1.사진을 보고 얼굴을 찾는다

2.각각의 얼굴에 초점을 맞추고 얼굴이 틀어지거나 조명이 않좋은 상황이라도 같은 사람인 것을 인식

3. 사람들간에 얼굴의 고유한 특징을 찾아낸다.

4. 기존에 알고있는 사람들과 비교해서 이름을 결정

step.1 얼굴 찾기

Histogram of Oriented Gradients(HOG)

.1)흑백으로 사진을 변환

2) 이미지의 모든 단일 픽새을 살펴보고 이를 둘러싸고있는 픽샐을 살표봄. 해당픽샐과 이를 직접둘러싸고 있는 픽셀을 비교해서 얼마나 어두운지를 알아내는 것, 이미지가 어두워지는 방향을 나타내는 화살표를 그림, 이를 gradient로 분석하면 동일한 사람이 어두운 이미지 또는 밝은 이미지에 상관없이 같은 표현을 얻을 수 있다.

3)드러나 모든 것을 하면 너무 자세함 이미지를 16\*16으로 분해 얼마나 많은 gradient가 주요방향을 가리키는지 세어 볼것이다.

4) 훈련된 이미지로부터 추출된 HOG패턴과 가장 유사하기 보이는 부분을 이미지에서 찾음

step2 얼굴의 위치 교정, 투영

1)같은 사람이지만 서로 다른 방향을 보게 될 경우 각각의 사진을 비틀어 얼굴 특징들이 항상 표준 위치에 오게끔 해야한다.

=>face landmark estimation

‘모든 얼굴에 존재하는 68개의 랜드마크라 부르는 특정 포인트(턱의 상단,눈 바깥의 가장자리, 눈썸 안쪽의 가장자리등등)을 찾아내는 것이다. 기계학습 알고리즘을 훈련시켜 어떤얼굴이든지 68개의 포인트를 찾을 수 있도록 함

2)이를 rotation, scale, shear를 통하여 가운데 올 수 있도록 만든다.

step3 얼굴 인코딩

각각의 얼굴에서 몇가지 기본적인 측정값을 추출해야 한다. 그 다음 모르는 얼굴을 이 방법을 가지고 측정하고 비교하여 가장 가까운 값을 가지는 얼굴을 찾아낸다.

컴퓨터 스스로가 수집할 측정값을 스스로 파악하는 것 => deep covolutional Neural Network

훈련과정

3개의 얼굴 이미지를 차례대로 확인하는 것이다.

1. 훈련용 아는 사람이 얼굴 사진 적재

2.동일한 아는 사람의 다른 사진 적재

3.전혀 다른 사람의 사진 적재

판단: 1,2 측정값은 서로 가깝게하고 2,3의 측정값은 멀어지도록 신경망을 조정(엄청 많이 해야됨)

=🡺임베딩

원시 데이터를 컴퓨터가 생성한 숫자값의 목록으로 축소

신경망을 만들기 위해서는 엄청난 데이터와 시간이 필요하다. 그렇기에 몇가지 훈련된 신경망을 가지고 임베딩을 실시 (OpenFace)

step4 인코딩에서 사람의 이름 찾기

1.테스트 이미지에서 가장 근접한 측정값을 갖고 있는 사람을 우리가 가지고있는 사람에 대한 데이터베이스에서 찾아내는 것이다. 기본적인 기계학습 분류알고리즘을 사용하여 수행 가능하다. 간단한 선형SVM분류기(classifier)를 사용

2.사진에 대한 임베딩을 가지고 분류기를 훈련시킨다.