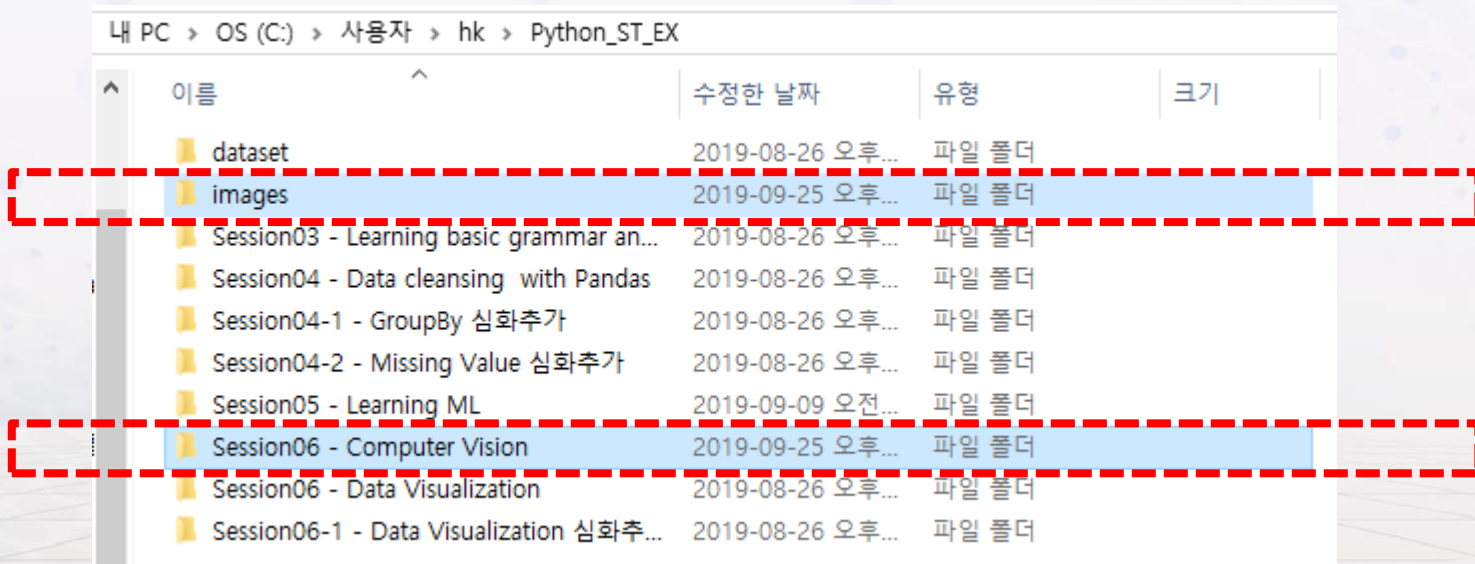


바탕화면 내 " 4. 컴퓨터 비전 모듈 " 을 다운로드 받은 후

코드는 코드 폴더 내

Images 은 dataset과 동일한 레벨에 저장해주세요.



이름	수정한 날짜	유형	크기
dataset	2019-08-26 오후...	파일 폴더	
images	2019-09-25 오후...	파일 폴더	
Session03 - Learning basic grammar an...	2019-08-26 오후...	파일 폴더	
Session04 - Data cleansing with Pandas	2019-08-26 오후...	파일 폴더	
Session04-1 - GroupBy 심화추가	2019-08-26 오후...	파일 폴더	
Session04-2 - Missing Value 심화추가	2019-08-26 오후...	파일 폴더	
Session05 - Learning ML	2019-09-09 오전...	파일 폴더	
Session06 - Computer Vision	2019-09-25 오후...	파일 폴더	
Session06 - Data Visualization	2019-08-26 오후...	파일 폴더	
Session06-1 - Data Visualization 심화추...	2019-08-26 오후...	파일 폴더	

# 1. 컴퓨터 비전 적용 분야

## 활용 사례

### 감정분석 (만족도)



- 매장 내 고객 만족도
- 영화평점 판별
- 도시 거주자 행복지수

### 사물/얼굴 인식



- 사원관리 시스템
- VIP 고객 인식
- 꽃이름 등 사물 상세정보

### 이미지 내 텍스트 인식



- 표지판 인식
- 우편번호 인식
- 자막인식
- 자동차 번호판 인식

# 1. 컴퓨터 비전 적용 분야

## 활용 사례

순번	구분	활용	비고
1	극장	관객 표정분석 및 영화평점 시스템 개발	
2	병원	등록된 환자가 입장 시 자동 도어 오픈 등록되지 않은 사람에 대한 출입 알림 제한된 구역에 출입 탐지	
3	보안업체	CCTV에 노출된 사람 및 차량 이력 기록 안면 인식을 이용한 사내 출입 시스템	
4	매장	감정 분석을 통한 매장 방문 고객 만족도 분석 시스템	
5	도시	CCTV 분석을 통한 도시 만족도 분석 시스템	



# 컴퓨터비전 활용사례 및 이미지 영상 다루기

김 효 관

교육목표: OpenCV 라이브러리를 이해하고 영상데이터 LOAD/WRITE 학습

# CONTENTS

1

라이브러리 소개

2

이미지 불러오기/저장하기

3

비디오 불러오기/저장하기

4

핵심정리 및 Q&A



## 1. 라이브러리 소개

opencv python


- 1 OpenCV는 1996년 인텔에서 시작 2000년 첫 배포
- 2 컴퓨터 비전, 머신러닝등 다양한 라이브러리 지원
- 3 Numpy를 통해 컴퓨터 그래픽, 이미지 연산 수행

```
pip install opencv_python
```

```
Microsoft Windows [Version 10.0.17763.503]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Wkopo>pip install opencv_python
Collecting opencv_python
  Downloading https://files.pythonhosted.org/packages/a3/50/04d0669afe884f137c2f4
9064275fe8c4a658254300a9eaf253d1e643085/opencv_python-4.1.0.25-cp37-cp37m-win_amd
64.whl (37.3MB)
100% |████████████████████████████████████████████████████████████████████████████████| 37.4MB 503kB/s
Requirement already satisfied: numpy>=1.14.5 in c:\Users\Wkopo\Anaconda3\lib\site-
packages (from opencv_python) (1.16.2)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.1.0.25

C:\Users\Wkopo>
```



The screenshot shows the JupyterLab interface. The top bar indicates the environment is 'OpenCV' and shows the last checkpoint was 2 minutes ago with unsaved changes. The left sidebar displays the file explorer with 'OpenCV' selected. The main area contains a terminal window showing the command 'pip install opencv-python' and its output, followed by a code editor with a Python script that imports cv2, numpy, and matplotlib.

Jupyter OpenCV Last Checkpoint: 2분 전 (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Save Undo Copy Paste Run Stop Restart Code

In [3]:

```
1 import cv2
2 import numpy
3 import matplotlib
```

[https://docs.opencv.org/2.4/modules/highgui/doc/reading\\_and\\_writing\\_images\\_and\\_video.html?highlight=imwrite](https://docs.opencv.org/2.4/modules/highgui/doc/reading_and_writing_images_and_video.html?highlight=imwrite)

<https://opencv-python-tutroals.readthedocs.org/en/latest/index.html>



## 2. 이미지 불러오기/저장하기

### 1. 이미지 불러오기/저장하기

- 불러오기: `cv2.imread`(파일경로, 옵션)

```
import cv2
import matplotlib.pyplot as plt

%matplotlib inline

imgfile = '../images/it_show.jpg'
# 이미지 불러오기, 색상표변환
img = cv2.imread(imgfile, cv2.IMREAD_COLOR)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

plt.axis("off")
plt.imshow(gray, cmap="gray")
plt.show()
```



IMREAD\_COLOR: 컬러로 읽음 (1)  
IMREAD\_GRAYSCALE: Grayscale로 읽음 (0)  
IMREAD\_UNCHANGED: alpha채널 포함해서 읽음 (-1)

## 2. 이미지 불러오기/저장하기

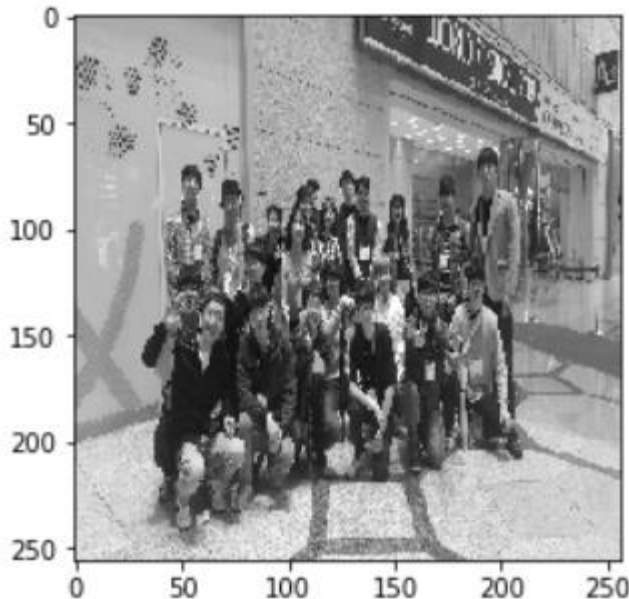
### 2. 이미지 크기 변환

- 이미지 크기변환: `cv2.resize(src=이미지변수, dsize = (가로, 세로))`

```
IMG_SIZE = 256
# 이미지 크기 변환
new_img = cv2.resize(src= gray, dsize = (IMG_SIZE, IMG_SIZE))
plt.imshow(new_img, cmap = 'gray')
plt.show()
```

## matplotlib 에서 컬러 표현 시

```
b,g,r = cv2.split(img)
img2 = cv2.merge([r,g,b])
plt.imshow(img2)
```





## 2. 이미지 불러오기/저장하기

### 3. 이미지 불러오기/저장하기

- 불러오기: `cv2.imread`(파일경로, 옵션), `cv2.imshow`(“타이틀명“, 불러온 이미지 변수)
- 저장하기: `cv2.imwrite`(저장경로, 이미지 저장변수)

```
import numpy as np
import cv2
```

```
def showImage():
```

```
    imgfile = '../images/it_show.jpg'
```

```
    # 이미지 불러오기
```

```
    img = cv2.imread(imgfile, cv2.IMREAD_COLOR)
```

```
    img = cv2.resize(src=img, dsize=(1024, 768))
```

```
    # 창조절가능 이미지 뷰
```

```
    cv2.namedWindow('it_show', cv2.WINDOW_NORMAL)
```

```
    cv2.imshow('it_show', img)
```

```
    # wait for keyboard input, unit(ms)
```

```
    # 파라미터 0인경우 무한 대기
```

```
    inputKey = cv2.waitKey(0)
```

```
    if inputKey == 27: # esc
```

```
        cv2.destroyAllWindows()
```

```
    elif inputKey == 26: # ctrl + z
```

```
        cv2.imwrite('../images/it_show_copy.jpg', img)
```

```
        cv2.destroyAllWindows()
```

```
    else:
```

```
        cv2.destroyAllWindows()
```

```
showImage()
```

키 설명은  
다음장에!

이미지 불러오기/저장하기

```
1 import numpy as np
2 import cv2
3
4 def showImage():
5     imgfile = '../images/it_show.jpg'
6     # load image, color
7     img = cv2.imread(imgfile, cv2.IMREAD_COLOR)
8
9     cv2.namedWindow('it_show', cv2.WINDOW_NORMAL)
10    # window title and image object
11    cv2.imshow('it_show', img)
12    # wait for keyboard input, unit(ms)
13    # if parameter is zero, wait for keyboard input, esc = 27
14    inputKey = cv2.waitKey(0) & 0xFF
15
16    if inputKey == 27:
17        cv2.destroyAllWindows()
18    elif inputKey == ord('c'):
19        cv2.imwrite('../images/it_show_copy.jpg', img)
20        cv2.destroyAllWindows()
21    else:
22        cv2.destroyAllWindows()
23
24 showImage()
```



사용자 > kopo > Python\_CJ > Python\_ST\_EX > images

이름

it\_show.jpg

it\_show\_copy.jpg

## 2. 이미지 불러오기/저장하기

### 파이썬 주요 키 입력 (event key code)

순번	숫자	키입력
1	27	ESC 키
2	26	Ctrl + Z
3	24	Ctrl + X
4	3	Ctrl + C
5	0	아무키
6	32	스페이스바
7	q	q

<https://keycode.info/>

### 3. 비디오 불러오기/저장하기

#### 참조 (이미지 위에 사각형 삽입)

- 사각형 그리기: `cv2.rectangle`("이미지변수명", (시작x축, 시작y축), 색상, 색상채널)

```
import cv2
```

```
# 이미지 불러오기 및 크기 조절
```

```
sampleImg = cv2.imread("../images/it_show.jpg")
```

```
sampleImg = cv2.resize(src=sampleImg, dsize=(256,256))
```

```
# 불러온 이미지 내 사각형 그리기
```

```
rectImg = cv2.rectangle(sampleImg, (10,10), (200,200), (255,0,0))
```

```
cv2.imshow("img_show", rectImg)
```

```
k = cv2.waitKey(0)
```

```
if k==27:
```

```
    cv2.destroyAllWindows()
```

```
else:
```

```
    cv2.destroyAllWindows()
```

(10,10) 좌측상단 기준 10, 10 에서 시작  
(200,200) x,y 축 길이  
(255,0,0) :BGR 컬러코드



### 3. 비디오 불러오기/저장하기

#### 참조 (이미지 위에 글자 삽입)

- 글자 삽입: `cv2.putText`("이미지변수명", (시작x축, 시작y축), 폰트, 크기, 색상채널)

```
import cv2
```

```
# 이미지 불러오기 및 크기 조절
```

```
sampleImg = cv2.imread("../images/it_show.jpg")
```

```
sampleImg = cv2.resize(src=sampleImg, dsize=(256,256))
```

```
# 불러온 이미지 내 사각형 그리기
```

```
rectImg = cv2.rectangle(sampleImg, (10,10), (200,200), (255,0,0))
```

```
cv2.putText(rectImg, "haiteam", (0,100), cv2.FONT_HERSHEY_SIMPLEX, 1, (0,255,0))
```

```
cv2.imshow("img_show", rectImg)
```

```
k = cv2.waitKey(0)
```

```
if k==27:
```

```
    cv2.destroyAllWindows()
```

```
else:
```

```
    cv2.destroyAllWindows()
```

(0,100) 좌측상단 기준 10, 10 에서 시작  
cv2.FONT\_HERSHEY... : 폰트속성  
(0,255,0) :BGR 컬러코드



Hershey Simplex: The quick brown fox jumps over the lazy dog 0123456789

Hershey Plain: The quick brown fox jumps over the lazy dog 0123456789

Hershey Duplex: The quick brown fox jumps over the lazy dog 0123456789

Hershey Complex: The quick brown fox jumps over the lazy dog 0123456789

Hershey Triplex: The quick brown fox jumps over the lazy dog 0123456789

Hershey Complex Small: The quick brown fox jumps over the lazy dog 0123456789

*Hershey Script Simplex: The quick brown fox jumps over the lazy dog 0123456789*

*Hershey Script Complex: The quick brown fox jumps over the lazy dog 0123456789*

# 3. 비디오 불러오기/저장하기

## 1. 비디오 불러오기

- 불러오기: `cv2.VideoCapture`(파일경로 or 웹캠번호)  
`ret, frame = cap.read()`

```
import cv2
cap = cv2.VideoCapture("../images/hkcam.avi")
#cap = cv2.VideoCapture(0)

width = cap.get(cv2.CAP_PROP_FRAME_WIDTH)
height = cap.get(cv2.CAP_PROP_FRAME_HEIGHT)
```

## 비디오가 정상적으로 열렸는지 확인

```
while (cap.isOpened):
    ret, frame = cap.read()
```

```
if ret:
    cv2.imshow("video", frame)
```

# 33ms 30fps = 1000ms/30frame -> 1000/30 = 33

```
k = cv2.waitKey(33)
```

```
if k == 27:
```

```
    break
```

```
else:
```

```
    break
```

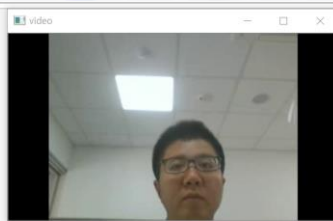
```
cap.release()
```

```
cv2.destroyAllWindows()
```

디바이스 활용 캡처 시  
`cap.set(3, width)`  
`cap.set(4, height)` 활용하여  
웹캠 출력사이즈를 조정할 수 있다.

30 fps 를 적용 시 (초당 30프레임 업데이트 시)  
33 설정하면 됨.  
\* 0은 무한 대기

```
1 import cv2
2 cap = cv2.VideoCapture("../images/hkcam.avi")
3 #cap = cv2.VideoCapture(0)
4
5 width = cap.get(cv2.CAP_PROP_FRAME_WIDTH)
6 height = cap.get(cv2.CAP_PROP_FRAME_HEIGHT)
7
8 ## 비디오가 정상적으로 열렸는지 확인
9 while (cap.isOpened):
10     ret, frame = cap.read()
11
12     if ret:
13         cv2.imshow("video", frame)
14
15         # 33ms 30fps = 1000ms/30frame -> 1000/30 = 33
16         k = cv2.waitKey(33)
17         if k == 27:
18             break
19     else:
20         break
21 cap.release()
22 cv2.destroyAllWindows()
```



30fps -> 30 frame : 1000ms -> 1frame : 33ms  
이미지 하나를 33ms 마다 업데이트 해야함

\* NTSC는 미국, 캐나다, 대한민국 등에서 널리 사용하는  
아날로그 텔레비전 방식이다.



### 3. 비디오 불러오기/저장하기

```
import datetime
import cv2
```

```
#capture = cv2.VideoCapture(0)
capture =
cv2.VideoCapture("../images/image_sample.mp4")
fcc = cv2.VideoWriter_fourcc('D','I','V','X')
record = False
```

```
width = int(capture.get(3))
height = int(capture.get(4))
```

# 캡처 로직 (우측 참조)

```
capture.release()
cv2.destroyAllWindows()
```

디빅: DivX사가 개발한 영상 코덱  
(원래 크기의 15%까지 압축가능)

```
while (capture.isOpened):
    ret, frame = capture.read()
    if ret == False:
        break
    cv2.imshow("VideoFrame", frame)
```

더이상 영상 없는  
경우 탈출!

```
now = datetime.datetime.now().strftime("%d_%H-%M-%S")
key = cv2.waitKey(33)#1) & 0xFF
```

```
if key == 27: # esc 종료
```

```
    break
```

```
elif key == 26: # ctrl + z
```

```
    print("캡처")
```

```
    cv2.imwrite("./" + str(now) + ".png", frame)
```

```
elif key == 24: # ctrl + x
```

```
    print("녹화 시작")
```

```
    record = True
```

```
    outfps = 30
```

```
    video = cv2.VideoWriter("./" + str(now) + ".avi",
                             fcc, # 프레임 압축코드
```

```
    outfps,
```

```
    (width, height))
```

```
elif key == 3:
```

```
    print("녹화 중지") # ctrl + c
```

```
    record = False
```

```
    video.release()
```

```
if record == True:
```

```
    print("녹화 중..")
```

```
    video.write(frame)
```

### 3. 비디오 불러오기/저장하기

#### 2. 비디오 저장하기

사용자 > kopo > Python\_CJ > Python\_ST\_EX > images



it\_show.jpg



it\_show\_copy.jpg

### 3. 비디오 불러오기/저장하기

일반 실습

[실습]  
비디오 영상 일부를  
Gray 컬러로 캡처하여 저장하세요

### 3. 비디오 불러오기/저장하기

일반 실습

[실습]  
자유롭게 동영상을 웹캠을 통해  
저장해보세요.

# 3. 비디오 불러오기/저장하기

## 3. 참조

```
import cv2

#cap = cv2.VideoCapture(0)
cap = cv2.VideoCapture("../images/hkcam.avi")

print('width :%d, height : %d' % (cap.get(3), cap.get(4)))

while(True):
    #ret, frame = cap.read() # Read 결과와 frame
    # 비디오를 한 프레임씩 읽는다. 정상 ret: True, 에러: False
    ret, frame = cap.read() # Read 결과와 frame

    print(frame)
    key = cv2.waitKey(33)
    if(ret) :
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY) # 입력 받은 화면 Gray로 변환

        cv2.imshow('frame_color', frame) # 컬러 화면 출력 (윈도우 타이틀명, frame)
        cv2.imshow('frame_gray', gray) # Gray 화면 출력
        if key==27: # esc 키 누르면 종료
            break
    else:
        break
cap.release()
cv2.destroyAllWindows()
```

# 3. 비디오 불러오기/저장하기

## 3. 참조

- 사각형 그리기: `cv2.rectangle`("이미지변수명", (시작x축, 시작y축), 색상, 색상채널)

```
import cv2

# 이미지 불러오기 및 크기 조절
sampleImg = cv2.imread("../images/it_show.jpg")
sampleImg = cv2.resize(src=sampleImg, dsize=(256,256))

# 불러온 이미지 내 사각형 그리기
rectImg = cv2.rectangle(sampleImg, (10,10), (200,200), (255,0,0))

cv2.imshow("img_show", rectImg)

k = cv2.waitKey(0)

if k==27:
    cv2.destroyAllWindows()
else:
    cv2.destroyAllWindows()
```





# 3. 비디오 불러오기/저장하기

## 3. 참조

- 사각형 그리기: `cv2.rectangle`("이미지변수명", (시작x축, 시작y축), 색상, 색상채널)

```
import cv2
```

```
# 이미지 불러오기 및 크기 조절
```

```
sampleImg = cv2.imread("../images/it_show.jpg")
```

```
sampleImg = cv2.resize(src=sampleImg, dsize=(256,256))
```

```
# 불러온 이미지 내 사각형 그리기
```

```
rectImg = cv2.rectangle(sampleImg, (10,10), (200,200), (255,0,0))
```

```
cv2.imshow("img_show", rectImg)
```

```
k = cv2.waitKey(0)
```

```
if k==27:
```

```
    cv2.destroyAllWindows()
```

```
else:
```

```
    cv2.destroyAllWindows()
```

(10,10) 좌측상단 기준 10, 10 에서 시작  
(200,200) x,y 축 길이  
(255,0,0) :BGR 컬러코드



## 4. 핵심정리 및 Q&A

### 기억합시다

1

OpenCV 라이브러리를 익히고 Tutorial 학습

2

간단한 이미지 데이터 불러오기 / 저장하기

3

간단한 비디오 데이터 불러오기 / 저장하기

4

딤러닝과 연동을 위한 색상/크기 변경 방법 학습

5

딤러닝 예측결과를 화면에 시연하기 위한 사각영역 및 글자 그리는 방법 학습

## 4. 핵심정리 및 Q&A

감사합니다.

