

# Vyhľadávanie konkrétnej osoby pri čiastočnej oklúzii tváre

Marianna Ráchelová

**Vedúca:** RNDr. Zuzana Černeková, PhD.

# Cieľ

- Vyhľadávanie konkrétnej osoby vo videu
- Detekcia ľudských tvárí
- Analýza existujúcich riešení
- Vytvorenie databázy videozáznamov
- Návrh a implementácia metódy vyhľadania konkrétnej osoby s čiastočne zakrytou tvárou vo videu
- Vyhodnotenie výsledkov

# Databáza

- **Labeled faces in the Wild**
- Dostupné online
- <http://vis-www.cs.umass.edu/lfw/>
- 13000 obrázkov
- 1680 ľudí
- Postupne pribudne databáza videozáznamov

# Technológie

- Programovací jazyk Python
- Knižnice jazyka Python
- Tensorflow, Numpy, Cv2, Scipy...

# Vedecké články

- *Recognition and detection of occluded faces by a neural network classifier with recursive data reconstruction*, T.Kurita, M.Pic, T.Takahashi
- *Face occlusion detection using deep convolutional neural networks*, Yizhang Xia\* 4 , Bailing Zhang, Frans Coenen
- *A Survey: Face Recognition Techniques under Partial Occlusion*, Aisha Azeem, Muhammad Sharif, Mudassar Raza, and Marryam Murtaza
- *Multi-view face detection using deep convolutional neural networks*, Sachin Sudhakar Farfade, Mohammad Saberian, Li-Jia Li

# Podobné práce

- <https://github.com/majetyhk/FacialRecognition/tree/master/NN1>
- <https://github.com/AlSangam/Facenet-Real-time-face-recognition-using-deep-learning-Tensorflow>

# Prototyp

- Načítavanie videa po framoch
- Rozbeh a vyskúšanie si neurónovej siete – Facenet
- Klasifikácia tvárí použitím HOG deskriptora, PCA a SVM metód  
(klasifikácia bez použitia Neurónových sietí)

Ukážky



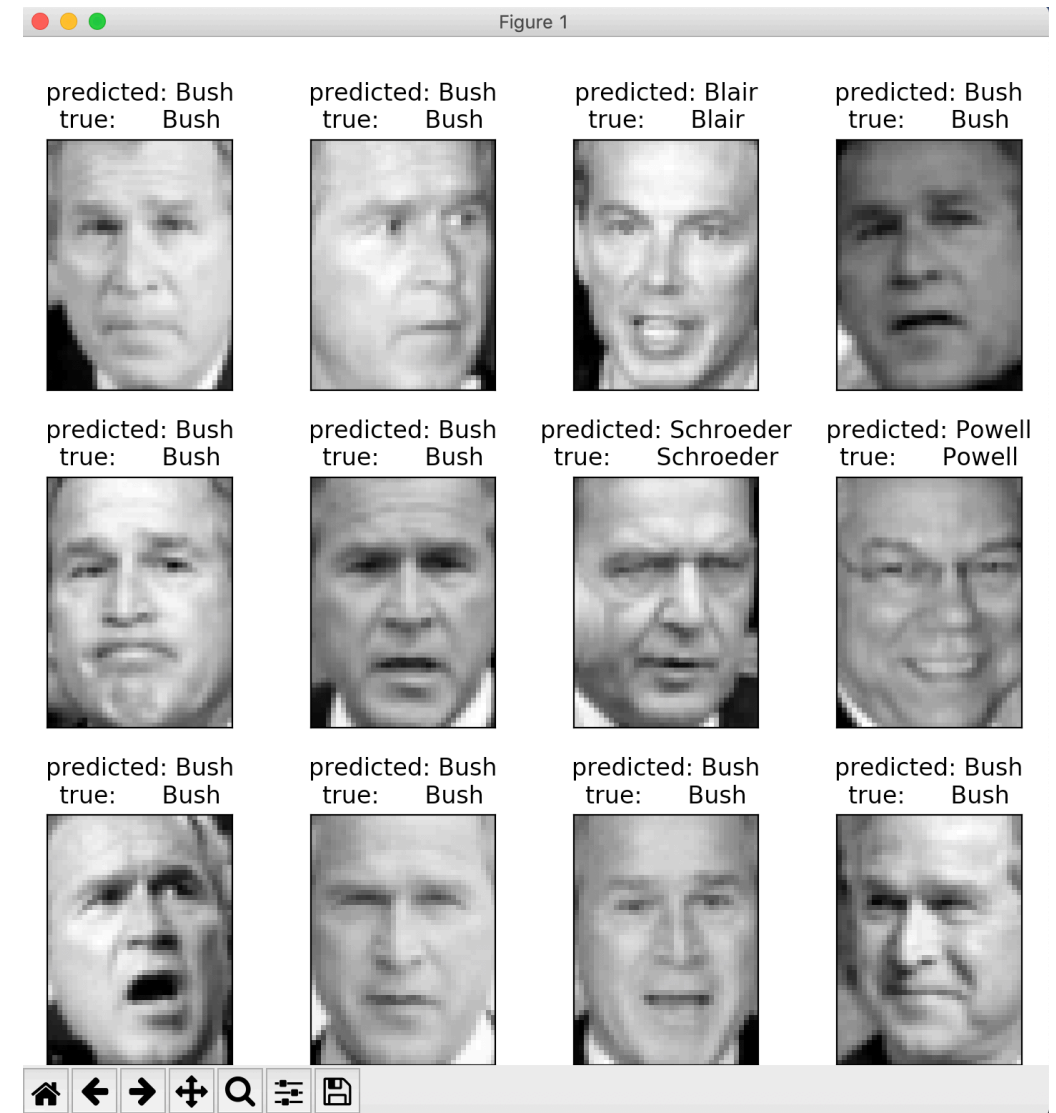
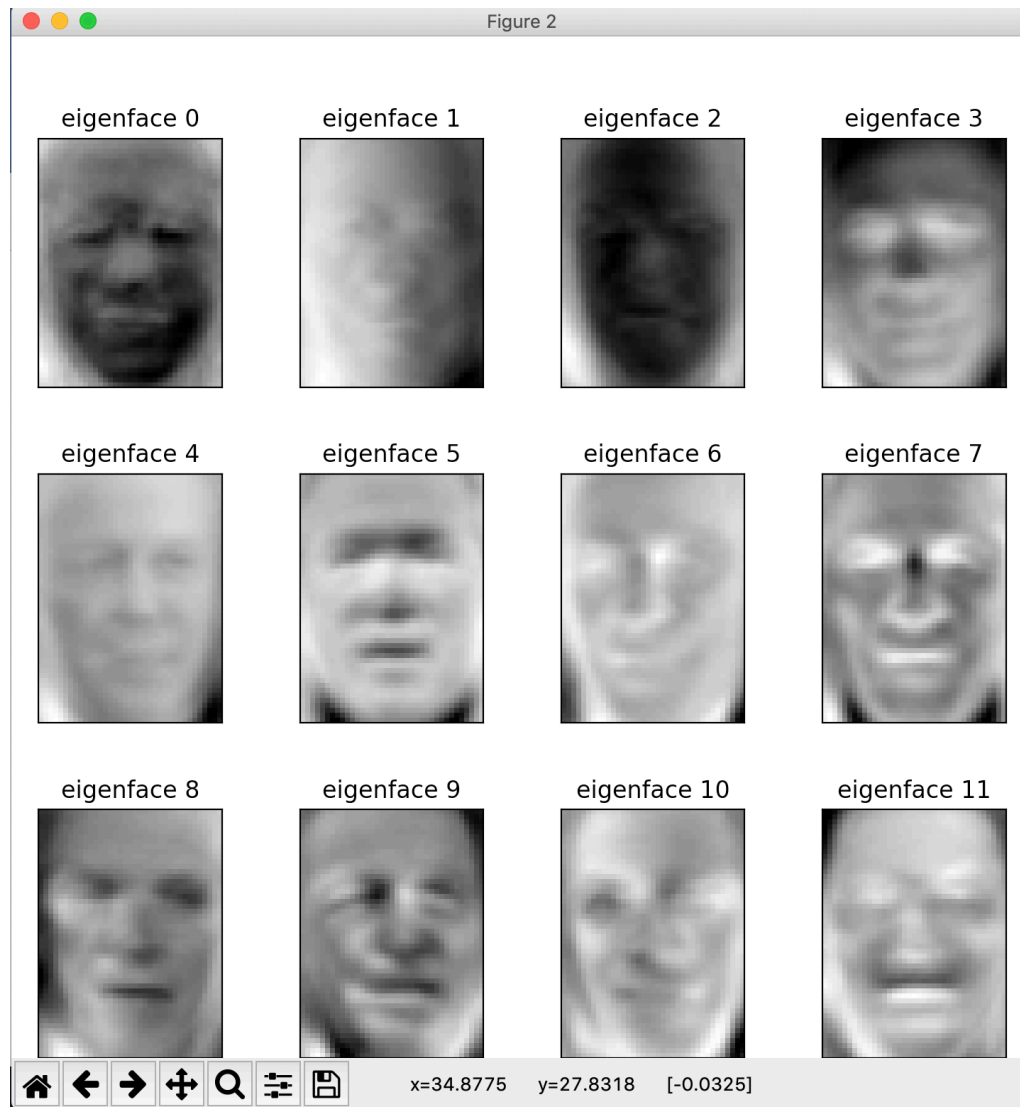


Figure 2

predicted: Albert\_Costa  
true: Albert\_Costa

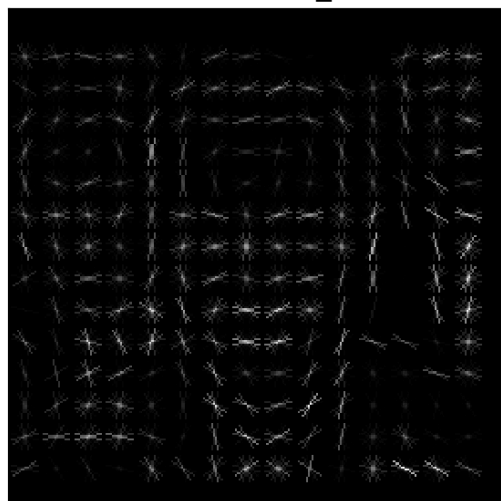


Figure 3

predicted: Allyson\_Felix  
true: Allyson\_Felix

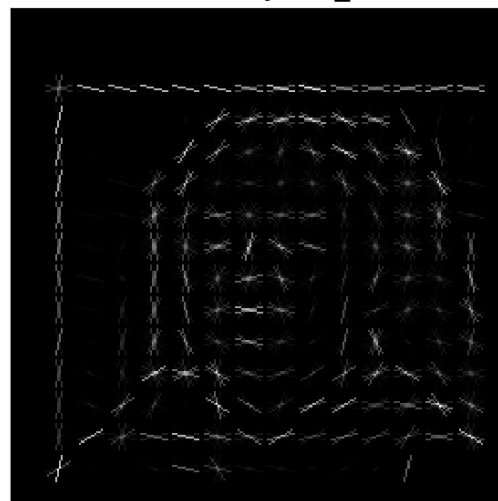
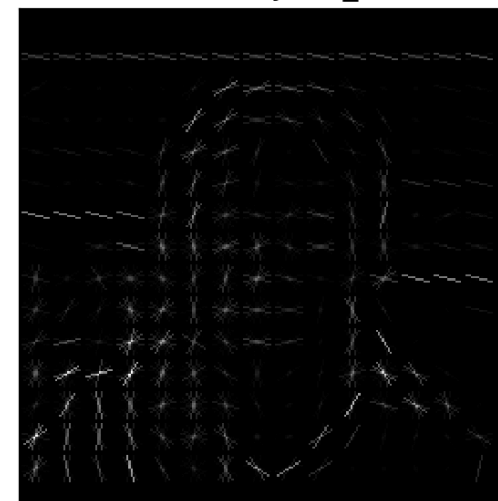


Figure 4

predicted: Ali\_Naimi  
true: Allyson\_Felix





- ▼ masterThesis ~/PycharmProjects/masterT
  - ▶ Facenet-Real-time-face-recognition-usi
  - ▶ HOG\_example
  - ▼ images
    - frame0.jpg
    - frame1.jpg
    - frame2.jpg
    - frame3.jpg
    - frame4.jpg
    - frame5.jpg
    - frame6.jpg
    - frame7.jpg
    - frame8.jpg
    - frame9.jpg
    - frame10.jpg
    - frame11.jpg
    - frame12.jpg
    - frame13.jpg
    - frame14.jpg
    - frame15.jpg
    - frame16.jpg
    - frame17.jpg
    - frame18.jpg
    - frame19.jpg
    - frame20.jpg

```
1 import cv2
2 import numpy as np
3 import os
4
5 # set video file path of input video with name and extension
6 vid = cv2.VideoCapture('DoctorStrange.mp4')
7
8
9 if not os.path.exists('images'):
10     os.makedirs('images')
11
12 #for frame identity
13 index = 0
14 while(True):
15     # Extract images
16     ret, frame = vid.read()
17     # end of frames
18     if not ret:
19         break
20     # Saves images
21     name = './images/frame' + str(index) + '.jpg'
22     print('Creating...' + name)
23     cv2.imwrite(name, frame)
24
25     # next frame
26     index += 1
27
```

Ďakujem za pozornosť