

Python 데이터 분석과 이미지 처리

나동빈

OpenCV 이미지 연산

이미지 크기 및 픽셀 확인

```
import cv2

image = cv2.imread('image_basic.png')
# 픽셀 수 및 이미지 크기 확인
print(image.shape)
print(image.size)

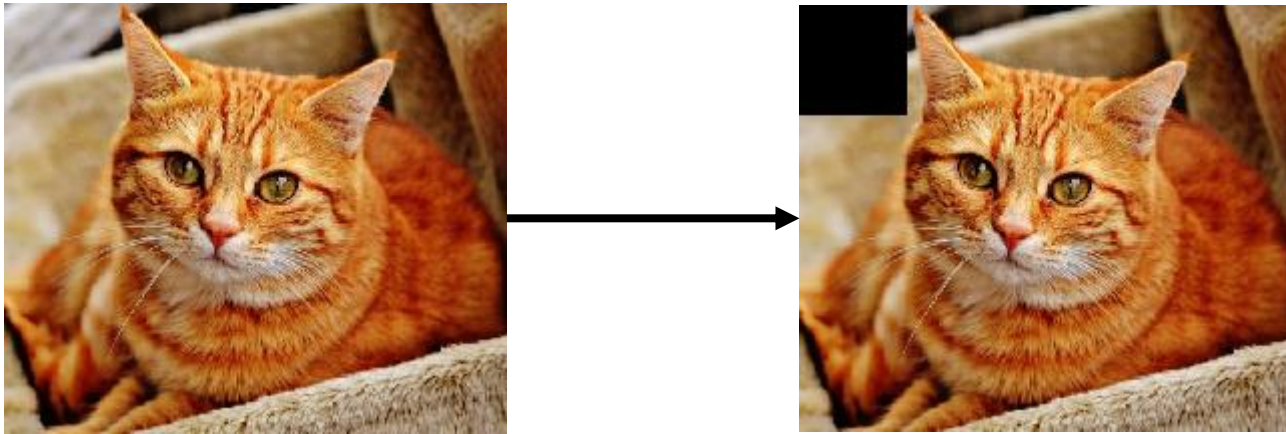
# 이미지 Numpy 객체의 특정 픽셀을 가리킵니다.
px = image[100, 100]

# B, G, R 순서로 출력됩니다.
# Gray Scale: B, G, R로 구분되지 않습니다.
print(px)

# R 값만 출력하기
print(px[2])
```

OpenCV 이미지 연산

특정 범위 픽셀 변경



OpenCV 이미지 연산

특정 범위 픽셀 변경

```
import cv2
import time

image = cv2.imread('image_basic.png')

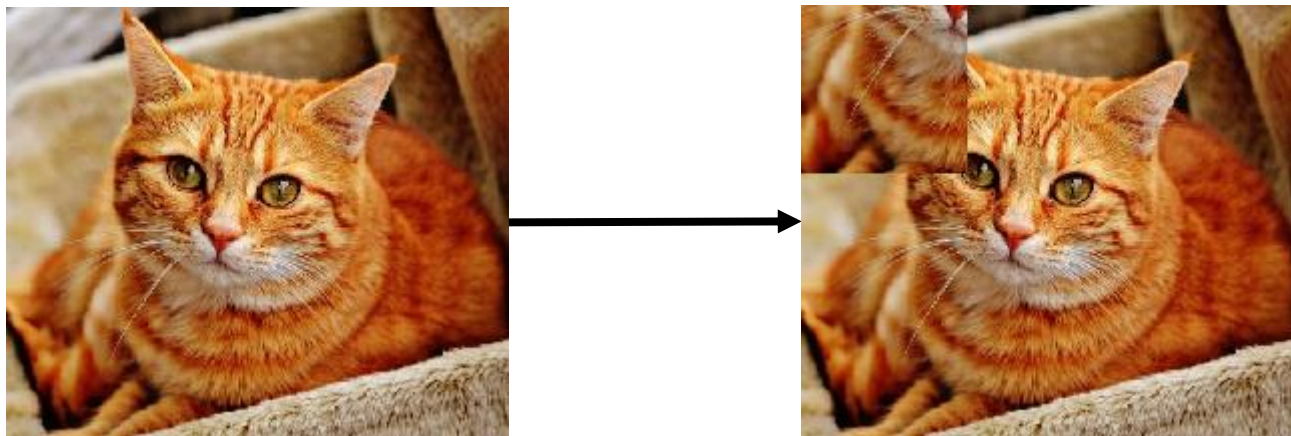
start_time = time.time()
for i in range(0, 100):
    for j in range(0, 100):
        image[i, j] = [255, 255, 255]
print("--- %s seconds ---" % (time.time() - start_time))

start_time = time.time()
image[0:100, 0:100] = [0, 0, 0]
print("--- %s seconds ---" % (time.time() - start_time))

cv2.imshow('Image', image)
cv2.waitKey(0)
```

OpenCV 이미지 연산

ROI 추출 및 복사



OpenCV 이미지 연산

ROI 추출 및 복사

```
import cv2

image = cv2.imread('image_basic.png')

# Numpy Slicing: ROI 처리 가능
logo = image[20:150, 70:200]
cv2.imshow('Image', logo)
cv2.waitKey(0)

# ROI 단위로 이미지 복사하기
image[0:130, 0:130] = logo
cv2.imshow('Image', image)
cv2.waitKey(0)
```

OpenCV 이미지 연산

픽셀별로 색상 다루기



OpenCV 이미지 연산

픽셀별로 색상 다루기

```
import cv2

image = cv2.imread('image_basic.png')

# 특정 색상만 보여주기 (0: Blue, 1: Green, 2: Red)
cv2.imshow('Image', image[:, :, 0])
cv2.waitKey(0)

# 특정 색상만 제거하기
image[:, :, 2] = 0
cv2.imshow('Image', image)
cv2.waitKey(0)
```