2021-09-13 조수익

1. 연립방정식의 해는 직선의 교점을 나타내는 직선을 그려 교차점을 표시하라

5x + 2x = 9 -5x + 3x = 1

* 코드
* import numpy as np

import matplotlib.pyplot as plt

K = np.array([[5, 2], [-5, 3]])

B = np.array([9, 1])

x = np.linalg.solve(K, B)

# 해벡터인 x를 그래프로 표현하기 위해 x1, y1로 표기합니다.

x1, y1 = x

print(x1, y1)

fig = plt.figure()

ax = fig.add\_subplot(1, 1, 1)

# 선형 방정식을 직선으로 그립니다.

a1 = K[:, 0]

b1 = K[:, 1]

c1 = -B

for c1, c2, c3 in zip(K[:, 0], K[:, 1], B):

x = np.linspace(-5, 5, 100)

y = (c3 - c1 \* x) / c2

ax.plot(x, y, color="blue")

ax.plot(x1, y1, 'go')

ax.axis([-5, 5, -5, 5])

ax.set\_xticks(range(-5, 5))

ax.set\_yticks(range(-5, 5))

ax.grid()

ax.set\_axisbelow(True)

ax.set\_aspect('equal', adjustable='box')

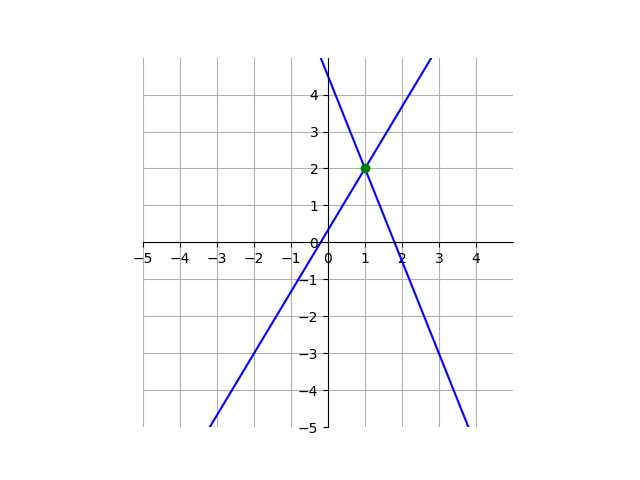
ax.spines['left'].set\_position('zero')

ax.spines['bottom'].set\_position('zero')

ax.spines['right'].set\_color('none')

ax.spines['top'].set\_color('none')

plt.show()

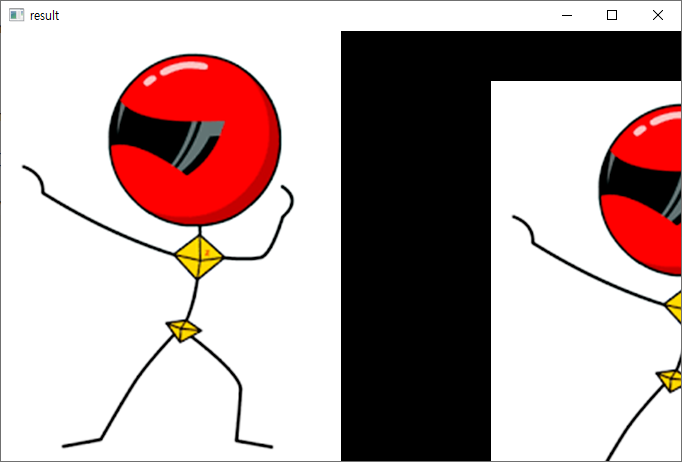


1. 다음 행렬 이동식을 참고하여 이미지를 로드하여 가로 150픽셀 세로 50픽셀씩 이동하는 코드를 구현하고 화면에 표시하라.

X’ = x + Tx

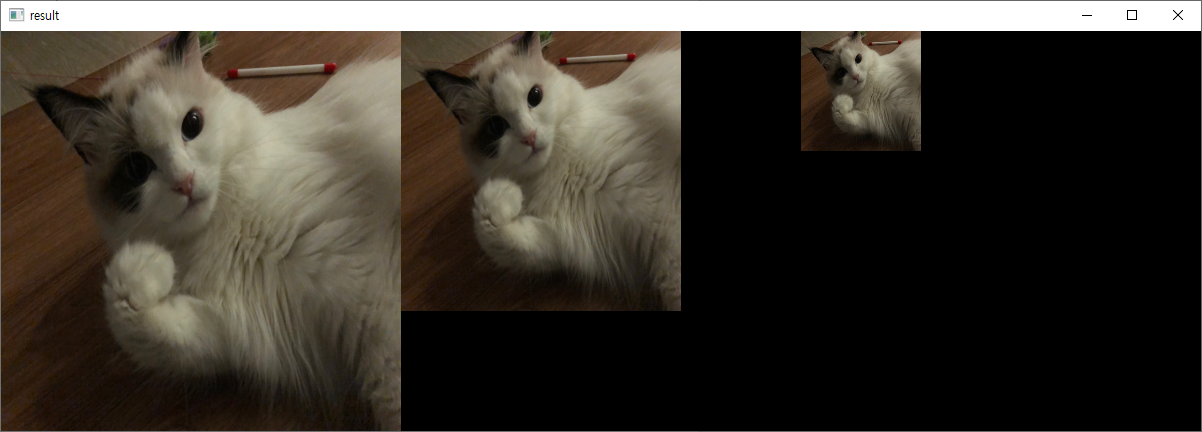
Y’ = y + Ty

* 코드
* import cv2  
  import numpy as np  
    
  img = cv2.imread('img/img1.png', cv2.IMREAD\_COLOR)  
    
  height, width = img.shape[:2]  
    
  KM = np.array([[1, 0, 150], [0, 1, 50]], dtype=float)  
    
  dt = cv2.warpAffine(img, KM, (width, height))  
    
  result = cv2.hconcat([img, dt])  
    
  cv2.imshow("result", result)  
  cv2.waitKey(0)



1. 다음의 행렬 확대 / 축소를 식을 참고하여 이미지를 로드하여 0.7배, 0.3배 축소하는 코드를 구현하고 화면에 표시하라

* 코드
* import cv2  
  import numpy as np  
    
  img = cv2.imread('img/img2.jpg', cv2.IMREAD\_COLOR)  
    
  *# 이미지 사이즈 조정*img = cv2.resize(img, dsize=(400, 400), interpolation=cv2.INTER\_AREA)  
    
  height, width = img.shape[:2]  
    
  scales = [1, 0.7, 0.3]  
    
  dst = []  
    
  for i, scale in enumerate(scales):  
    
   img\_zeros = np.zeros\_like(img, dtype=np.uint8) *# np.zeros\_like() 함수는 검정 화면 출력* tmp = cv2.resize(img, None, fx=scale, fy=scale) *# scale값 만큼 이미지 크기 줄임* height, width = tmp.shape[:2] *# resize()된 이미지 height, width 구함* img\_zeros[0: height, 0: width, :] = tmp  
    
   dst.append(img\_zeros)  
    
   if i == 2:  
   h1con = cv2.hconcat([dst[i - 2], dst[i - 1]])  
   h2con = cv2.hconcat([h1con, dst[i]])  
    
  cv2.imshow("result", h2con)  
  cv2.waitKey(0)



1. 다음의 행렬 회전을 참고하여 이미지를 로드하여 60도씩 회전하는 코드를 구현하고 화면에 표시하라. (1행 : 원본, 60도, 120도, 2행 : 180도, 240도, 300도)

* 코드

import cv2

import numpy as np

img = cv2.imread('img/img3.jpg', cv2.IMREAD\_COLOR)

# 이미지 사이즈 조정

img = cv2.resize(img, dsize=(400, 400), interpolation=cv2.INTER\_AREA)

height, width = img.shape[:2]

scale = 0.5

angles = [0, 60, 120, 180, 240, 300]

center\_x = width / 2

center\_y = height / 2

dt = []

for i, angle in enumerate(angles):

rotation\_matrix = cv2.getRotationMatrix2D((center\_x, center\_y), angle, 1)

dt1 = cv2.warpAffine(img, rotation\_matrix, (0, 0))

dt.append(dt1)

if i == 2:

h1con = cv2.hconcat([dt[i - 2], dt[i - 1]])

h1con\_1 = cv2.hconcat([h1con, dt[i]])

if i == 5:

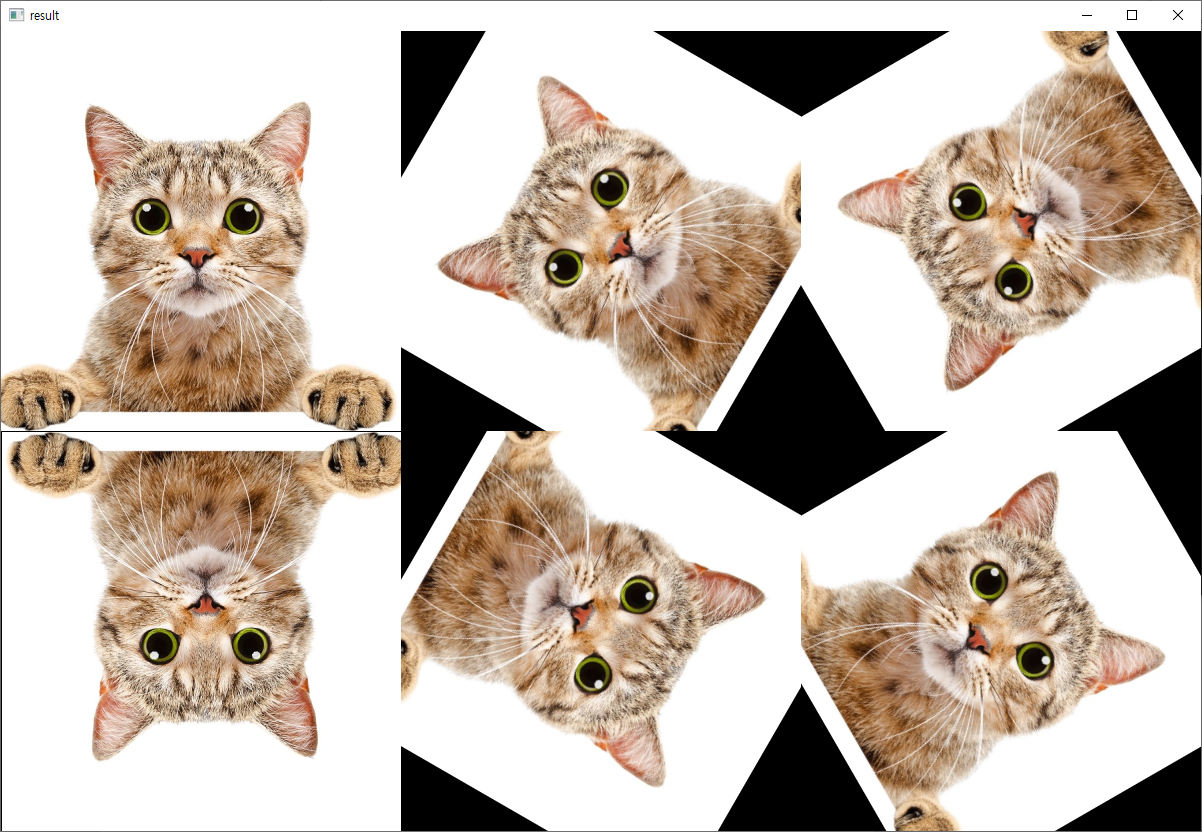
h2con = cv2.hconcat([dt[i - 2], dt[i - 1]])

h2con\_2 = cv2.hconcat([h2con, dt[i]])

result = cv2.vconcat([h1con\_1, h2con\_2])

cv2.imshow("result", result)

cv2.waitKey(0)



1. 위의 이동, 회전 확대 / 축소를 모두 적용하는 코드를 작성하고 이미지를 화면에 표시하라

* 코드
* import cv2  
  import numpy as np  
    
  img = cv2.imread('img/img4.jpg', cv2.IMREAD\_COLOR)  
  img = cv2.resize(img, dsize=(400, 400), interpolation=cv2.INTER\_AREA)  
  height, width = img.shape[:2]  
    
  center\_x = width / 2  
  center\_y = height / 2  
    
  scale = 0.7  
  img\_zeros = np.zeros\_like(img, dtype=np.uint8) *# np.zeros\_like() 함수는 검정 화면 출력*tmp = cv2.resize(img, None, fx=scale, fy=scale) *# scale값 만큼 이미지 크기 줄임*h, w = tmp.shape[:2] *# resize()된 이미지 height, width 구함*img\_zeros[0:h, 0:w, :] = tmp  
  h1con = cv2.hconcat([img, img\_zeros])  
    
  M = np.array([[1, 0, width\*0.15], [0, 1, height\*0.15]], dtype=float)  
  tmp2 = cv2.warpAffine(img\_zeros, M, (width, height))  
  rotation\_matrix = cv2.getRotationMatrix2D((center\_x, center\_y), 45, scale)  
  tmp3 = cv2.warpAffine(tmp2, rotation\_matrix, (width, height))  
  h2con = cv2.hconcat([tmp2,tmp3])  
    
  result = cv2.vconcat([h1con, h2con])  
    
  cv2.imshow("result", result)  
  cv2.waitKey(0)  
  cv2.destroyAllWindows()

