Image Processing

실습 12.

2022. 05. 23.

실습 수업 소개

- **과목 홈페이지**
  - 충남대학교 사이버 캠퍼스 (http://e-learn.cnu.ac.kr)

- **TA 연락처**
  - 신준호
  - wnsgh578@naver.com

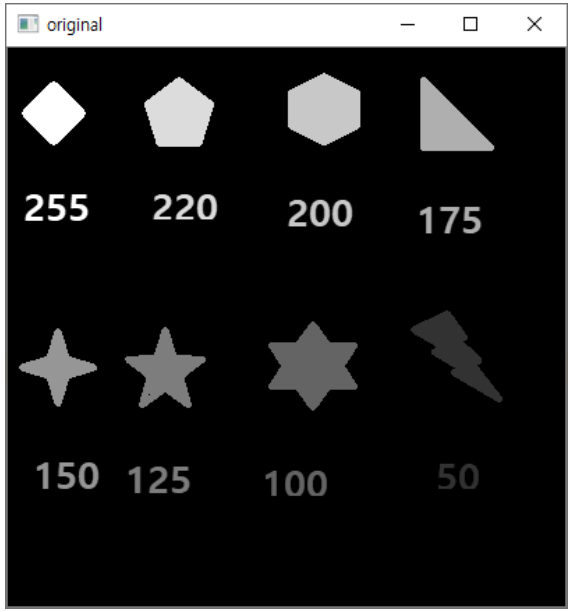
- **튜터 연락처**
  - 한승오
  - so.h4ns@gmail.com

- **실습 중 질문사항**
  - 수업중 질문 or 메일을 통한 질문
  - 메일로 질문할 때 [IP] 를 제목에 붙여주세요

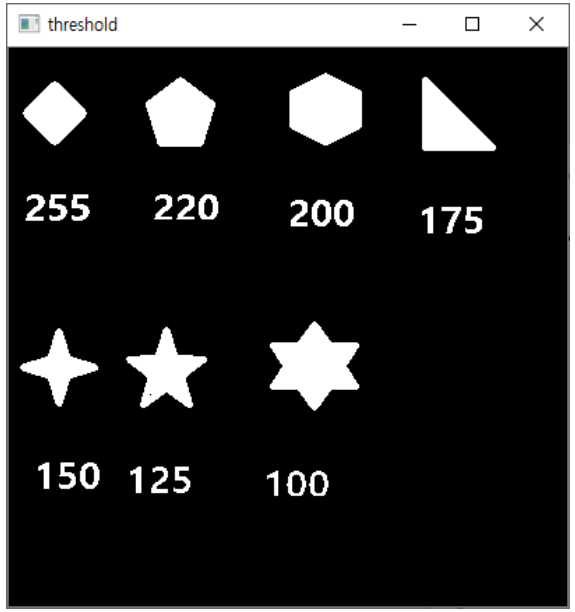
목 차

실습

- **Threshold**
  - ➢ 영상을 흑/백으로 분류하여 처리하는 것을 이진화 라고 한다.
  - ➢ 이때 기준이 되는 임계값을 threshold value라고 한다



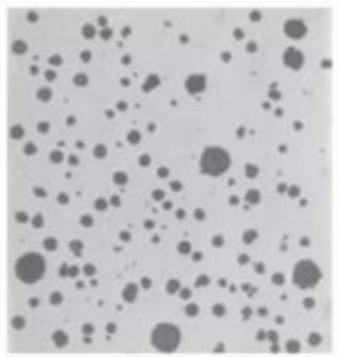
original

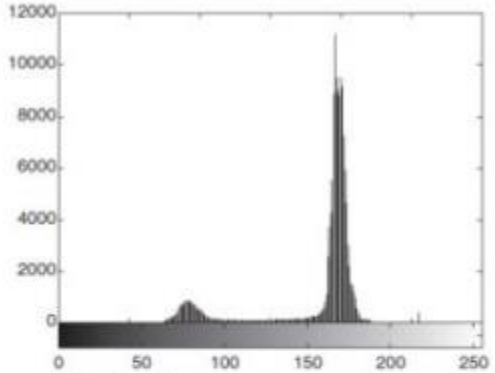


thresholding
threshold value = 85

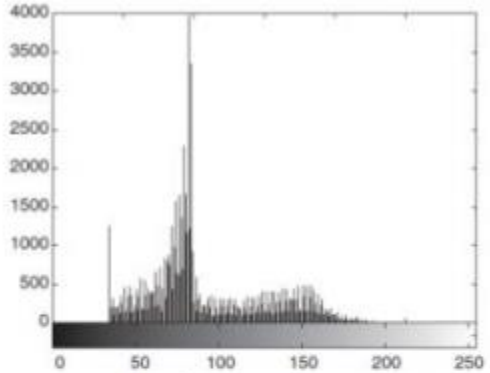
실습

- **Threshold**

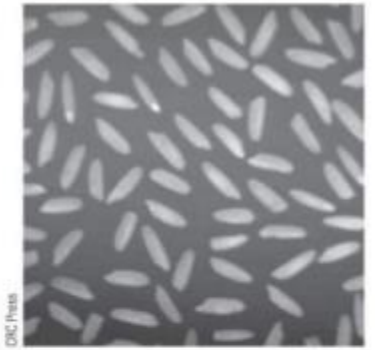


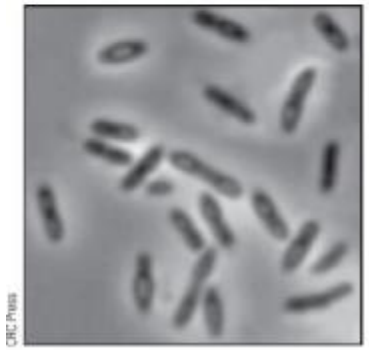


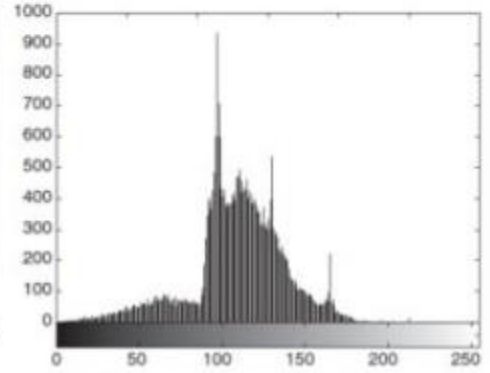
Nodules image : nodules.tif



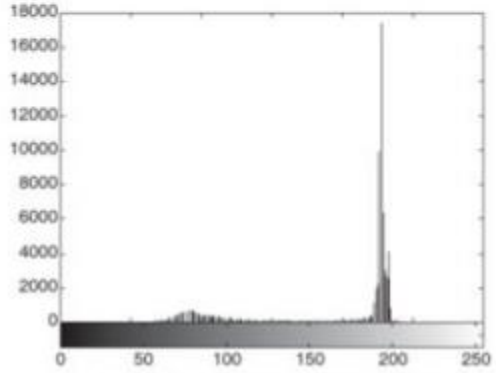
Rice image : rice.tif







Bacteria image : bacteria.tif



Coins image : eight.tif

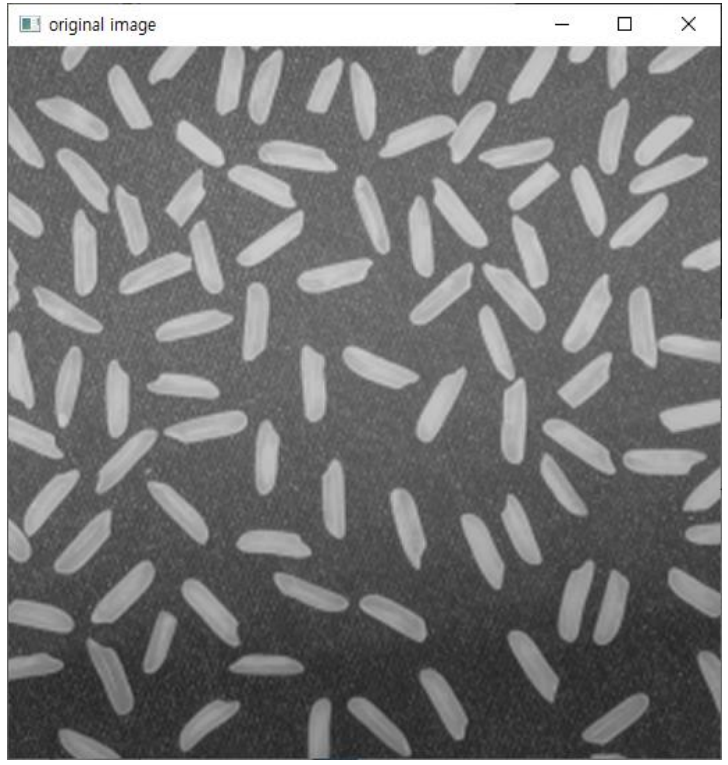
실습

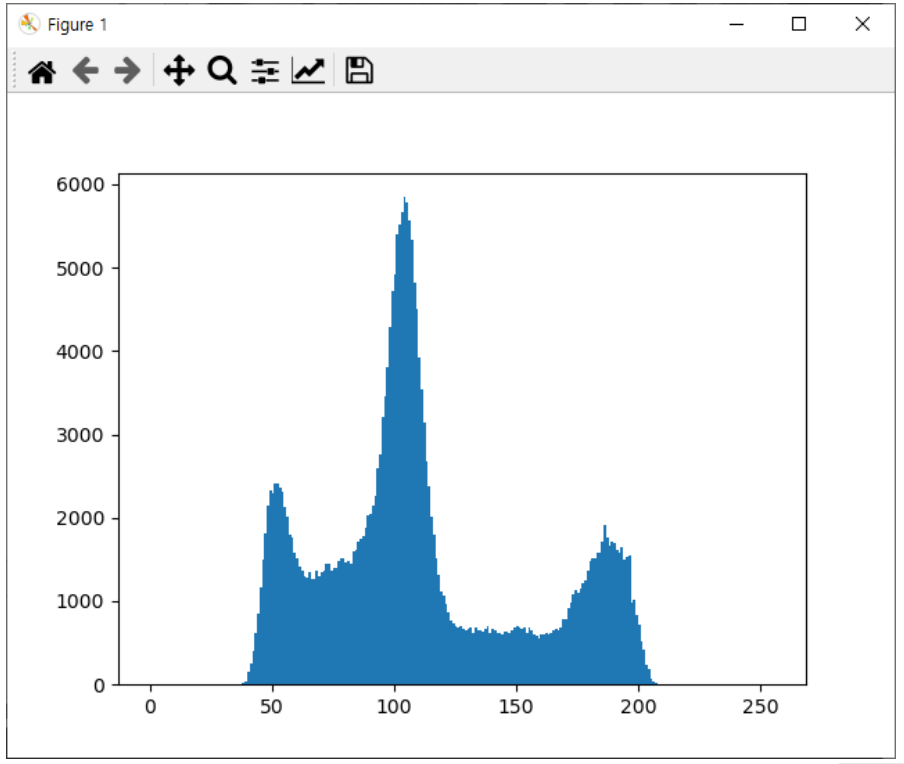
- Threshold 실습

```python
def apply_threshold(img, th=120):
    dst = np.zeros(img.shape, img.dtype)
    ##################################################
    # TODO                                           #
    # Apply threshold                                #
    ##################################################

    return dst

if __name__ == '__main__':
    img = cv2.imread('circles_adaptive_threshold.png', cv2.IMREAD_GRAYSCALE)

    img_th = apply_threshold(img, th=120)

    cv2.imshow('original image', img)
    cv2.imshow('threshold applied image', img_th)

    plt.hist(img.ravel(), 256, [0, 256])
    plt.show()
    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

실습

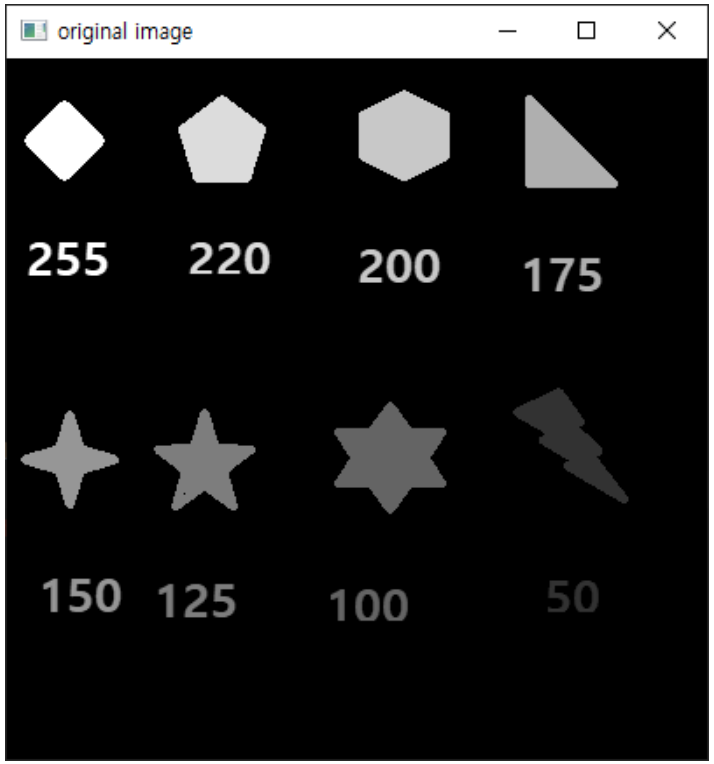
- **Threshold 실습**

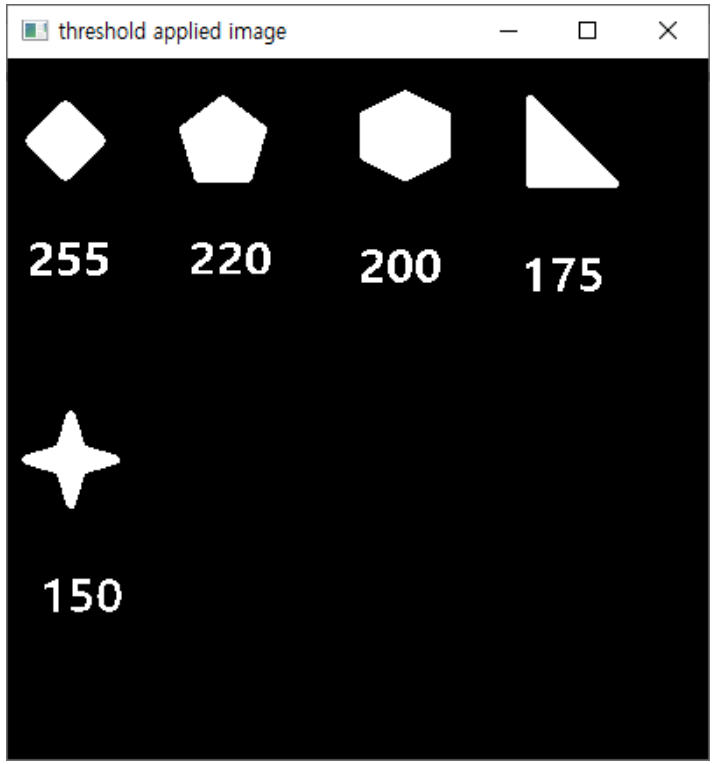


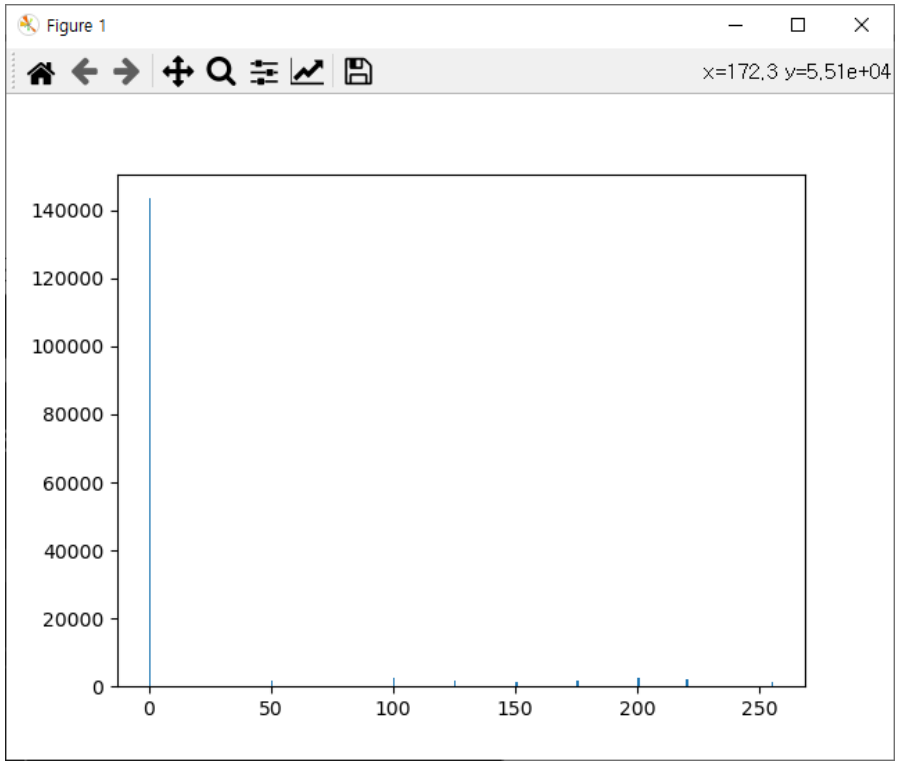


실습

- Threshold 실습

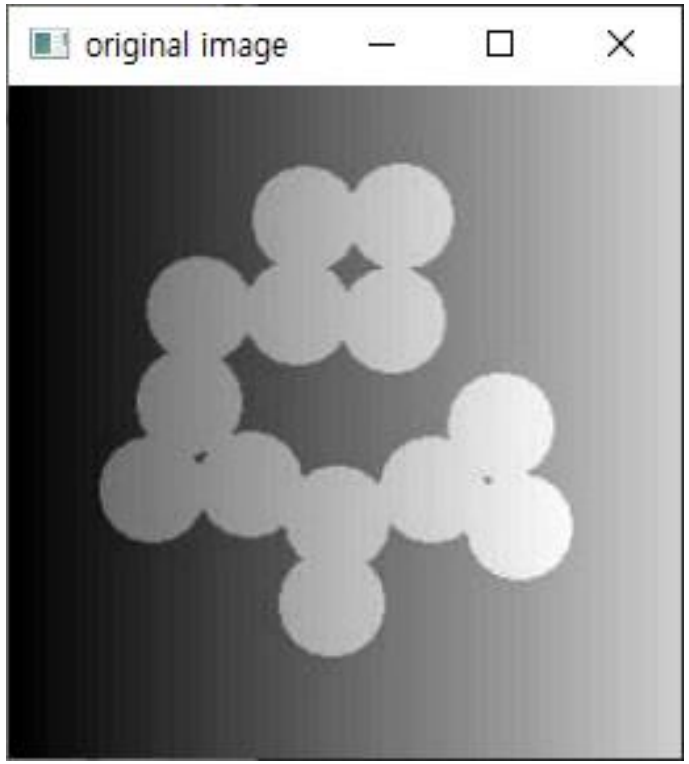


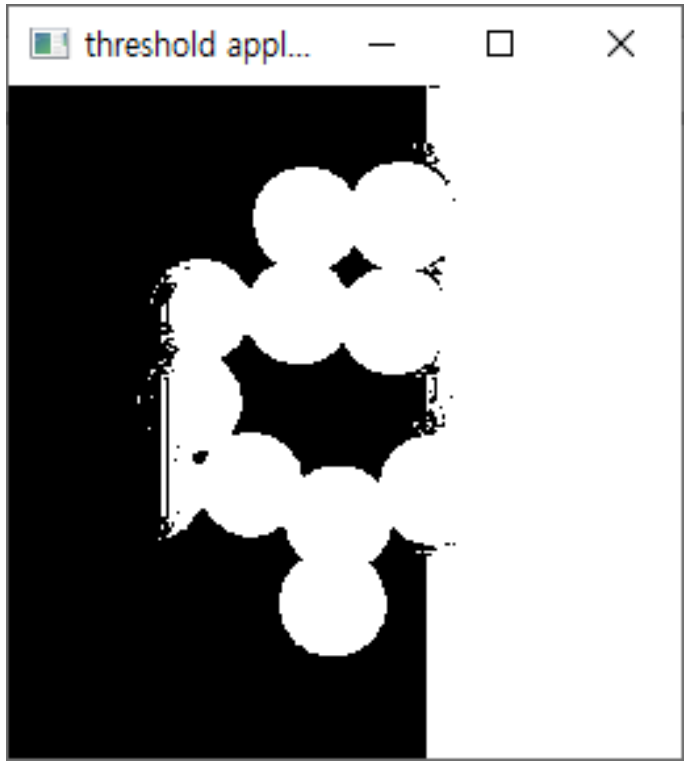


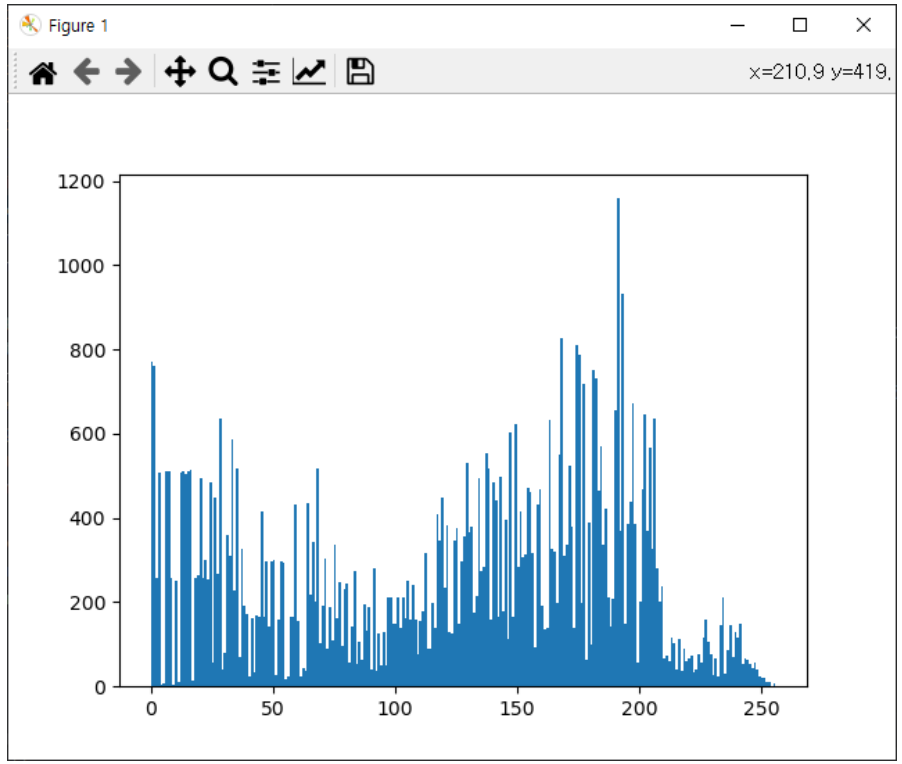


실습

- **Threshold 실습**

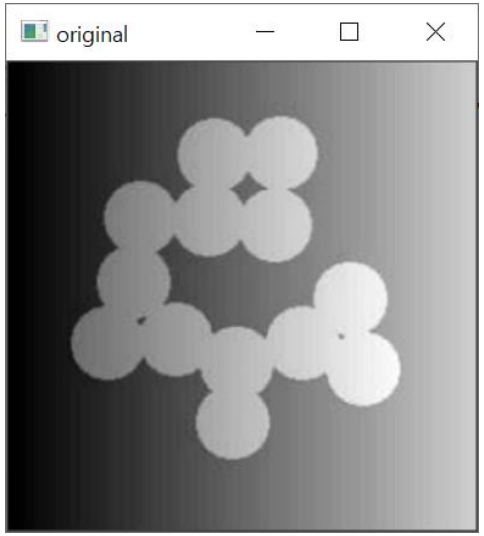


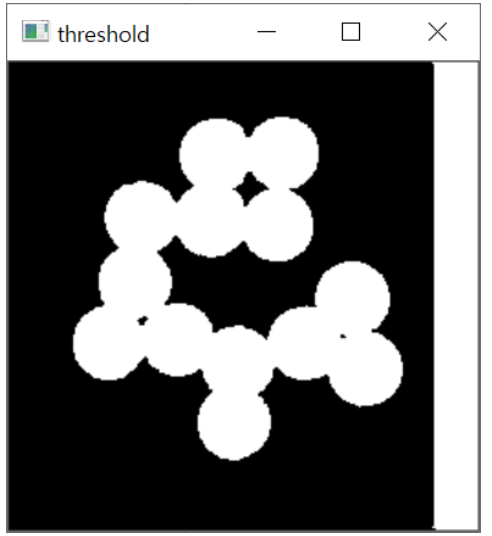




실습

- **Adaptive Threshold 실습**





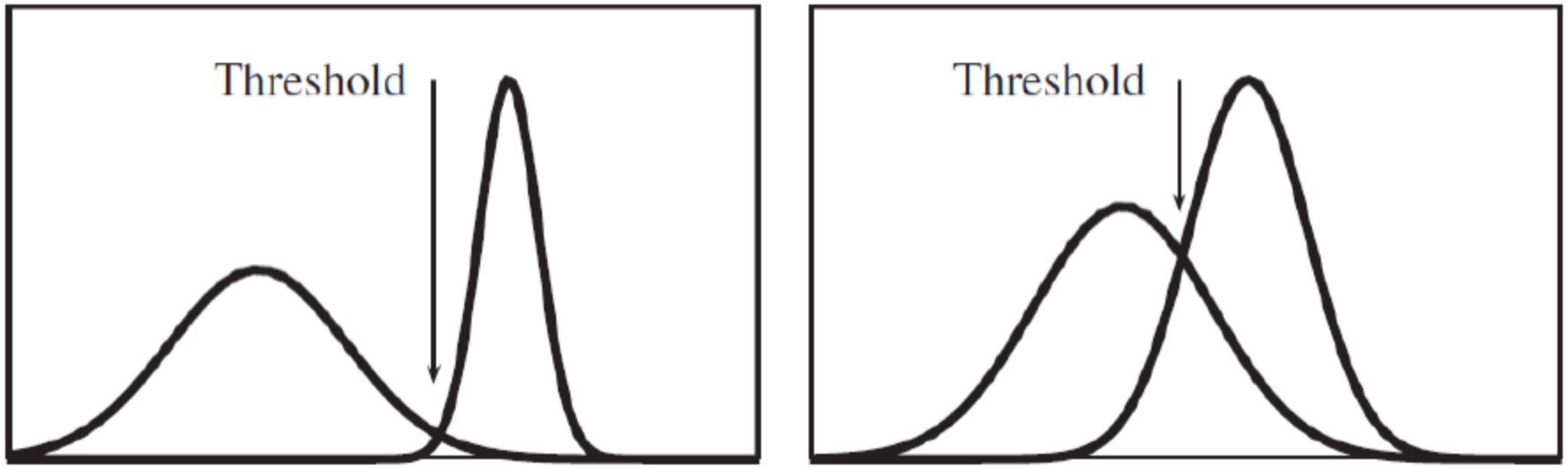
실습

- Adaptive Threshold 실습

```python
def adaptive_threshold(img, group_num=4):
    img_split = np.hsplit(img, group_num)

    thresholds = list()
    dst = list()

    ###################################################
    # TODO                                            #
    # 구분된 이미지마다 cv2의 otsu 방법으로 이진화.       #
    # 이후, 하나의 이미지로 다시 합친다.                  #
    ###################################################

    return dst, thresholds


if __name__ == '__main__':
    img = cv2.imread('circles_adaptive_threshold.png', cv2.IMREAD_GRAYSCALE)

    dst, val = adaptive_threshold(img, group_num=4)
    print('Threshold: ', val)

    cv2.imshow('original', img)
    cv2.imshow('adaptive threshold', dst)
    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

과제

- **Threshold: Otsu’s method**



과제

- **Threshold: Otsu's method**

$$p_i = n_i / MN \quad \text{and} \quad \sum_{i=0}^{L-1} p_i = 1$$

$$q_1(k) = \sum_{i=0}^{k} p(i) \qquad q_2(k) = \sum_{i=k+1}^{L-1} p(i)$$

$$m_1(k) = \frac{\sum_{i=0}^{k} ip(i)}{\sum_{i=0}^{k} p(i)} = \frac{1}{q_1(k)} \sum_{i=0}^{k} ip(i) \qquad m_2(k) = \frac{\sum_{i=k+1}^{L-1} ip(i)}{\sum_{i=k+1}^{L-1} p(i)} = \frac{1}{q_2(k)} \sum_{i=k+1}^{L-1} ip(i)$$

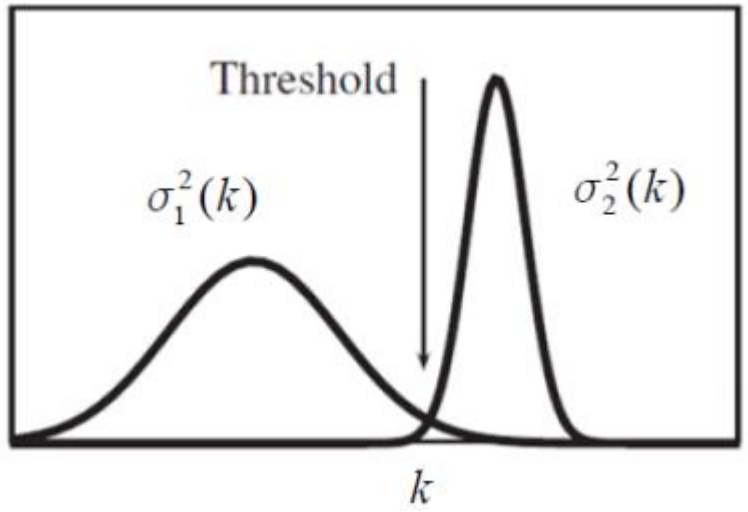
$$\sigma_1^2(k) = \frac{1}{q_1(k)} \sum_{i=0}^{k} [i - m_1(k)]^2 p(i) \qquad \sigma_2^2(k) = \frac{1}{q_2(k)} \sum_{i=k+1}^{L-1} [i - m_2(k)]^2 p(i)$$

$$= \frac{1}{q_1(k)} \sum_{i=0}^{k} i^2 p_i - m_1^2(k) \qquad = \frac{1}{q_2(k)} \sum_{i=k+1}^{L-1} i^2 p(i) - m_2^2(k)$$

For an entire image,

$$m_G = \sum_{i=0}^{L-1} ip(i), \qquad \sigma_G^2 = \sum_{i=0}^{L-1} [i - m_G]^2 p(i)$$

$$q_1(k) + q_2(k) = 1$$
$$q_1(k)m_1(k) + q_2(k)m_2(k) = m_G$$



$$\sigma_W^2(k) = q_1(k)\sigma_1^2(k) + q_2(k)\sigma_2^2(k)$$

$$\sigma_B^2(k) = q_1(k)[m_1(k) - m_G]^2 + q_2(k)[m_2(k) - m_G]^2$$
$$= q_1(k)q_2(k)[m_1(k) - m_2(k)]^2$$

$$k_{opt} = \arg\min_k \sigma_W^2(k)$$

$$k^* = \arg\max_k \sigma_B^2(k)$$

과제

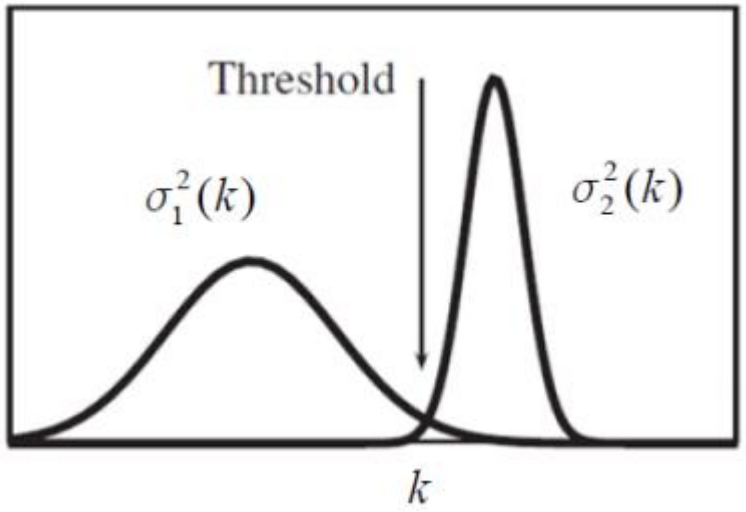
- **Threshold: Otsu's method**
  - Iterative하게 구할 수 있음

$$q_1(0) = p_0, m_1(0) = 0$$

$$q_1(k+1) = q_1(k) + p(k+1)$$

$$m_1(k+1) = \frac{q_1(k)m_1(k)+(k+1)p(k+1)}{q_1(k+1)}$$

$$m_2(k+1) = \frac{q_2(k)m_2(k)-(k+1)p(k+1)}{q_2(k+1)}$$



과제

- **Threshold: Otsu's method**

```python
def my_otsu_threshold(img):
    hist, bins = np.histogram(img.ravel(),256,[0,256])
    p = hist / np.sum(hist) + 1e-7

    ########################################################
    # TODO                                                 #
    # Otsu 방법을 통해 threshold 구한 후 이진화 수행          #
    # cv2의 threshold 와 같은 값이 나와야 함                 #
    ########################################################

    th = ???
    dst = apply_threshold(img, th)

    return th, dst
```

```
Threshold from cv2: 88.0
Threshold from my: 88
```

original

cv2

과제

과제

- **제출 방법**
  - 코드 파일
    - 구현 결과가 포함된 python 파일(.py)
  - 보고서
    - [IP]201900000_홍길동_11주차_과제.pdf
    - 보고서 양식 사용
    - PDF 파일 형식으로 제출(pdf가 아닌 다른 양식으로 제출시 감점)
  - 제출 파일
    - [IP]201900000_홍길동_11주차_과제.zip
    - .py 파일과 pdf 보고서를 하나의 파일로 압축한 후, 양식에 맞는 이름으로 제출

QnA