

OpenCV와 얼굴 검출

OpenCV에서 지원하는 얼굴 검출 기법

- Haar Cascade

- 2001년 Viola & Jones에 의해 제안된 방법

- Haar-like 특징과 Adaboost 알고리즘, Cascade 구조를 사용하여 빠르고 정확한 얼굴 검출을 수행

- DNN(Deep Neural Net) 방법

- OpenCV 3.3.1부터 DNN 모듈을 사용한 얼굴 검출을 기본 예제로 제공(2017년)

- Resnet-10과 SSD를 기반으로 학습된 얼굴 검출 네트워크 사용

DNN - 기존의 CascadeClassifier보다 대체로 더 좋은 성능을 보임(속도, 정확도 측면에서)

Haar는 정면 얼굴, 측면 얼굴, 가려짐이 있어도 검출 불가하지만 DNN은 잘 찾음

무조건 DNN이 더 낫다고 한다.

DL : End-to-End 알아서 학습하는 것이 가능하다.

ML : 빨간 것은 사과 .. 이런 식으로 어떤 특징을 사용할 것인지 등을 직접 설정해줘야 한다.

Input -> Sum($w \cdot x + b$) -> Sigmoid -> Output

OpenCV DNN

OpenCV에서 제공하는 Deep Learning module

네트워크 불러오기 : 딥러닝 구조를 생성

네트워크 불러오기

```
cv2.dnn.readNet(model, config=None, framework=None) -> retval
```

- model: 훈련된 가중치를 저장하고 있는 이진 파일 이름
- config: 네트워크 구성을 저장하고 있는 텍스트 파일 이름
- framework: 명시적인 딥러닝 프레임워크 이름
- retval: `cv2.dnn_Net` 클래스 객체

딥러닝 프레임워크	model 파일 확장자	config 파일 확장자	framework 문자열
카페	*.caffemodel	*.prototxt	"caffe"
텐서플로우	*.pb	*.pbtxt	"tensorflow"
토치	*.t7 또는 *.net		"torch"
다크넷	*.weights	*.cfg	"darknet"
DLDLT	*.bin	*.xml	"dldt"
ONNX	*.onnx		"onnx"

네트워크 입력 blob(blob) 만들기

blob : 다차원 행렬, tensor와 유사한 개념이라고 생각하면 된다.

네트워크 입력 blob(blob) 만들기

```
cv2.dnn.blobFromImage(image, scalefactor=None, size=None,
                       mean=None, swapRB=None, crop=None,
                       ddepth=None) -> retval
```

- image: 입력 영상
- scalefactor: 입력 영상 픽셀 값에 곱할 값. 기본값은 1.
- size: 출력 영상의 크기. 기본값은 (0, 0).
- mean: 입력 영상 각 채널에서 뺄 평균 값. 기본값은 (0, 0, 0, 0).
- swapRB: R과 B 채널을 서로 바꿀 것인지를 결정하는 플래그. 기본값은 False
- crop: 크롭(crop) 수행 여부. 기본값은 False.
- ddepth: 출력 blob의 깊이. CV_32F 또는 CV_8U. 기본값은 CV_32F.
- retval: 영상으로부터 구한 blob 객체. [numpy.ndarray](#).
`shape=(N,C,H,W), dtype=float32`

네트워크 입력 설정하기

네트워크 입력 설정하기

```
cv2.dnn_Net.setInput(blob, name=None, scalefactor=None,
                     mean=None) -> None
```

- blob: blob 객체
- name: 입력 레이어 이름
- scalefactor: 추가적으로 픽셀 값에 곱할 값
- mean: 추가적으로 픽셀 값에서 뺄 평균 값

네트워크 순방향 실행 (추론)

네트워크 순방향 실행 (추론)

```
cv2.dnn_Net.forward(outputName=None) -> retval
```

- outputName: 출력 레이어 이름
- retval: 지정한 레이어의 출력 blob. 네트워크마다 다르게 결정됨.