

1. 회전

- 회전(Rotate)은 선형 변환 중 하나에 포함되며, 회전 변환 행렬(Rotation matrix)을 통해 변환이 진행된다..
 - 임의의 행렬을 원점을 중심으로 회전시킨다.

$$\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

- 좌표 회전 행렬은 원점을 중심으로 좌푯값을 회전시켜 매핑하며, 좌표 축 회전 행렬은 원점을 중심으로 행렬 자체를 회전시켜 새로운 행렬의 값을 구성합니다.
- OpenCV의 회전 함수는 좌표 축의 회전 이동 행렬과 동일한 형태이며, 비율을 조정하거나 중심점의 기준을 변경하여 회전할 수 있습니다.

`matrix=cv2.getRotationMatrix2D((width/2,height/2),90,1)`

2*3 회전 행렬 생성 함수(`cv2.getRotationMatrix2D`)로 회전 변환 행렬을 계산한다.

`cv2.getRotationMatrix2D(중심점, 각도, 비율)`

중심점 : 회전의 기준점

각도 : 중심점을 기준으로 회전할 각도

비율 : 이미지의 확대 및 축소 비율

`rotate=cv2.warpAffine(image,matrix,(width,height))`

어파인 변환 함수(`cv2.warpAffine`)로 회전 변환을 계산한다.

`cv2.warpAffine(원본이미지, 어파인 맵 행렬, 출력 이미지 크기)`

어파인 맵 행렬 : 회전 행렬 생성 함수에서 반환된 매핑 변환 행렬을 사용

출력이미지크기 : 출력이미지의 너비와 높이

어파인변환?

- 선형 변환에 이동 변환까지 포함된 변환이다.
- 선의 수평성을 유지하며, 변환 전의 서로 평행한 선은 변환 후에도 평행함을 의미한다.
- 즉 길이의 비와 평행성이 보존되는 변환이다. ex) 사각형을 평행사변형으로 변환하는 것
- 어파인 변환을 표현하는 방법은 2*3 행렬의 실수형에 회전, 크기, 이동 정보가 들어가 있다.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad \begin{matrix} 2 \times 3 \text{ matrix} \\ (6 \text{ DOF}) \end{matrix}$$

- 어파인 변환은 미지수가 6개 이기 때문에 6개의 식이 필요하며, 점 3개가 어디로 이동했는지 알면 행렬로 표시할 수 있다.

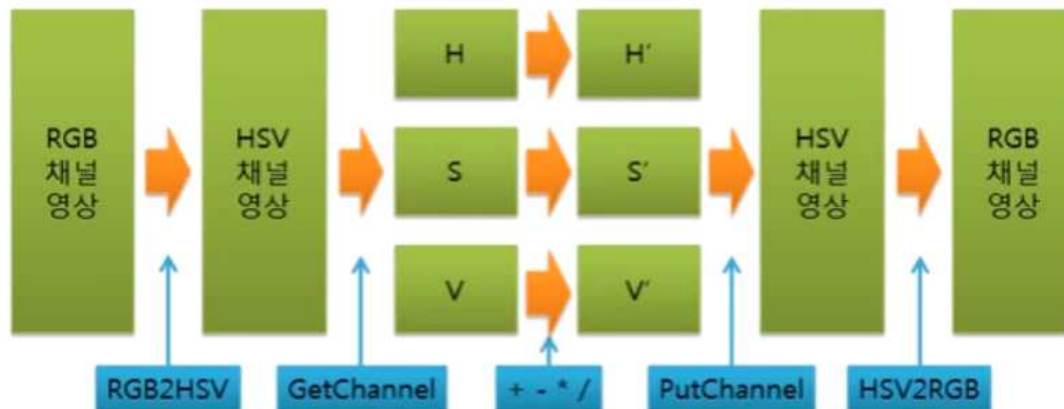


- OpenCV에서 점 3개의 이동 전, 이동 후 좌표를 입력하면 어파인 변환 행렬을 반환하는 함수를 제공한다.

함수 식 → `cv2.getAffineTransform(src,dst)`

- src : 3개의 원본 좌표점 ex) `np.array([[x1,y1],[x2,y2],[x3,y3]], np.float32)`
- dst : 3개의 결과 좌표점 ex) `numpy.ndarray.shape=(3,2)`

2. 채도



- OpenCV는 BGR기반으로 이미지를 출력해주므로 BGR에서 HSV로 바꿔준다.
- `cv2.split(hsv)` : h,s,v채널로 각각 나눠준다.
- 그 후 `cv2.add(image1,image2)`를 사용해서 s채널에 일정한 값을 더해준다.
- `cv2.merge(h,s,v)` : 나눠진 h,s,v 채널을 다시 합쳐준다.