import sys

import os

sys.path.append("/Users/manfrednde/Library/Python/3.9/lib/python/site-packages")

import io

import datetime

import pandas as pd

from PIL import Image

import requests

import io

import glob, os, sys, time, uuid

from matplotlib.pyplot import imshow

import matplotlib.pyplot as plt

from urllib.parse import urlparse

from io import BytesIO

from PIL import Image, ImageDraw

from video\_indexer import VideoIndexer

from azure.cognitiveservices.vision.face import FaceClient

from azure.cognitiveservices.vision.face.models import TrainingStatusType

from msrest.authentication import CognitiveServicesCredentials

CONFIG = {

'SUBSCRIPTION\_KEY': '621d1daaa9ff4041bc484d4f7b9d5d10',

'LOCATION': 'trial',

'ACCOUNT\_ID': '6723d089-da1a-41ab-ae6a-7074c23d084b'

}

video\_analysis = VideoIndexer(

vi\_subscription\_key=CONFIG['SUBSCRIPTION\_KEY'],

vi\_location=CONFIG['LOCATION'],

vi\_account\_id=CONFIG['ACCOUNT\_ID']

)

video\_analysis.check\_access\_token()

video\_id = '30fa204ae0'

video\_analysis.get\_video\_info(video\_id)

info = video\_analysis.get\_video\_info(video\_id, video\_language='English')

if len(info['videos'][0]['insights']['faces'][0]['thumbnails']):

print("We found {} faces in this video.".format(str(len(info['videos'][0]['insights']['faces'][0]['thumbnails']))))

info['videos'][0]['insights']['faces'][0]['thumbnails']

images = []

img\_raw = []

img\_strs = []

for each\_thumb in info['videos'][0]['insights']['faces'][0]['thumbnails']:

if 'fileName' in each\_thumb and 'id' in each\_thumb:

file\_name = each\_thumb['fileName']

thumb\_id = each\_thumb['id']

img\_code = video\_analysis.get\_thumbnail\_from\_video\_indexer(video\_id, thumb\_id)

img\_strs.append(img\_code)

img\_stream = io.BytesIO(img\_code)

img\_raw.append(img\_stream)

img = Image.open(img\_stream)

images.append(img)

for img in images:

print(img.info)

plt.figure()

plt.imshow(img)

i = 1

for img in images:

print(type(img))

img.save('human-face' + str(i) + '.jpg')

i= i+ 1

thumbnail\_id = "40294a85-eb72-4085-84c4-081e094ac866"

img\_code = video\_analysis.get\_thumbnail\_from\_video\_indexer(video\_id, thumbnail\_id)

print(img\_code)

img\_code = video\_analysis.get\_thumbnail\_from\_video\_indexer(video\_id, thumbnail\_id)

img\_stream = io.BytesIO(img\_code)

img = Image.open(img\_stream)

imshow(img)

keyframes = []

for shot in info["videos"][0]["insights"]["shots"]:

for keyframe in shot["keyFrames"]:

keyframes.append(keyframe["instances"][0]['thumbnailId'])

for keyframe in keyframes:

img\_str = video\_analysis.get\_thumbnail\_from\_video\_indexer(video\_id, keyframe)

info['summarizedInsights']['sentiments']

info['summarizedInsights']['emotions']