2020년 컴퓨터그래픽스

- HW 06 -

제출일자	2020.11.10.
이 름	장수훈
학 번	201402414
분 반	00

구현코드

```
if not RANSAC:

    A = []
    B = []
    for idx, point in enumerate(points):
        '''
        #ToDo
        #A, B 완성
        # A.append(???) 이런식으로 할 수 있음
        # 결과만 잘 나오면 다른방법으로 해도 상관없음
        '''
        A.append([points[idx][0][0], points[idx][0][1], 1, 0, 0, 0])
        A.append([0, 0, 0, points[idx][0][0], points[idx][0][1], 1])
        B.append(points[idx][1][0])
        B.append(points[idx][1][1])

    A = np.array(A)
    B = np.array(B)
```

이론과 실습ppt에 나와 있는 이론을 이용해 A와 B에 각각 points의 x, y, x′, y′ 값들을 넣어주었다.

$$x_1' = ax_1 + by_1 + c \qquad x_2' = ax_2 + by_2 + c$$
$$y_1' = dx_1 + ey_1 + f \qquad y_2' = dx_2 + ey_2 + f$$

$$x_3' = ax_3 + by_3 + c \qquad x_4' = ax_4 + by_4 + c$$
$$y_3' = dx_3 + ey_3 + f \qquad y_4' = dx_4 + ey_4 + f$$

$$\underset{8}{\updownarrow}\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ & & & \vdots & & \end{bmatrix}\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} x_1' \\ y_1' \\ \vdots \end{bmatrix} \Rightarrow \mathbf{Ax = b}$$

$$\mathbf{A} \qquad \mathbf{x} \qquad \mathbf{b}$$

```
#ToDo
#X 완성
#np.linalg.inv(V) : V의 역행렬 구하는것
#np.dot(V1, V2) : V1과 V2의 행렬곱
# V1.T : V1의 transpose
'''

X = np.dot(np.dot(np.linalg.inv(np.dot(A.T,A)),A.T),B)
```

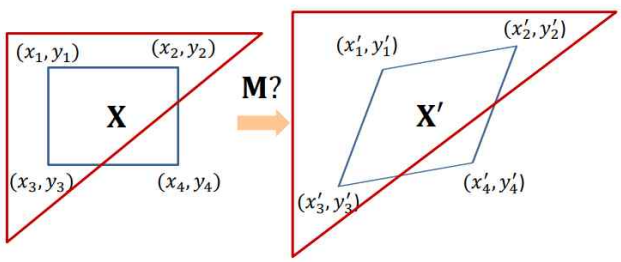
이것 또한 이론과 실습 ppt에 있는 공식을 이용해 구해놓은 A,B를 사용하여 X를 구하였다.

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$$

```
# ToDo
# 위에서 구한 X를 이용하여 M 완성
'''
M = [[X[0], X[1], X[2]],
     [X[3], X[4], X[5]],
     [0,    0,       1]]
```

M또한 구해놓은 X를 이용해 구하였다.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} ax + by + c \\ dx + ey + f \\ 1 \end{bmatrix}$$



```
h, w = img1.shape[:2]
dst = np.zeros((h, w,3))
count = dst.copy()

for row in range(h):
    for col in range(w):
        vec = np.dot(M_, np.array([[col,row,1]]).T)
        c = vec[0,0]
        r = vec[1,0]
        c_left = int(c)
        c_right = min(int(c+1),w-1)
        r_top = int(r)
        r_bottom = min(int(r+1),h-1)
        s = c - c_left
        t = r - r_top

        intensity = (1-s) * (1-t) * img1[r_top,c_left] + s * (1-t) * img1[r_top,c_right] + (1-s) * t * im
        dst[row, col] = intensity
dst = dst.astype(np.uint8)
```

backward 방식을 사용하려 했지만 잘 이해가 되지 않아서 구현을 못하였다.

```
for idx, point in enumerate(three_points):
    '''
    #ToDo
    #A, B 완성
    # A.append(???) 이런식으로 할 수 있음
    # 결과만 잘 나오면 다른방법으로 해도 상관없음
    '''
    A.append([points[idx][0][0], points[idx][0][1], 1, 0, 0, 0])
    A.append([0, 0, 0, points[idx][0][0], points[idx][0][1], 1])
    B.append(points[idx][1][0])
    B.append(points[idx][1][1])

A = np.array(A)
B = np.array(B)
```

ransac 방식도 위와같이 A와 B, X를 채워넣었다.

```python
def L2_distance(vector1, vector2):
    '''
    #vector1과 vector2의 거리 구하기 (L2 distance)
    #distance 는 스칼라
    #np.sqrt(), np.sum() 를 잘 활용하여 구하기
    #L2 distance를 구하는 내장함수로 거리를 구한 경우 감점
    '''
    # distance = np.sqrt(np.sum(np.dot(vector1-vector2,vector1-vector2)))
    distance = np.sqrt(np.sum((vector1 - vector2)**2))
    # distance2 = np.linalg.norm(vector1-vector2)
    # if distance != distance2:
    #     print('ffff')

    # print(distance)
    return distance
```

L2 distance를 구하는 식을 이용해 구현하였다.

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

결과값

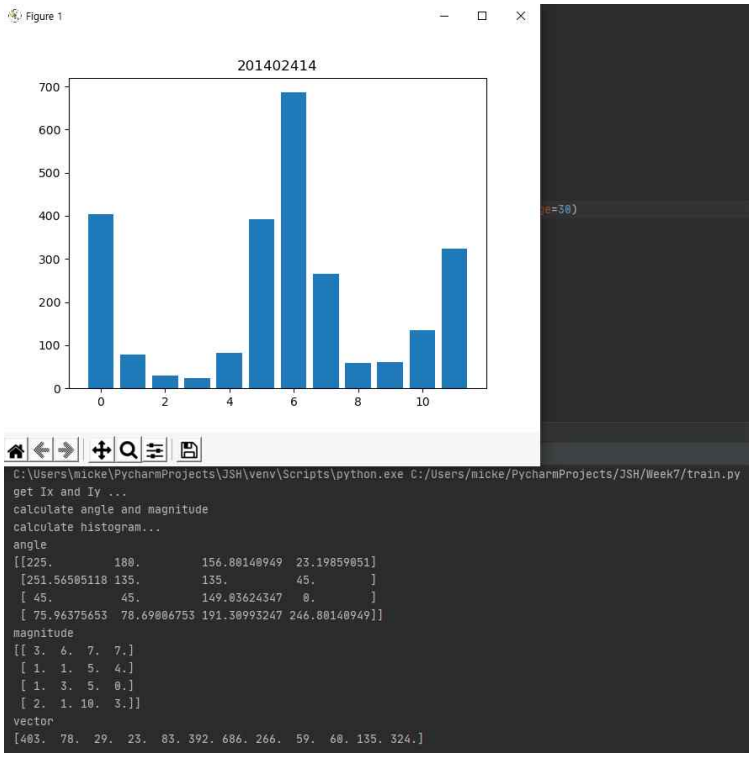
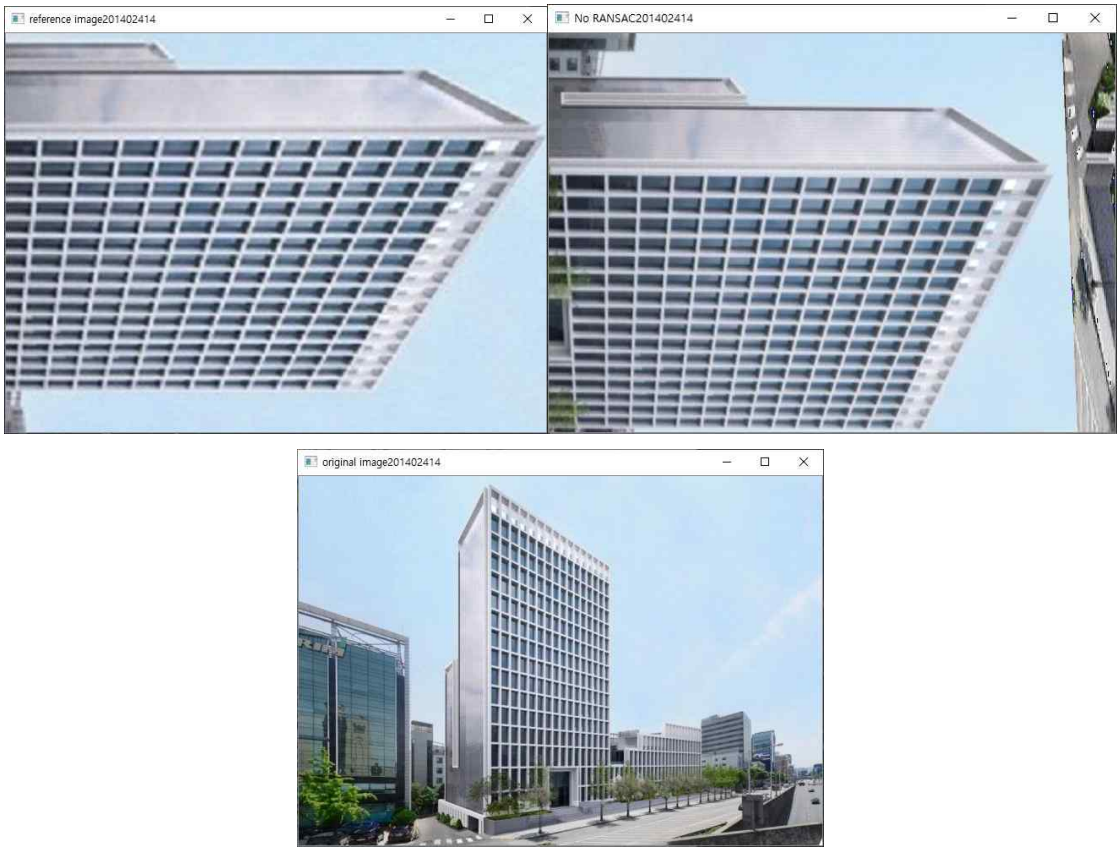


Figure 1
201402414
700
600
500
400
300
200
100
0
0
2
4
6
8
10
C:\Users\micke\PycharmProjects\JSH\venv\Scripts\python.exe C:/Users/micke/PycharmProjects/JSH/Week7/train.py
get Ix and Iy ...
calculate angle and magnitude
calculate histogram...
angle
[[225.   180.   156.80140949 23.19859051]
 [251.56505118 135.   135.   45.   ]
 [45.   45.   149.83624347   0.   ]
 [75.96375653 78.69006753 191.30993247 246.80140949]]
magnitude
[[3. 6. 7. 7.]
 [1. 1. 5. 4.]
 [1. 3. 5. 0.]
 [2. 1. 10. 3.]]
vector
[403. 78. 29. 23. 83. 392. 686. 266. 59. 68. 135. 324.]

실습



결과가 심상치 않다... 왼쪽이 짤려서 오른쪽에 붙었다.

느낀점

backward 방식과 forward 방식이 제대로 이해가 되지않아서 힘들었다.

난이도

이번과제가 여태까지 나왔던 과제중에서 제일 어려웠던 과제같다.