 0.73004, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 107s 559ms/step - loss: 0.1387 - iou_score: 0.7612 - val_loss: 0.1604 - val_iou_scor
Epoch 8/10
192/192 [==============================] - ETA: 0s - loss: 0.1366 - iou_score: 0.7664
Epoch 8: val_iou_score did not improve from 0.73004
192/192 [==============================] - 108s 563ms/step - loss: 0.1366 - iou_score: 0.7664 - val_loss: 0.1652 - val_iou_scor
Epoch 9/10
192/192 [==============================] - ETA: 0s - loss: 0.1670 - iou_score: 0.7251
Epoch 9: val_iou_score did not improve from 0.73004
192/192 [==============================] - 101s 526ms/step - loss: 0.1670 - iou_score: 0.7251 - val_loss: 0.2779 - val_iou_scor
Epoch 10/10
192/192 [==============================] - ETA: 0s - loss: 0.1368 - iou_score: 0.7641
Epoch 10: val_iou_score did not improve from 0.73004
192/192 [==============================] - 99s 518ms/step - loss: 0.1368 - iou_score: 0.7641 - val_loss: 0.1675 - val_iou_score
```

```
# model4.save_weights("/content/drive/MyDrive/dl_model_save/efficientnetb4_unet_72.h5")
```

```
%%time
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 10, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 11/20
192/192 [==============================] - ETA: 0s - loss: 0.1273 - iou_score: 0.7784
Epoch 11: val_iou_score improved from 0.73004 to 0.74522, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 157s 813ms/step - loss: 0.1273 - iou_score: 0.7784 - val_loss: 0.1508 - val_iou_scor
Epoch 12/20
192/192 [==============================] - ETA: 0s - loss: 0.1144 - iou_score: 0.7976
```

```
Epoch 12: val_iou_score improved from 0.74522 to 0.75033, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 112s 583ms/step - loss: 0.1144 - iou_score: 0.7976 - val_loss: 0.1469 - val_iou_scor
Epoch 13/20
192/192 [==============================] - ETA: 0s - loss: 0.1150 - iou_score: 0.7966
Epoch 13: val_iou_score improved from 0.75033 to 0.76586, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 109s 569ms/step - loss: 0.1150 - iou_score: 0.7966 - val_loss: 0.1360 - val_iou_scor
Epoch 14/20
192/192 [==============================] - ETA: 0s - loss: 0.1108 - iou_score: 0.8038
Epoch 14: val_iou_score improved from 0.76586 to 0.76823, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 108s 561ms/step - loss: 0.1108 - iou_score: 0.8038 - val_loss: 0.1343 - val_iou_scor
Epoch 15/20
192/192 [==============================] - ETA: 0s - loss: 0.1095 - iou_score: 0.8066
Epoch 15: val_iou_score improved from 0.76823 to 0.77001, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 102s 530ms/step - loss: 0.1095 - iou_score: 0.8066 - val_loss: 0.1331 - val_iou_scor
Epoch 16/20
192/192 [==============================] - ETA: 0s - loss: 0.1098 - iou_score: 0.8049
Epoch 16: val_iou_score improved from 0.77001 to 0.77651, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 102s 533ms/step - loss: 0.1098 - iou_score: 0.8049 - val_loss: 0.1288 - val_iou_scor
Epoch 17/20
192/192 [==============================] - ETA: 0s - loss: 0.1070 - iou_score: 0.8093
Epoch 17: val_iou_score did not improve from 0.77651
192/192 [==============================] - 102s 531ms/step - loss: 0.1070 - iou_score: 0.8093 - val_loss: 0.1355 - val_iou_scor
Epoch 18/20
192/192 [==============================] - ETA: 0s - loss: 0.1042 - iou_score: 0.8141
Epoch 18: val_iou_score improved from 0.77651 to 0.77720, saving model to best_model_with_efficientnetb4_unet.hdf5
192/192 [==============================] - 106s 552ms/step - loss: 0.1042 - iou_score: 0.8141 - val_loss: 0.1291 - val_iou_scor
Epoch 19/20
192/192 [==============================] - ETA: 0s - loss: 0.1095 - iou_score: 0.8062
Epoch 19: val_iou_score did not improve from 0.77720
192/192 [==============================] - 121s 628ms/step - loss: 0.1095 - iou_score: 0.8062 - val_loss: 0.1294 - val_iou_scor
Epoch 20/20
192/192 [==============================] - ETA: 0s - loss: 0.0980 - iou_score: 0.8234
Epoch 20: val_iou_score did not improve from 0.77720
192/192 [==============================] - 107s 555ms/step - loss: 0.0980 - iou_score: 0.8234 - val_loss: 0.1310 - val_iou_scor
CPU times: user 14min 34s, sys: 8min 11s, total: 22min 46s
Wall time: 18min 45s
```

```
# callback_list = create_callback_lists(name = "efficientnetb4_unet")
optim = tf.keras.optimizers.Adam(learning_rate = 0.006)
```

```
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model4.compile(optim, loss, metrics=[iou_score])


history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=40,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 20,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 21/40
192/192 [==============================] - 93s 481ms/step - loss: 0.0993 - iou_score: 0.8212 - val_loss: 0.1185 - val_iou_score
Epoch 22/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0915 - iou_score: 0.8338 - val_loss: 0.1359 - val_iou_score
Epoch 23/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0993 - iou_score: 0.8218 - val_loss: 0.1196 - val_iou_score
Epoch 24/40
192/192 [==============================] - 92s 479ms/step - loss: 0.1001 - iou_score: 0.8203 - val_loss: 0.1293 - val_iou_score
Epoch 25/40
192/192 [==============================] - 92s 479ms/step - loss: 0.0963 - iou_score: 0.8277 - val_loss: 0.1177 - val_iou_score
Epoch 26/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0995 - iou_score: 0.8211 - val_loss: 0.1207 - val_iou_score
Epoch 27/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0933 - iou_score: 0.8311 - val_loss: 0.1358 - val_iou_score
Epoch 28/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0951 - iou_score: 0.8285 - val_loss: 0.1365 - val_iou_score
Epoch 29/40
192/192 [==============================] - 93s 482ms/step - loss: 0.1223 - iou_score: 0.7912 - val_loss: 0.1282 - val_iou_score
Epoch 30/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0948 - iou_score: 0.8291 - val_loss: 0.1216 - val_iou_score
Epoch 31/40
192/192 [==============================] - 92s 480ms/step - loss: 0.1106 - iou_score: 0.8083 - val_loss: 0.1219 - val_iou_score
Epoch 32/40
192/192 [==============================] - 92s 481ms/step - loss: 0.0893 - iou_score: 0.8379 - val_loss: 0.1152 - val_iou_score
Epoch 33/40
192/192 [==============================] - 92s 481ms/step - loss: 0.0981 - iou_score: 0.8240 - val_loss: 0.1208 - val_iou_score
Epoch 34/40
192/192 [==============================] - 92s 481ms/step - loss: 0.1077 - iou_score: 0.8129 - val_loss: 0.1241 - val_iou_score
Epoch 35/40
192/192 [==============================] - 92s 480ms/step - loss: 0.0914 - iou_score: 0.8344 - val_loss: 0.1107 - val_iou_score
```

```
Epoch 36/40
192/192 [==============================] - 93s 483ms/step - loss: 0.0883 - iou_score: 0.8390 - val_loss: 0.1116 - val_iou_score
Epoch 37/40
192/192 [==============================] - 93s 482ms/step - loss: 0.0934 - iou_score: 0.8312 - val_loss: 0.1200 - val_iou_score
Epoch 38/40
192/192 [==============================] - 93s 482ms/step - loss: 0.0975 - iou_score: 0.8250 - val_loss: 0.1246 - val_iou_score
Epoch 39/40
192/192 [==============================] - 93s 484ms/step - loss: 0.0899 - iou_score: 0.8370 - val_loss: 0.1204 - val_iou_score
Epoch 40/40
192/192 [==============================] - 94s 491ms/step - loss: 0.0933 - iou_score: 0.8321 - val_loss: 0.1177 - val_iou_score
```

```
# model4
model4.save_weights("efficientnetb4_unet_79.hdf5")


# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```

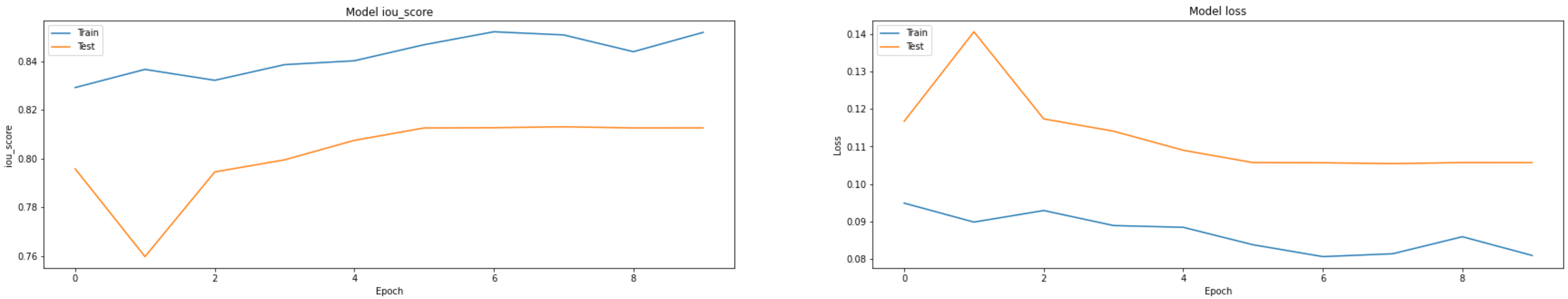

```
callback_list = create_callback_lists(name = "efficientnetb4_unet")
optim = tf.keras.optimizers.Adam(learning_rate = 0.008)
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model4.compile(optim, loss, metrics=[iou_score])
model4.load_weights("/content/drive/MyDrive/dl_model_save/efficientnetb4_unet_79.h5")



# model4.save_weights("efficientnetb4_unet_79.h5")


%%time
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=10,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/10
192/192 [==============================] - ETA: 0s - loss: 0.0949 - iou_score: 0.8292
Epoch 1: val_iou_score improved from -inf to 0.79579, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 132s 599ms/step - loss: 0.0949 - iou_score: 0.8292 - val_loss: 0.1167 - val_iou_scor
Epoch 2/10
192/192 [==============================] - ETA: 0s - loss: 0.0898 - iou_score: 0.8367
Epoch 2: val_iou_score did not improve from 0.79579
192/192 [==============================] - 105s 549ms/step - loss: 0.0898 - iou_score: 0.8367 - val_loss: 0.1406 - val_iou_scor
Epoch 3/10
192/192 [==============================] - ETA: 0s - loss: 0.0929 - iou_score: 0.8322
Epoch 3: val_iou_score did not improve from 0.79579
192/192 [==============================] - 102s 533ms/step - loss: 0.0929 - iou_score: 0.8322 - val_loss: 0.1173 - val_iou_scor
```

```
Epoch 4/10
192/192 [==============================] - ETA: 0s - loss: 0.0889 - iou_score: 0.8386
Epoch 4: val_iou_score improved from 0.79579 to 0.79948, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 104s 541ms/step - loss: 0.0889 - iou_score: 0.8386 - val_loss: 0.1141 - val_iou_scor
Epoch 5/10
192/192 [==============================] - ETA: 0s - loss: 0.0884 - iou_score: 0.8402
Epoch 5: val_iou_score improved from 0.79948 to 0.80749, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 102s 532ms/step - loss: 0.0884 - iou_score: 0.8402 - val_loss: 0.1090 - val_iou_scor
Epoch 6/10
192/192 [==============================] - ETA: 0s - loss: 0.0838 - iou_score: 0.8468
Epoch 6: val_iou_score improved from 0.80749 to 0.81259, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 101s 526ms/step - loss: 0.0838 - iou_score: 0.8468 - val_loss: 0.1057 - val_iou_scor
Epoch 7/10
192/192 [==============================] - ETA: 0s - loss: 0.0806 - iou_score: 0.8522
Epoch 7: val_iou_score improved from 0.81259 to 0.81269, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 100s 520ms/step - loss: 0.0806 - iou_score: 0.8522 - val_loss: 0.1056 - val_iou_scor
Epoch 8/10
192/192 [==============================] - ETA: 0s - loss: 0.0814 - iou_score: 0.8509
Epoch 8: val_iou_score improved from 0.81269 to 0.81308, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 105s 546ms/step - loss: 0.0814 - iou_score: 0.8509 - val_loss: 0.1054 - val_iou_scor
Epoch 9/10
192/192 [==============================] - ETA: 0s - loss: 0.0859 - iou_score: 0.8440
Epoch 9: val_iou_score did not improve from 0.81308
192/192 [==============================] - 128s 668ms/step - loss: 0.0859 - iou_score: 0.8440 - val_loss: 0.1057 - val_iou_scor
Epoch 10/10
192/192 [==============================] - ETA: 0s - loss: 0.0809 - iou_score: 0.8519
Epoch 10: val_iou_score did not improve from 0.81308
192/192 [==============================] - 120s 625ms/step - loss: 0.0809 - iou_score: 0.8519 - val_loss: 0.1057 - val_iou_scor
CPU times: user 14min 36s, sys: 7min 53s, total: 22min 30s
Wall time: 18min 19s
```

```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
```

```
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```




```
%%time
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=50,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 40, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

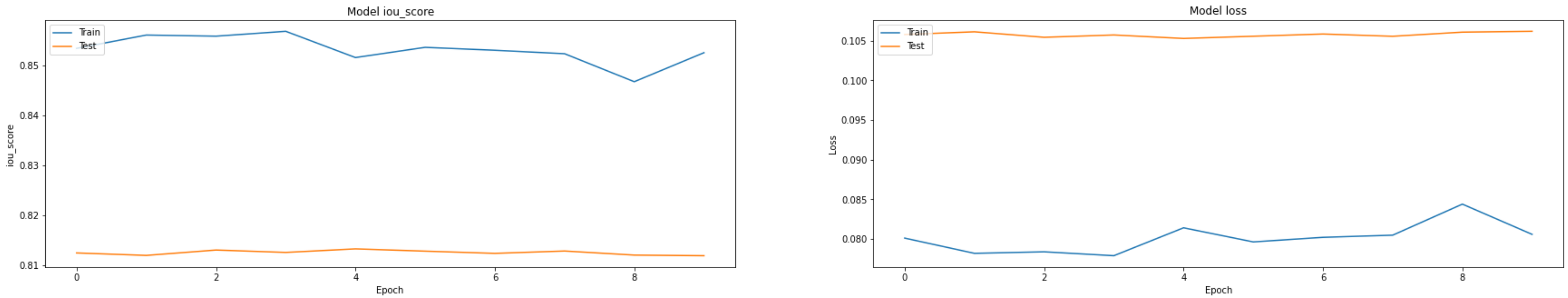
```
Epoch 41/50
192/192 [==============================] - ETA: 0s - loss: 0.0801 - iou_score: 0.8534
Epoch 41: val_iou_score did not improve from 0.81308
192/192 [==============================] - 173s 895ms/step - loss: 0.0801 - iou_score: 0.8534 - val_loss: 0.1058 - val_iou_scor
Epoch 42/50
192/192 [==============================] - ETA: 0s - loss: 0.0782 - iou_score: 0.8561
Epoch 42: val_iou_score did not improve from 0.81308
192/192 [==============================] - 107s 556ms/step - loss: 0.0782 - iou_score: 0.8561 - val_loss: 0.1061 - val_iou_scor
```

```
Epoch 43/50
192/192 [==============================] - ETA: 0s - loss: 0.0784 - iou_score: 0.8558
Epoch 43: val_iou_score did not improve from 0.81308
192/192 [==============================] - 127s 664ms/step - loss: 0.0784 - iou_score: 0.8558 - val_loss: 0.1054 - val_iou_scor
Epoch 44/50
192/192 [==============================] - ETA: 0s - loss: 0.0779 - iou_score: 0.8568
Epoch 44: val_iou_score did not improve from 0.81308
192/192 [==============================] - 128s 666ms/step - loss: 0.0779 - iou_score: 0.8568 - val_loss: 0.1057 - val_iou_scor
Epoch 45/50
192/192 [==============================] - ETA: 0s - loss: 0.0814 - iou_score: 0.8516
Epoch 45: val_iou_score improved from 0.81308 to 0.81323, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 122s 636ms/step - loss: 0.0814 - iou_score: 0.8516 - val_loss: 0.1053 - val_iou_scor
Epoch 46/50
192/192 [==============================] - ETA: 0s - loss: 0.0796 - iou_score: 0.8536
Epoch 46: val_iou_score did not improve from 0.81323
192/192 [==============================] - 120s 624ms/step - loss: 0.0796 - iou_score: 0.8536 - val_loss: 0.1056 - val_iou_scor
Epoch 47/50
192/192 [==============================] - ETA: 0s - loss: 0.0802 - iou_score: 0.8530
Epoch 47: val_iou_score did not improve from 0.81323
192/192 [==============================] - 115s 601ms/step - loss: 0.0802 - iou_score: 0.8530 - val_loss: 0.1059 - val_iou_scor
Epoch 48/50
192/192 [==============================] - ETA: 0s - loss: 0.0805 - iou_score: 0.8523
Epoch 48: val_iou_score did not improve from 0.81323
192/192 [==============================] - 102s 529ms/step - loss: 0.0805 - iou_score: 0.8523 - val_loss: 0.1056 - val_iou_scor
Epoch 49/50
192/192 [==============================] - ETA: 0s - loss: 0.0844 - iou_score: 0.8467
Epoch 49: val_iou_score did not improve from 0.81323
192/192 [==============================] - 102s 530ms/step - loss: 0.0844 - iou_score: 0.8467 - val_loss: 0.1061 - val_iou_scor
Epoch 50/50
192/192 [==============================] - ETA: 0s - loss: 0.0806 - iou_score: 0.8525
Epoch 50: val_iou_score did not improve from 0.81323
192/192 [==============================] - 102s 530ms/step - loss: 0.0806 - iou_score: 0.8525 - val_loss: 0.1062 - val_iou_scor
CPU times: user 14min 21s, sys: 9min 36s, total: 23min 58s
Wall time: 19min 57s
```

```
# model4.save_weights("/content/drive/MyDrive/dl_model_save/efficientnetb4_unet_81.h5")
```

```
# Plot training & validation iou_score values
```

```
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```




```
callback_list = create_callback_lists(name = "efficientnetb4_unet")
optim = tf.keras.optimizers.Adam(learning_rate = 0.008)
loss =  DiceLoss()#sm.losses.bce_jaccard_loss
```

```
model4.compile(optim, loss, metrics=[iou_score])
model4.load_weights("/content/best_model_with_efficientnetb4_unet.h5")
```

```
%%time
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=57,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 50, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 51/57
192/192 [==============================] - ETA: 0s - loss: 0.0869 - iou_score: 0.8423
Epoch 51: val_iou_score improved from -inf to 0.76627, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 110s 569ms/step - loss: 0.0869 - iou_score: 0.8423 - val_loss: 0.1361 - val_iou_scor
Epoch 52/57
192/192 [==============================] - ETA: 0s - loss: 0.0883 - iou_score: 0.8394
Epoch 52: val_iou_score improved from 0.76627 to 0.77983, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 130s 680ms/step - loss: 0.0883 - iou_score: 0.8394 - val_loss: 0.1264 - val_iou_scor
Epoch 53/57
192/192 [==============================] - ETA: 0s - loss: 0.0903 - iou_score: 0.8367
Epoch 53: val_iou_score improved from 0.77983 to 0.80219, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 142s 743ms/step - loss: 0.0903 - iou_score: 0.8367 - val_loss: 0.1125 - val_iou_scor
Epoch 54/57
192/192 [==============================] - ETA: 0s - loss: 0.0846 - iou_score: 0.8458
Epoch 54: val_iou_score improved from 0.80219 to 0.80394, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 115s 601ms/step - loss: 0.0846 - iou_score: 0.8458 - val_loss: 0.1113 - val_iou_scor
Epoch 55/57
192/192 [==============================] - ETA: 0s - loss: 0.0861 - iou_score: 0.8438
Epoch 55: val_iou_score improved from 0.80394 to 0.80464, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 106s 554ms/step - loss: 0.0861 - iou_score: 0.8438 - val_loss: 0.1109 - val_iou_scor
Epoch 56/57
192/192 [==============================] - ETA: 0s - loss: 0.0854 - iou_score: 0.8442
Epoch 56: val_iou_score improved from 0.80464 to 0.80532, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 104s 540ms/step - loss: 0.0854 - iou_score: 0.8442 - val_loss: 0.1105 - val_iou_scor
Epoch 57/57
192/192 [==============================] - ETA: 0s - loss: 0.0830 - iou_score: 0.8482
Epoch 57: val_iou_score improved from 0.80532 to 0.80556, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 104s 544ms/step - loss: 0.0830 - iou_score: 0.8482 - val_loss: 0.1103 - val_iou_scor
```

```
CPU times: user 10min 21s, sys: 5min 46s, total: 16min 8s
Wall time: 13min 34s
```

```
%%time
callback_list = create_callback_lists(name = "efficientnetb4_unet")
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=60,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 57, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 58/60
192/192 [==============================] - ETA: 0s - loss: 0.0790 - iou_score: 0.8553
Epoch 58: val_iou_score improved from -inf to 0.80629, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 129s 666ms/step - loss: 0.0790 - iou_score: 0.8553 - val_loss: 0.1098 - val_iou_scor
Epoch 59/60
192/192 [==============================] - ETA: 0s - loss: 0.0775 - iou_score: 0.8572
Epoch 59: val_iou_score improved from 0.80629 to 0.80656, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 116s 604ms/step - loss: 0.0775 - iou_score: 0.8572 - val_loss: 0.1097 - val_iou_scor
Epoch 60/60
192/192 [==============================] - ETA: 0s - loss: 0.0790 - iou_score: 0.8546
Epoch 60: val_iou_score improved from 0.80656 to 0.80777, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 117s 611ms/step - loss: 0.0790 - iou_score: 0.8546 - val_loss: 0.1089 - val_iou_scor
CPU times: user 4min 24s, sys: 3min 1s, total: 7min 26s
Wall time: 6min 2s
```

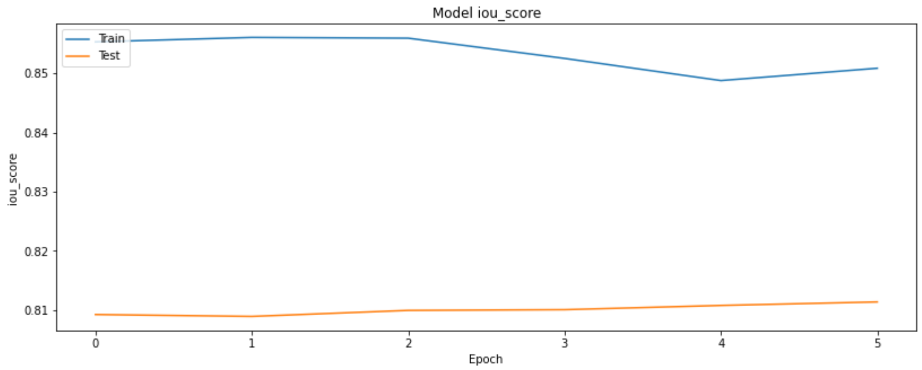
```
%%time
callback_list = create_callback_lists(name = "efficientnetb4_unet")
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=66,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 60, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

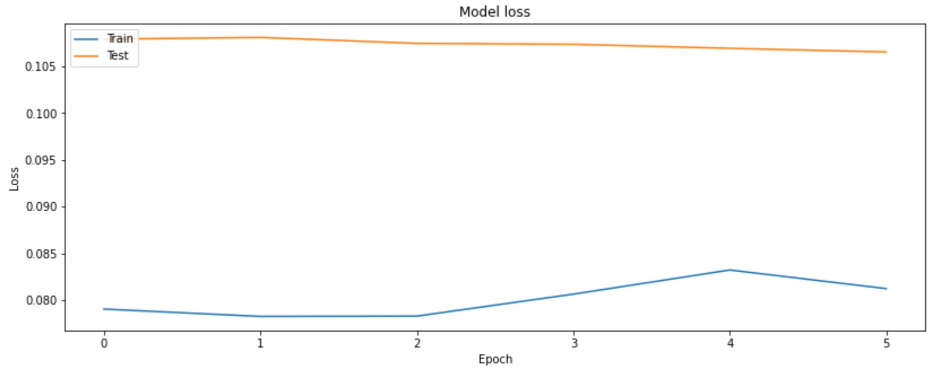
```
Epoch 61/66
192/192 [==============================] - ETA: 0s - loss: 0.0790 - iou_score: 0.8553
Epoch 61: val_iou_score improved from -inf to 0.80925, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 115s 594ms/step - loss: 0.0790 - iou_score: 0.8553 - val_loss: 0.1079 - val_iou_scor
```

```
Epoch 62/66
192/192 [==============================] - ETA: 0s - loss: 0.0783 - iou_score: 0.8561
Epoch 62: val_iou_score did not improve from 0.80925
192/192 [==============================] - 99s 518ms/step - loss: 0.0783 - iou_score: 0.8561 - val_loss: 0.1081 - val_iou_score
Epoch 63/66
192/192 [==============================] - ETA: 0s - loss: 0.0783 - iou_score: 0.8559
Epoch 63: val_iou_score improved from 0.80925 to 0.80994, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 104s 542ms/step - loss: 0.0783 - iou_score: 0.8559 - val_loss: 0.1075 - val_iou_scor
Epoch 64/66
192/192 [==============================] - ETA: 0s - loss: 0.0806 - iou_score: 0.8525
Epoch 64: val_iou_score improved from 0.80994 to 0.81007, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 130s 680ms/step - loss: 0.0806 - iou_score: 0.8525 - val_loss: 0.1074 - val_iou_scor
Epoch 65/66
192/192 [==============================] - ETA: 0s - loss: 0.0832 - iou_score: 0.8488
Epoch 65: val_iou_score improved from 0.81007 to 0.81078, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 124s 646ms/step - loss: 0.0832 - iou_score: 0.8488 - val_loss: 0.1069 - val_iou_scor
Epoch 66/66
192/192 [==============================] - ETA: 0s - loss: 0.0812 - iou_score: 0.8509
Epoch 66: val_iou_score improved from 0.81078 to 0.81137, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 104s 539ms/step - loss: 0.0812 - iou_score: 0.8509 - val_loss: 0.1066 - val_iou_scor
CPU times: user 8min 49s, sys: 4min 49s, total: 13min 39s
Wall time: 11min 22s
```

```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
```

```
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```






```
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=70,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 66, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 67/70
192/192 [==============================] - ETA: 0s - loss: 0.0767 - iou_score: 0.8591
Epoch 67: val_iou_score did not improve from 0.81137
192/192 [==============================] - 103s 533ms/step - loss: 0.0767 - iou_score: 0.8591 - val_loss: 0.1097 - val_iou_scor
Epoch 68/70
192/192 [==============================] - ETA: 0s - loss: 0.0760 - iou_score: 0.8598
Epoch 68: val_iou_score did not improve from 0.81137
192/192 [==============================] - 102s 534ms/step - loss: 0.0760 - iou_score: 0.8598 - val_loss: 0.1096 - val_iou_scor
Epoch 69/70
192/192 [==============================] - ETA: 0s - loss: 0.0772 - iou_score: 0.8578
Epoch 69: val_iou_score did not improve from 0.81137
192/192 [==============================] - 101s 525ms/step - loss: 0.0772 - iou_score: 0.8578 - val_loss: 0.1096 - val_iou_scor
Epoch 70/70
192/192 [==============================] - ETA: 0s - loss: 0.0801 - iou_score: 0.8532
Epoch 70: val_iou_score did not improve from 0.81137
192/192 [==============================] - 107s 560ms/step - loss: 0.0801 - iou_score: 0.8532 - val_loss: 0.1093 - val_iou_scor
```

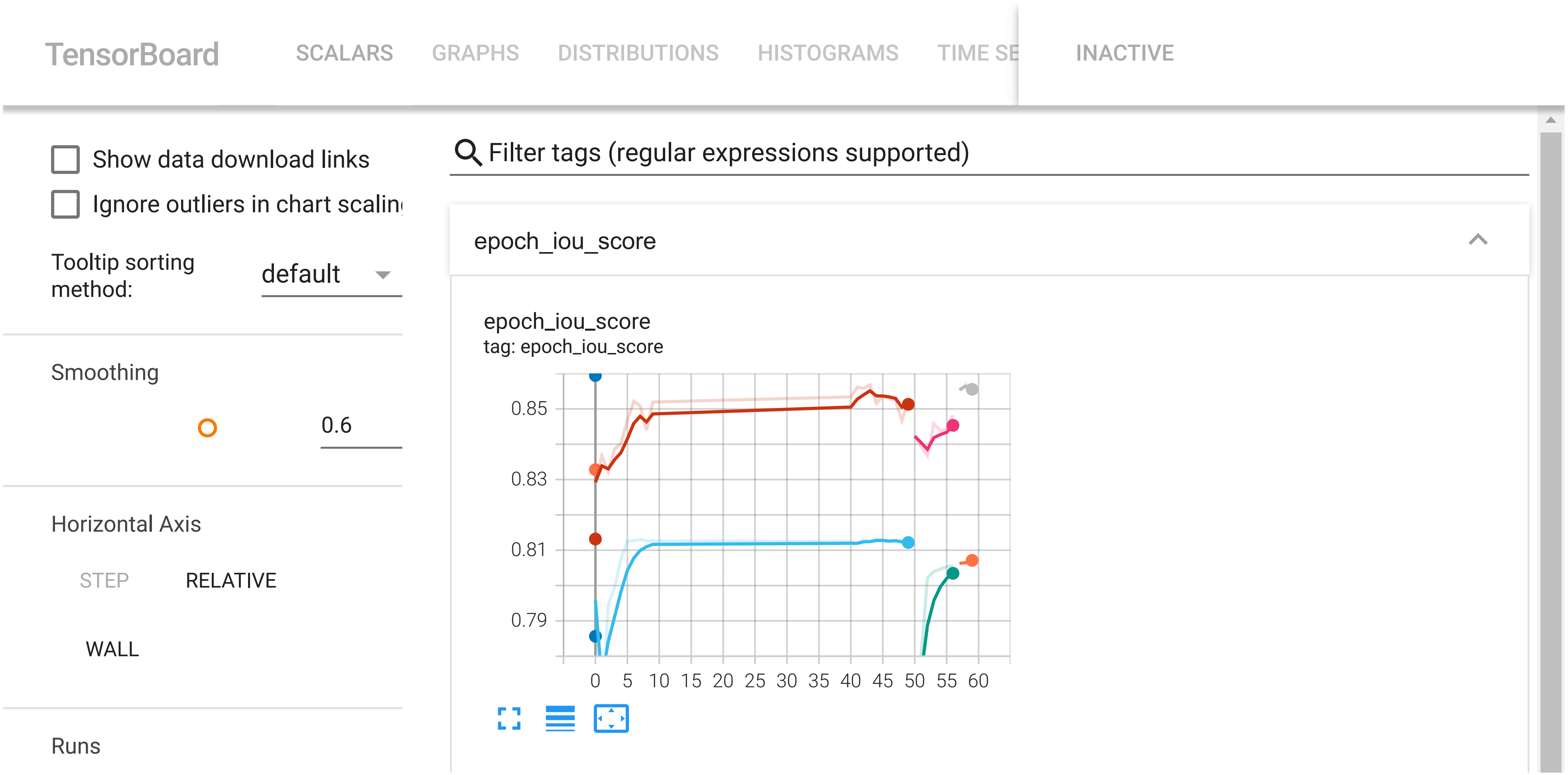
```
model4.save_weights("/content/drive/MyDrive/dl_model_save/efficientnetb4_unet_81.h5")
```

```
history = model4.fit(train_data_loader, steps_per_epoch=192, epochs=,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, callbacks = callback_list, use_multiprocessing = True )#callbacks = callback_list,
```

```
192/192 [==============================] - ETA: 0s - loss: 0.0765 - iou_score: 0.8595
Epoch 1: val_iou_score improved from 0.81137 to 0.81317, saving model to best_model_with_efficientnetb4_unet.h5
192/192 [==============================] - 105s 543ms/step - loss: 0.0765 - iou_score: 0.8595 - val_loss: 0.1055 - val_iou_scor
```

tensorboard

```
# loading tensorboard
%tensorboard --logdir logs/fit
```



highest accuracy got is .81 IOU score

20 epoch result

```
history = model4.fit(train_data_loader, steps_per_epoch=384, epochs=20,
                     validation_data = val_data_loader,  validation_steps = 96,
```

```
                  initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/20
384/384 [==============================] - ETA: 0s - loss: 0.4677 - iou_score: 0.4125
Epoch 1: val_iou_score improved from -inf to 0.45757, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 153s 324ms/step - loss: 0.4677 - iou_score: 0.4125 - val_loss: 0.4276 - val_iou_sc
Epoch 2/20
384/384 [==============================] - ETA: 0s - loss: 0.3037 - iou_score: 0.5707
Epoch 2: val_iou_score improved from 0.45757 to 0.60541, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 128s 334ms/step - loss: 0.3037 - iou_score: 0.5707 - val_loss: 0.2813 - val_iou_sc
Epoch 3/20
384/384 [==============================] - ETA: 0s - loss: 0.2938 - iou_score: 0.5854
Epoch 3: val_iou_score did not improve from 0.60541
384/384 [==============================] - 155s 403ms/step - loss: 0.2938 - iou_score: 0.5854 - val_loss: 0.2999 - val_iou_sc
Epoch 4/20
384/384 [==============================] - ETA: 0s - loss: 0.2696 - iou_score: 0.6130
Epoch 4: val_iou_score did not improve from 0.60541
384/384 [==============================] - 155s 403ms/step - loss: 0.2696 - iou_score: 0.6130 - val_loss: 0.2932 - val_iou_sc
Epoch 5/20
384/384 [==============================] - ETA: 0s - loss: 0.2347 - iou_score: 0.6512
Epoch 5: val_iou_score improved from 0.60541 to 0.63673, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 157s 408ms/step - loss: 0.2347 - iou_score: 0.6512 - val_loss: 0.2605 - val_iou_sc
Epoch 6/20
384/384 [==============================] - ETA: 0s - loss: 0.2388 - iou_score: 0.6514
Epoch 6: val_iou_score did not improve from 0.63673
384/384 [==============================] - 152s 397ms/step - loss: 0.2388 - iou_score: 0.6514 - val_loss: 0.3094 - val_iou_sc
Epoch 7/20
384/384 [==============================] - ETA: 0s - loss: 0.2078 - iou_score: 0.6849
Epoch 7: val_iou_score improved from 0.63673 to 0.67931, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 135s 352ms/step - loss: 0.2078 - iou_score: 0.6849 - val_loss: 0.2257 - val_iou_sc
Epoch 8/20
384/384 [==============================] - ETA: 0s - loss: 0.1812 - iou_score: 0.7187
Epoch 8: val_iou_score improved from 0.67931 to 0.70483, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 126s 328ms/step - loss: 0.1812 - iou_score: 0.7187 - val_loss: 0.2047 - val_iou_sc
Epoch 9/20
384/384 [==============================] - ETA: 0s - loss: 0.1820 - iou_score: 0.7219
Epoch 9: val_iou_score did not improve from 0.70483
384/384 [==============================] - 133s 347ms/step - loss: 0.1820 - iou_score: 0.7219 - val_loss: 0.2114 - val_iou_sc
Epoch 10/20
384/384 [==============================] - ETA: 0s - loss: 0.1961 - iou_score: 0.7109
```

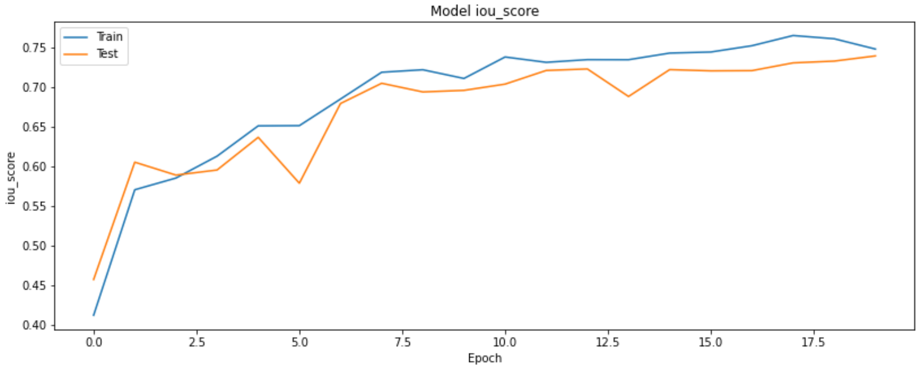
```
Epoch 10: val_iou_score did not improve from 0.70483
384/384 [==============================] - 125s 324ms/step - loss: 0.1961 - iou_score: 0.7109 - val_loss: 0.2121 - val_iou_sc
Epoch 11/20
384/384 [==============================] - ETA: 0s - loss: 0.1687 - iou_score: 0.7379
Epoch 11: val_iou_score did not improve from 0.70483
384/384 [==============================] - 122s 318ms/step - loss: 0.1687 - iou_score: 0.7379 - val_loss: 0.2057 - val_iou_sc
Epoch 12/20
384/384 [==============================] - ETA: 0s - loss: 0.1736 - iou_score: 0.7312
Epoch 12: val_iou_score improved from 0.70483 to 0.72098, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 135s 351ms/step - loss: 0.1736 - iou_score: 0.7312 - val_loss: 0.1915 - val_iou_sc
Epoch 13/20
384/384 [==============================] - ETA: 0s - loss: 0.1736 - iou_score: 0.7346
Epoch 13: val_iou_score improved from 0.72098 to 0.72286, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 123s 321ms/step - loss: 0.1736 - iou_score: 0.7346 - val_loss: 0.1858 - val_iou_sc
Epoch 14/20
384/384 [==============================] - ETA: 0s - loss: 0.1693 - iou_score: 0.7345
Epoch 14: val_iou_score did not improve from 0.72286
384/384 [==============================] - 123s 320ms/step - loss: 0.1693 - iou_score: 0.7345 - val_loss: 0.2203 - val_iou_sc
Epoch 15/20
```

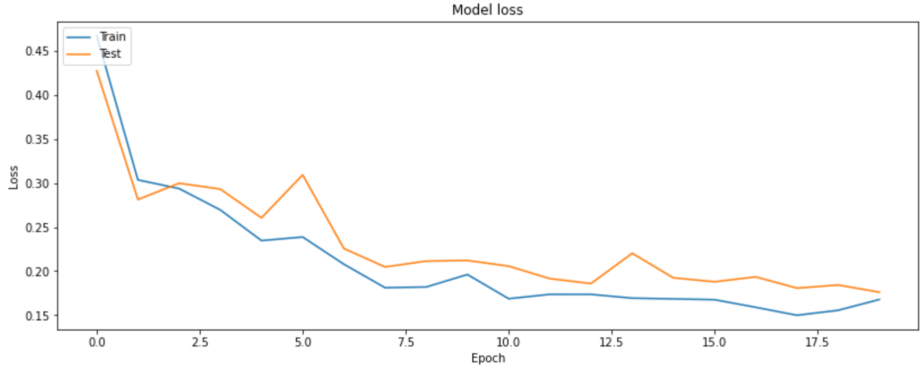
```
model4.load_weights("/content/best_model_with_efficientnetb4_unet.hdf5")
history = model4.fit(train_data_loader, steps_per_epoch=384, epochs=1,
                    validation_data = val_data_loader,  validation_steps = 96,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
384/384 [==============================] - ETA: 0s - loss: 0.1557 - iou_score: 0.7663
Epoch 1: val_iou_score improved from 0.73766 to 0.74307, saving model to best_model_with_efficientnetb4_unet.hdf5
384/384 [==============================] - 172s 444ms/step - loss: 0.1557 - iou_score: 0.7663 - val_loss: 0.1757 - val_iou_scor
```

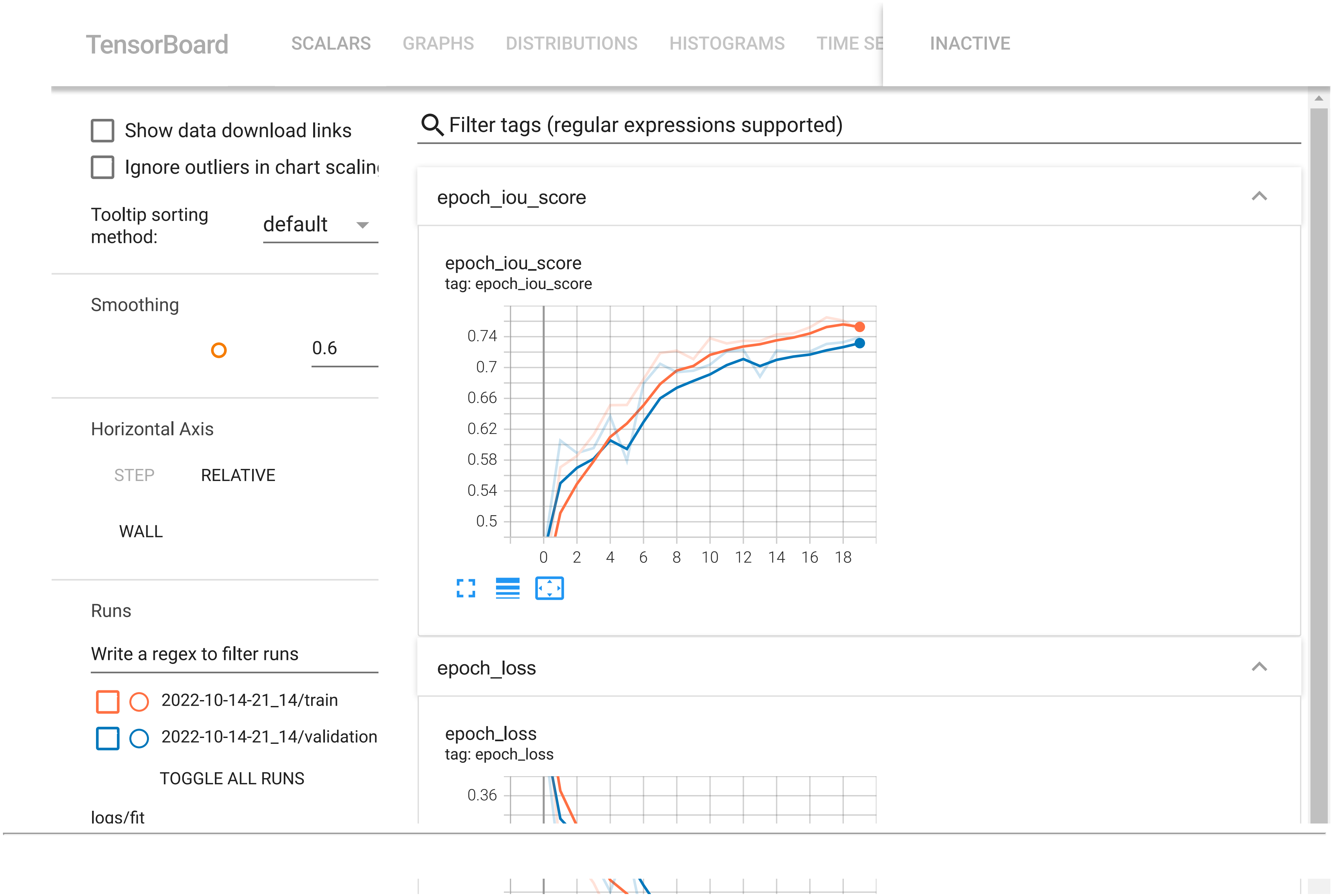
```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
```

```
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```






```
# loading tensorboard
%tensorboard --logdir logs/fit
```


TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
TIME SE
INACTIVE
Show data download links
Ignore outliers in chart scalin
Tooltip sorting method:
default
Smoothing
0.6
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
2022-10-14-21_14/train
2022-10-14-21_14/validation
TOGGLE ALL RUNS
logs/fit
Filter tags (regular expressions supported)
epoch_iou_score
epoch_iou_score
tag: epoch_iou_score
0.74
0.7
0.66
0.62
0.58
0.54
0.5
0
2
4
6
8
10
12
14
16
18
epoch_loss
epoch_loss
tag: epoch_loss
0.36

5.5 senet154_unet

```
# loading unet model with backbone - resnet34

model = sm.Unet('senet154', encoder_weights="imagenet", classes=1,
                activation='sigmoid',encoder_freeze=True, input_shape=(256, 256,3))
```

```
Downloading data from https://github.com/qubvel/classification_models/releases/download/0.0.1/senet154_imagenet_1000_no_top.h5
466635528/466635528 [==============================] - 86s 0us/step
```

```
callback_list = create_callback_lists(name = "senet154_unet")
optim = tf.keras.optimizers.Adam()
loss =  DiceLoss()#sm.losses.bce_jaccard_loss

model.compile(optim, loss, metrics=[iou_score])
```

```
history = model.fit(train_data_loader, steps_per_epoch=192, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 48,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

```
Epoch 1/20
192/192 [==============================] - ETA: 0s - loss: 0.4570 - iou_score: 0.4218
Epoch 1: val_iou_score improved from -inf to 0.27342, saving model to best_model_with_senet154_unet.hdf5
192/192 [==============================] - 599s 2s/step - loss: 0.4570 - iou_score: 0.4218 - val_loss: 0.5793 - val_iou_score:
Epoch 2/20
192/192 [==============================] - ETA: 0s - loss: 0.1984 - iou_score: 0.6774
Epoch 2: val_iou_score improved from 0.27342 to 0.65832, saving model to best_model_with_senet154_unet.hdf5
192/192 [==============================] - 391s 2s/step - loss: 0.1984 - iou_score: 0.6774 - val_loss: 0.2142 - val_iou_score:
Epoch 3/20
192/192 [==============================] - ETA: 0s - loss: 0.1808 - iou_score: 0.7035
Epoch 3: val_iou_score did not improve from 0.65832
192/192 [==============================] - 405s 2s/step - loss: 0.1808 - iou_score: 0.7035 - val_loss: 0.2400 - val_iou_score:
Epoch 4/20
192/192 [==============================] - ETA: 0s - loss: 0.1526 - iou_score: 0.7396
```

```
Epoch 4: val_iou_score improved from 0.65832 to 0.75427, saving model to best_model_with_senet154_unet.hdf5
192/192 [==============================] - 403s 2s/step - loss: 0.1526 - iou_score: 0.7396 - val_loss: 0.1448 - val_iou_score:
Epoch 5/20
192/192 [==============================] - ETA: 0s - loss: 0.1305 - iou_score: 0.7735
Epoch 5: val_iou_score did not improve from 0.75427
192/192 [==============================] - 382s 2s/step - loss: 0.1305 - iou_score: 0.7735 - val_loss: 0.1506 - val_iou_score:
Epoch 6/20
192/192 [==============================] - ETA: 0s - loss: 0.1303 - iou_score: 0.7740
Epoch 6: val_iou_score improved from 0.75427 to 0.78345, saving model to best_model_with_senet154_unet.hdf5
192/192 [==============================] - 401s 2s/step - loss: 0.1303 - iou_score: 0.7740 - val_loss: 0.1249 - val_iou_score:
Epoch 7/20
  8/192 [>.............................] - ETA: 5:16 - loss: 0.1453 - iou_score: 0.7569
```

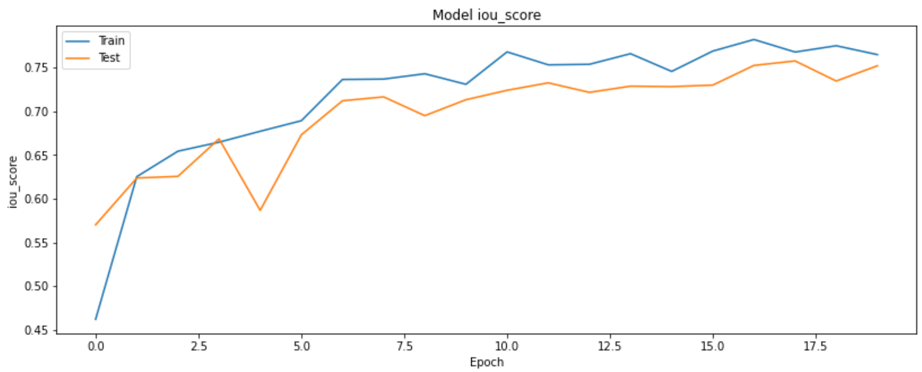
```
history = model.fit(train_data_loader, steps_per_epoch=384, epochs=20,
                    validation_data = val_data_loader,  validation_steps = 96,
                    initial_epoch = 0, callbacks = callback_list,use_multiprocessing = True )#callbacks = callback_list,
```

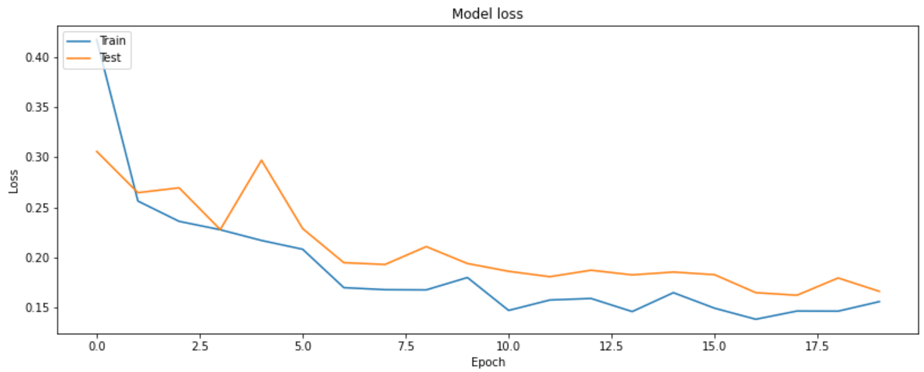
```
Epoch 1/20
384/384 [==============================] - ETA: 0s - loss: 0.4173 - iou_score: 0.4621
Epoch 1: val_iou_score improved from -inf to 0.57017, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 636s 1s/step - loss: 0.4173 - iou_score: 0.4621 - val_loss: 0.3057 - val_iou_score
Epoch 2/20
384/384 [==============================] - ETA: 0s - loss: 0.2563 - iou_score: 0.6252
Epoch 2: val_iou_score improved from 0.57017 to 0.62372, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 439s 1s/step - loss: 0.2563 - iou_score: 0.6252 - val_loss: 0.2646 - val_iou_score
Epoch 3/20
384/384 [==============================] - ETA: 0s - loss: 0.2360 - iou_score: 0.6544
Epoch 3: val_iou_score improved from 0.62372 to 0.62558, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 440s 1s/step - loss: 0.2360 - iou_score: 0.6544 - val_loss: 0.2695 - val_iou_score
Epoch 4/20
384/384 [==============================] - ETA: 0s - loss: 0.2277 - iou_score: 0.6647
Epoch 4: val_iou_score improved from 0.62558 to 0.66856, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 439s 1s/step - loss: 0.2277 - iou_score: 0.6647 - val_loss: 0.2278 - val_iou_score
Epoch 5/20
384/384 [==============================] - ETA: 0s - loss: 0.2170 - iou_score: 0.6772
Epoch 5: val_iou_score did not improve from 0.66856
384/384 [==============================] - 433s 1s/step - loss: 0.2170 - iou_score: 0.6772 - val_loss: 0.2970 - val_iou_score
Epoch 6/20
```

```
384/384 [==============================] - ETA: 0s - loss: 0.2082 - iou_score: 0.6893
Epoch 6: val_iou_score improved from 0.66856 to 0.67313, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 441s 1s/step - loss: 0.2082 - iou_score: 0.6893 - val_loss: 0.2288 - val_iou_score
Epoch 7/20
384/384 [==============================] - ETA: 0s - loss: 0.1699 - iou_score: 0.7364
Epoch 7: val_iou_score improved from 0.67313 to 0.71207, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 441s 1s/step - loss: 0.1699 - iou_score: 0.7364 - val_loss: 0.1948 - val_iou_score
Epoch 8/20
384/384 [==============================] - ETA: 0s - loss: 0.1679 - iou_score: 0.7369
Epoch 8: val_iou_score improved from 0.71207 to 0.71651, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 440s 1s/step - loss: 0.1679 - iou_score: 0.7369 - val_loss: 0.1931 - val_iou_score
Epoch 9/20
384/384 [==============================] - ETA: 0s - loss: 0.1677 - iou_score: 0.7430
Epoch 9: val_iou_score did not improve from 0.71651
384/384 [==============================] - 429s 1s/step - loss: 0.1677 - iou_score: 0.7430 - val_loss: 0.2109 - val_iou_score
Epoch 10/20
384/384 [==============================] - ETA: 0s - loss: 0.1800 - iou_score: 0.7309
Epoch 10: val_iou_score did not improve from 0.71651
384/384 [==============================] - 429s 1s/step - loss: 0.1800 - iou_score: 0.7309 - val_loss: 0.1940 - val_iou_score
Epoch 11/20
384/384 [==============================] - ETA: 0s - loss: 0.1473 - iou_score: 0.7680
Epoch 11: val_iou_score improved from 0.71651 to 0.72404, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 439s 1s/step - loss: 0.1473 - iou_score: 0.7680 - val_loss: 0.1862 - val_iou_score
Epoch 12/20
384/384 [==============================] - ETA: 0s - loss: 0.1576 - iou_score: 0.7532
Epoch 12: val_iou_score improved from 0.72404 to 0.73258, saving model to best_model_with_senet154_unet.hdf5
384/384 [==============================] - 438s 1s/step - loss: 0.1576 - iou_score: 0.7532 - val_loss: 0.1809 - val_iou_score
Epoch 13/20
384/384 [==============================] - ETA: 0s - loss: 0.1592 - iou_score: 0.7538
Epoch 13: val_iou_score did not improve from 0.73258
384/384 [==============================] - 430s 1s/step - loss: 0.1592 - iou_score: 0.7538 - val_loss: 0.1873 - val_iou_score
Epoch 14/20
384/384 [==============================] - ETA: 0s - loss: 0.1461 - iou_score: 0.7659
Epoch 14: val_iou_score did not improve from 0.73258
384/384 [==============================] - 433s 1s/step - loss: 0.1461 - iou_score: 0.7659 - val_loss: 0.1827 - val_iou_score
Epoch 15/20
```

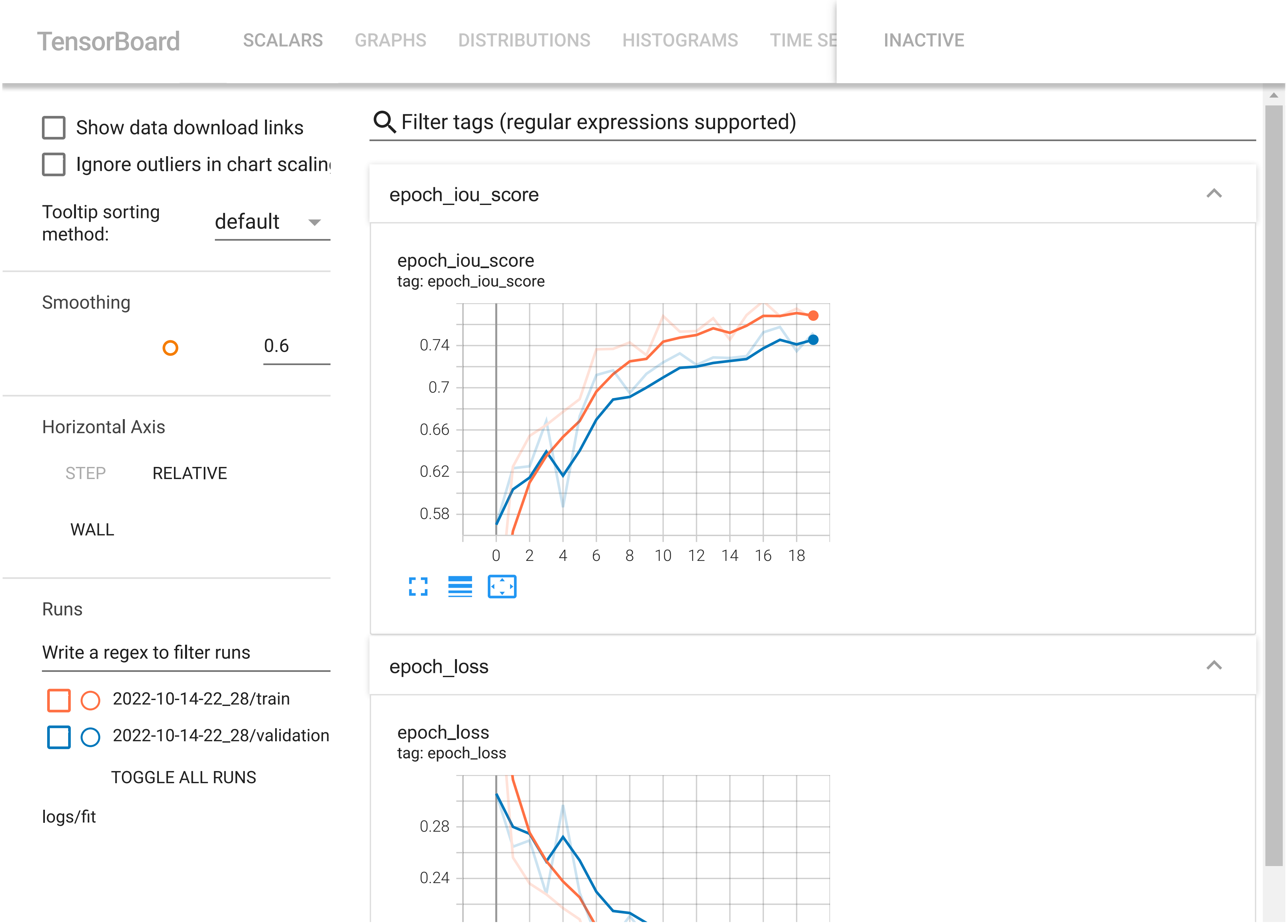
```
# Plot training & validation iou_score values
plt.figure(figsize=(30, 5))
```

```
plt.subplot(121)
plt.plot(history.history['iou_score'])
plt.plot(history.history['val_iou_score'])
plt.title('Model iou_score')
plt.ylabel('iou_score')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')

# Plot training & validation loss values
plt.subplot(122)
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```






```
# loading tensorboard
%tensorboard --logdir logs/fit
```


TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
TIME SE
INACTIVE
Show data download links
Ignore outliers in chart scalin
Tooltip sorting method:
default
Smoothing
0.6
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
2022-10-14-22_28/train
2022-10-14-22_28/validation
TOGGLE ALL RUNS
logs/fit
Filter tags (regular expressions supported)
epoch_iou_score
epoch_iou_score
tag: epoch_iou_score
0.74
0.7
0.66
0.62
0.58
0
2
4
6
8
10
12
14
16
18
epoch_loss
epoch_loss
tag: epoch_loss
0.28
0.24

performance table

```
from prettytable import PrettyTable
x = PrettyTable()
x.field_names = ["Model","Backbon","loss", "val loss", "IOU score","val_IOU score", "epoch"]
x.add_rows(
    [
        ["UNET", "resnet_34", .1951, .2221, .70, .67,20],
        ["UNET", "inceptionv3", .1657, .2066, .74, .69, 20],
        ["UNET", "effiecientnetb1", .1654, .1827, .74, .72, 20],
        ["UNET", "efficientnetb4", .1678, .1760, .74, .73, 20],
        ["UNET", "senet154",.1466, .1624, .76, .75, 20],

    ]
)
print(x)
```

```
+-------+-----------------+--------+----------+-----------+---------------+-------+
| Model |     Backbon     |  loss  | val loss | IOU score | val_IOU score | epoch |
+-------+-----------------+--------+----------+-----------+---------------+-------+
|  UNET |    resnet_34    | 0.1951 |  0.2221  |    0.7    |      0.67     |   20  |
|  UNET |   inceptionv3   | 0.1657 |  0.2066  |    0.74   |      0.69     |   20  |
|  UNET | effiecientnetb1 | 0.1654 |  0.1827  |    0.74   |      0.72     |   20  |
|  UNET |  efficientnetb4 | 0.1678 |  0.176   |    0.74   |      0.73     |   20  |
|  UNET |     senet154    | 0.1466 |  0.1624  |    0.76   |      0.75     |   20  |
+-------+-----------------+--------+----------+-----------+---------------+-------+
```

Note: will be continue in part2