

# 카카오톡 포즈 API

파이썬

## 제공 기능

- 이미지 분석하기: 하나의 이미지 내에서 사람들을 탐색하고, 사람 별 키 포인트를 추출하여 자세를 분석합니다.
- 영상 분석하기: 요청한 영상의 프레임 별로 사람들을 탐색하고 키포인트를 추출합니다.
- 영상 분석 결과 확인하기: 영상 분석하기 API를 통해 분석한 작업물의 처리 상태와 결과를 반환합니다.

## 기능 소개

카카오 포즈(Pose) API는 이미지나 영상을 분석해 사람의 자세를 추출하는 기능을 제공합니다. 이미지에서 사람들을 찾고 사람의 코, 눈, 귀, 어깨, 팔꿈치, 손목, 골반, 무릎, 발목의 위치를 키 포인트(Key Points)로 추출합니다.



## 활용 예시

포즈 분석 딥러닝 기술을 이용한 포즈 API를 활용하여 이미지나 동영상 기반 서비스에서 아래와 같이 다양한 인공지능 서비스를 만들 수 있습니다.

- 요가나 스포츠에서 볼 수 있는 일반적인지 않은 고난이도 자세를 분석합니다.
- 사진이나 영상 속 인물과 동작 일치 정도를 점수화하거나 부위별 차이를 알려 자세를 교정합니다.
- 사진이나 영상 속 인물과 같은 자세의 캐릭터를 표현합니다.
- 축구 동영상 속 슈팅과 같은 특정 동작을 검색합니다.
- 특정 이상 행동을 감지하는 모니터링에 사용합니다.

이 포즈 API는 사용자의 실시간 관절 움직임을 추출하여 정교한 분석 후 올바른 운동 자세를 추천해 주는 카카오 VX 스마트홈트에서 활용되고 있습니다.



```

import cv2
import numpy as np
from matplotlib import pyplot as plt
from pycocotools.coco import COCO
from requests import Session

APP_KEY = 'APP키 입력하는곳'
session = Session()
session.headers.update({'Authorization': 'KakaoAK ' + APP_KEY})

def inference(filename):
    with open(filename, 'rb') as f:
        response = session.post('https://cv-api.kakaobrain.com/pose', files={'file': f})
        response = session.post('https://cv-api.kakaobrain.com/pose', data={'image_url': 'asd.jpg'})
        response.raise_for_status()
        return response.json()

def visualize(filename, annotations, threshold=0.2):
    # 낮은 신뢰도를 가진 keypoints들은 무시
    for annotation in annotations:
        keypoints = np.asarray(annotation['keypoints']).reshape(-1, 3)
        low_confidence = keypoints[:, -1] < threshold
        keypoints[low_confidence, :] = [0, 0, 0]
        annotation['keypoints'] = keypoints.reshape(-1).tolist()

    # COCO API를 활용한 시각화
    image = cv2.cvtColor(cv2.imread(filename, cv2.IMREAD_COLOR), cv2.COLOR_BGR2RGB)
    plt.imshow(image)
    plt.axis('off')
    coco = COCO()
    coco.dataset = {
        "categories": [
            {
                "supercategory": "person",
                "id": 1,
                "name": "person",
                "keypoints": ["nose", "left_eye", "right_eye", "left_ear", "right_ear", "left_shoulder",
                             "right_shoulder", "left_elbow", "right_elbow", "left_wrist", "right_wrist", "left_hip",
                             "right_hip", "left_knee", "right_knee", "left_ankle", "right_ankle"],
                "skeleton": [[1, 2], [1, 3], [2, 3], [2, 4], [3, 5], [4, 6], [5, 7], [6, 7], [6, 8], [6, 12], [7, 9],
                             [7, 13], [8, 10], [9, 11], [12, 13], [14, 12], [15, 13], [16, 14], [17, 15]]
            }
        ]
    }
    coco.createIndex()
    coco.showAnns(annotations)
    plt.show()

IMAGE_FILE_PATH = '이미지 경로 입력하는곳'
result = inference(IMAGE_FILE_PATH)
visualize(IMAGE_FILE_PATH, result)

```

파일의 요청방식은  
이미지의 url경로  
path 경로 두가지

이미지의 크기가  
형식에 맞지 않는 크기면 에러가  
발생한다.

인공지능은 17개의 사람 신체  
부위를 찾아냄

배열은 1.2 코로 시작해서  
17.15 왼쪽 오른쪽 발목으로 끝

이미지에 사람이 여러 명 있을  
경우, 검출된 사람들의 키 포인트  
정보 값은 Person 객체의  
목록으로 반환됨



