CNN

import tensorflow as tf

from tensorflow import keras

from tensorflow.keras import layers

from tensorflow.keras.layers import Dense,Conv2D,MaxPooling2D,Flatten

from tensorflow.keras.models import Sequential

import cv2

import matplotlib.pyplot as plt

train = keras.utils.image\_dataset\_from\_directory(

    directory = '/content/train',

    labels = 'inferred',

    label\_mode = 'int',

    batch\_size = 32,

    image\_size = (256,256)

)

test = keras.utils.image\_dataset\_from\_directory(

    directory = '/content/validation',

    labels = 'inferred',

    label\_mode = 'int',

    batch\_size = 32,

    image\_size = (256,256)

)

def normalize(image,label):

    image = tf.cast(image/255. ,tf.float32)

    return image,label

train = train.map(normalize)

test = test.map(normalize)

model = Sequential([

    Conv2D(32, (3, 3), activation='relu', input\_shape=(256, 256, 3)),

    MaxPooling2D((2, 2)),

    Conv2D(32, (3, 3), activation='relu'),

    MaxPooling2D((2, 2)),

    Flatten(),

    Dense(64, activation='relu'),

    Dense(1, activation='sigmoid')

])

model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])

cn = model.fit(train, validation\_data=test, epochs=5)

test\_image = cv2.imread('/content/cheetah\_015\_val\_resized.jpg')

test\_image = cv2.resize(test\_image, (256, 256))

plt.imshow(test\_image)

test\_image = test\_image.reshape(1,256, 256,3)

prediction = model.predict(test\_image)

if prediction < 0.5:

  print('Cheetah')

else:

  print('Hyena')