https://www.kaggle.com/saidakbarp/face-recognition-part-1 (https://www.kaggle.com/saidakbarp/face-recognition-part-1)

In [9]:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
#Visiulazation
import matplotlib.pyplot as plt
#image processing
import cv2
#extracting zippped file
import tarfile
#systems
import os
print(os.listdir("/home/hduser/jupyter"))
```

```
['winequality-red.csv', 'winequality-red_StratifieldKFold.csv', 'dt_0.bin', 'dt
_1.bin', 'dt_3.bin', 'Face_Detection_with_OpenCV', 'Face_recognition_part_1.ipy
nb', 'jupyter.csv', 'test.py', 'remte_server_testing.ipynb', '.ipynb_checkpoint
s', 'testfile.csv', 'Untitled.ipynb', 'Categorical_Feature_Encoding_Challenge',
'Face_Comparison_deepface', 'dob_checker_file.csv', 'kaggle-Comprehensive data
exploration with Python.ipynb', 'ageron_aurelien_math_linear_algebra.ipynb', 'a
geron_aurelien_numpy.ipynb', 'ageron_aurelien_pandas.ipynb', 'cdp_demo_file_10
k.ipynb', 'data_visualization.ipynb', 'DOB_CHECKER.ipynb', 'IBM_DB2.ipynb', 'IB
M_DB2_data_analysis.ipynb', 'IBM_DB2_final_initial.ipynb', 'IBM_DB2_final_new_l
atest-Copy1.ipynb', 'IBM_DB2_final_new_latest-Copy2.ipynb', 'IBM_DB2_final_new_
latest.ipynb', 'IBM_DB2_READ_WRITE.ipynb', 'IBM_DB2_testing.ipynb', 'myfirst no
de book.ipynb', 'posilitics.ipynb', 'practice_stackoverflow.ipynb', 'Python dat
a visualizations on the Iris dataset.ipynb', 'stackoverflow_numpy.ipynb', 'test
data_posilytics.ipynb', 'testdata_posilytics_ckecking-Copy1.ipynb', 'testdata_p
osilytics_ckecking-Copy2.ipynb', 'testdata_posilytics_ckecking-Copy3.ipynb', 't
estdata_posilytics_ckecking.ipynb', 'veera_initial_modified code.ipynb', 'index
-of-my-kaggle-notebooks.ipynb', 'name tokenization_my3.ipynb', 'posilitics2.ipy
nb', 'Session 1.0 - Python-Installation (1).ipynb', 'Session 1.0 - Python-Insta
llation.ipynb', 'IBM_DB2_final_new_latest_WITH_TIMESTAMP.ipynb', 'test', 'IBM_D
B2_final_new_latest_WITH_TIMESTAMP_10l_testing-Copy1.ipynb', 'IBM_DB2_1lakh.ipy
nb', 'Approaching_Almost_Any_Machine_Learning_Problem_4.ipynb', 'Face_Compariso
n_deepface.ipynb', 'testdata_posilytics.csv', 'concrete_strength_ML_Prediction.
ipynb', 'concrete_data.csv', 'dt_4.bin', 'Face_Detection_with_OpenCV.ipynb', 'U
ntitled1.ipynb', 'IBM_DB2_final_new_latest_WITH_TIMESTAMP_10l_testing.ipynb',
'dt_2.bin', 'Approaching_Almost_Any_Machine_Learning_Problem_5.ipynb', 'posilit
cs_10L_1.txt', 'Approaching_Almost_Any_Machine_Learning_Problem_1.ipynb', 'Appr
oaching_Almost_Any_Machine_Learning_Problem_2.ipynb', 'winequality-red_n_folds.
csv', 'Approaching_Almost_Any_Machine_Learning_Problem_3.ipynb']
```

In [10]:

```
#example
imgg="/home/hduser/jupyter/Face_Detection_with_OpenCV/b97ea33b5842c7894b804923c6c
imgg2 = "/home/hduser/jupyter/Face_Comparison_deepface/1.jpg"
celeb=cv2.imread(imgg2)
```

In [11]:
```
def show_image(image):
    plt.figure(figsize=(8,5))
    #Before showing image, bgr color order transformed to rgb order
    plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    plt.xticks([])
    plt.yticks([])
    plt.show()
```

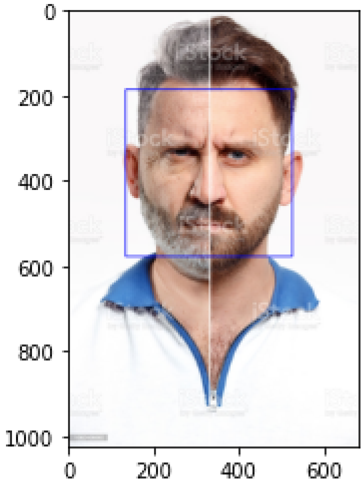
In [12]:
```
show_image(celeb)
```

In [13]:
```
# Our face detection function that uses haarcascade from OpenCV
def face_detection(img):
    face_cascade = cv2.CascadeClassifier('/home/hduser/jupyter/face-recognition/h

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray)
    print('Number of faces detected:', len(faces))

    for (x,y,w,h) in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
        #img = img[y:y+h, x:x+w] # for cropping
    cv_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    return cv_rgb
```

In [14]:

```
#imgg2=cv2.imread("/kaggle/input/photos/ben.jpg")
a=face_detection(celeb)
plt.imshow(a)
plt.show()

# as shown below, the library is not detecting this particular face angle of Ben
```

```
Number of faces detected: 1
```



In [15]:
```
plt.figure(figsize=(15,18))
img=cv2.imread("/home/hduser/jupyter/Face_Detection_with_OpenCV/b97ea33b5842c7894
c=face_detection(img)
plt.imshow(c)
plt.show()
```

```
Number of faces detected: 9
```