

제5장 기하학적 처리

영상 처리

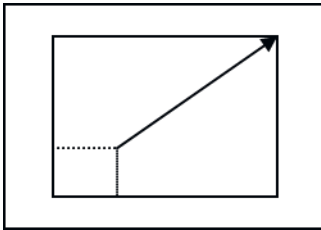
청주대학교 대학원 전자공학과
한철수

목차

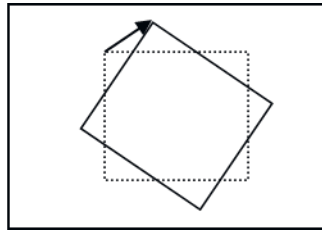
- 확대
- 축소
- 회전
- 대칭
- 실습

기하학적 처리

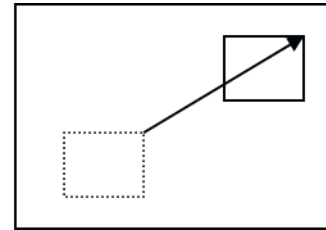
- 기하학적 변환에 의하여 픽셀들의 위치를 변경하는 처리



확대/축소



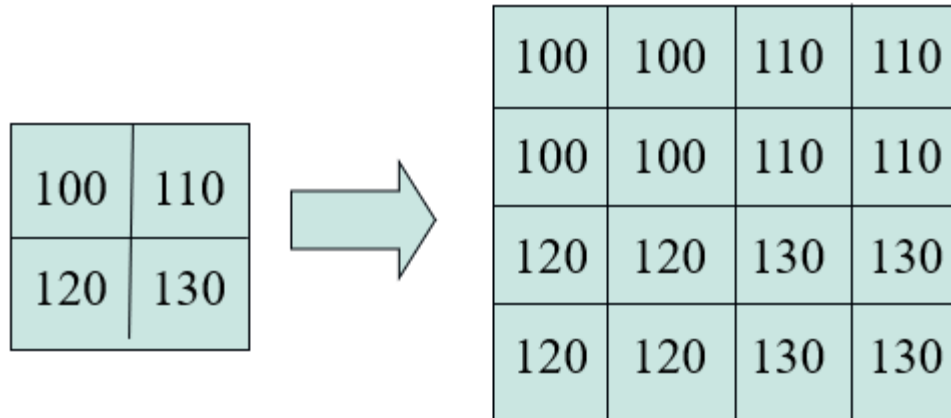
회전



이동

픽셀 복제 방법을 이용한 확대

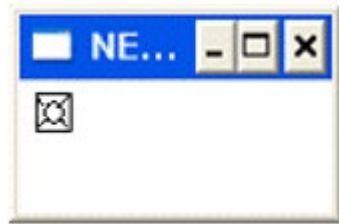
- 배율만큼 픽셀을 반복적으로 복사함
- 확대 배율이 정수일 경우에 사용 가능함



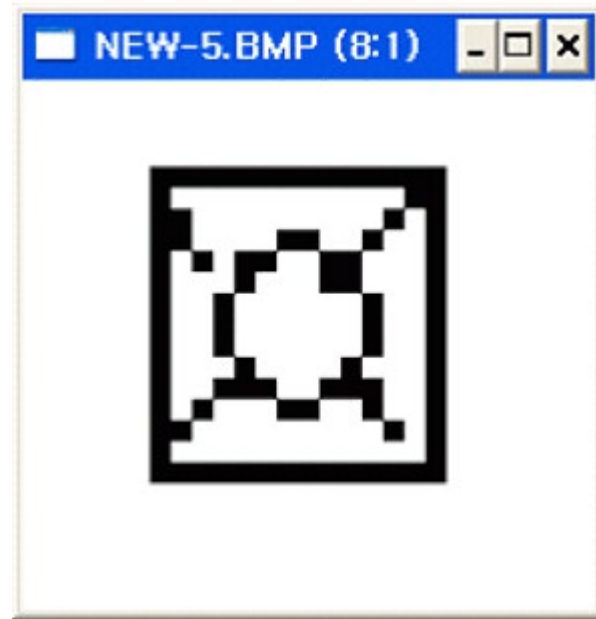
확대 배율이 2일 때

픽셀 복제 방법의 활용

- 픽셀 단위의 세밀한 편집에 활용



입력 영상



영상을 8배 확대하여 편집하는 화면

픽셀 복제 방법의 단점

- 확대한 영상에 계단 현상이 발생함
 - 영상이 매끄럽지 못함
- 확대 배율이 정수배가 아닌 실수배인 경우 처리가 어려움



입력 영상



확대 영상

역방향 사상과 양선형 보간법을 이용한 확대

- 확대 결과 영상에서 계단 현상을 줄일 수 있음
- 확대 비율이 실수배인 경우에도 처리가 가능함



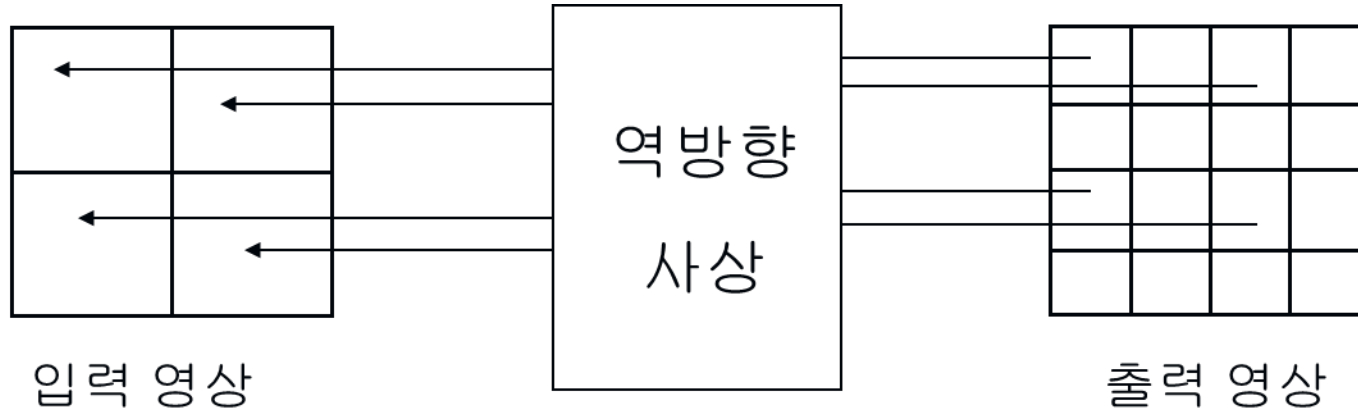
픽셀 복제 방법



역방향 사상과 양선형 보간법

역방향 사상

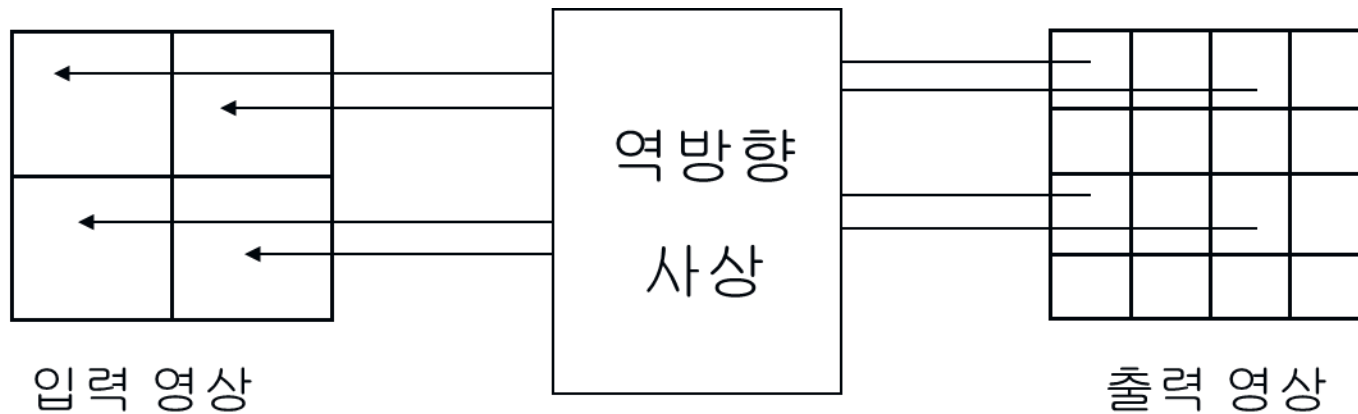
- 확대 비율이 2배인 경우의 역방향 사상



$$x_{source} = \frac{x_{dest}}{2}, \quad y_{source} = \frac{y_{dest}}{2}$$

양선형 보간법

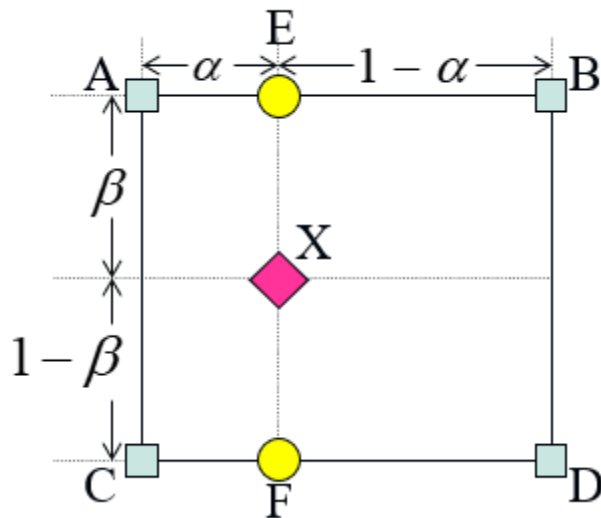
- 역방향 사상의 결과가 정수가 아닐 수 있음
 - 출력 영상의 픽셀 (1,1)은 입력 영상의 픽셀 (0.5, 0.5)로 사상 됨
 - 픽셀 (0.5, 0.5)가 입력 영상에 존재하지 않을 때, (0,0), (0,1), (1,0), (1,1)의 주변 픽셀 4개를 보간함



$$x_{source} = \frac{x_{dest}}{2}, \quad y_{source} = \frac{y_{dest}}{2}$$

입력 영상의 픽셀 값 계산 방법

- 역방향 사상에 의해 계산된 점 주위의 4개의 픽셀들의 값에 가중치를 곱하여 합함



□ 원시 화소

● X축 보간 화소

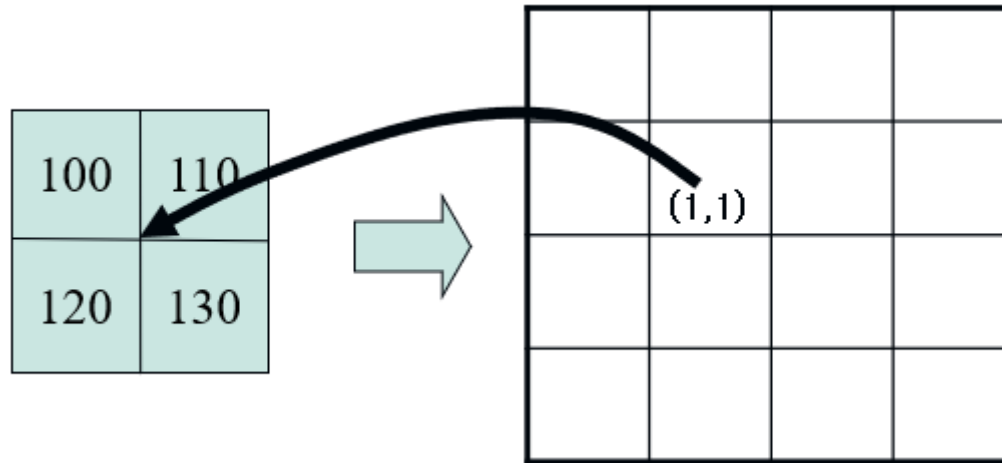
◆ 최종 보간 화소

$$E = (1 - \alpha)A + \alpha B = A + \alpha(B - A)$$

$$F = (1 - \alpha)C + \alpha D = C + \alpha(D - C)$$

$$X = (1 - \beta)E + \beta F = E + \beta(F - E)$$

양선형 보간법의 적용 예



$I(0.5, 0.5)$ 의 주위 4 픽셀: $I(0,0)=100$, $I(0,1)=110$, $I(1,0)=120$, $I(1,1)=130$

가로 방향 보간: $E(0, 0.5) = I(0,0)*0.5 + I(0,1)*0.5 = 100*0.5 + 110*0.5 = 105$

$F(1, 0.5) = I(1,0)*0.5 + I(1,1)*0.5 = 120*0.5 + 130*0.5 = 125$

세로 방향 보간: $O(0.5, 0.5) = E(0, 0.5)*0.5 + F(1, 0.5)*0.5 = 105 * 0.5 + 125 * 0.5$
 $= 115$

서브샘플링을 이용한 축소

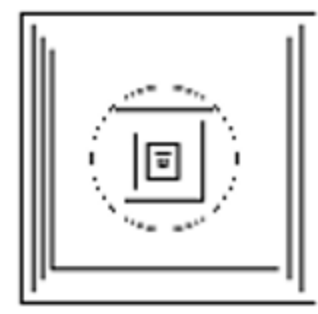
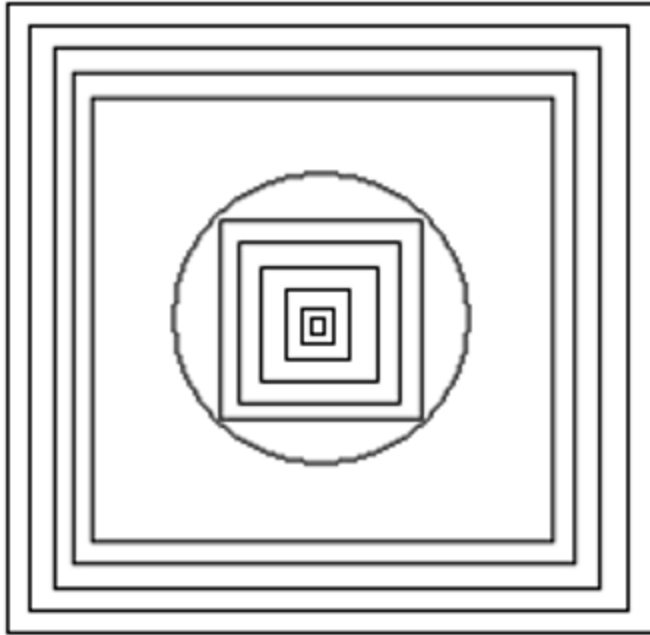
- 축소 배율만큼 건너뛰면서 픽셀 값을 취하는 방법
- 예
 - 가로 세로 방향으로 2배씩 축소
 - 모든 짝수 행과 짝수 열의 픽셀을 출력 영상에 복사
- 문제점
 - 영상의 세밀한 정보를 잃어버릴 수 있음
- 해결책
 - 영상을 흐리게 한 후에 서브샘플링을 수행

서브샘플링을 이용한 축소 예



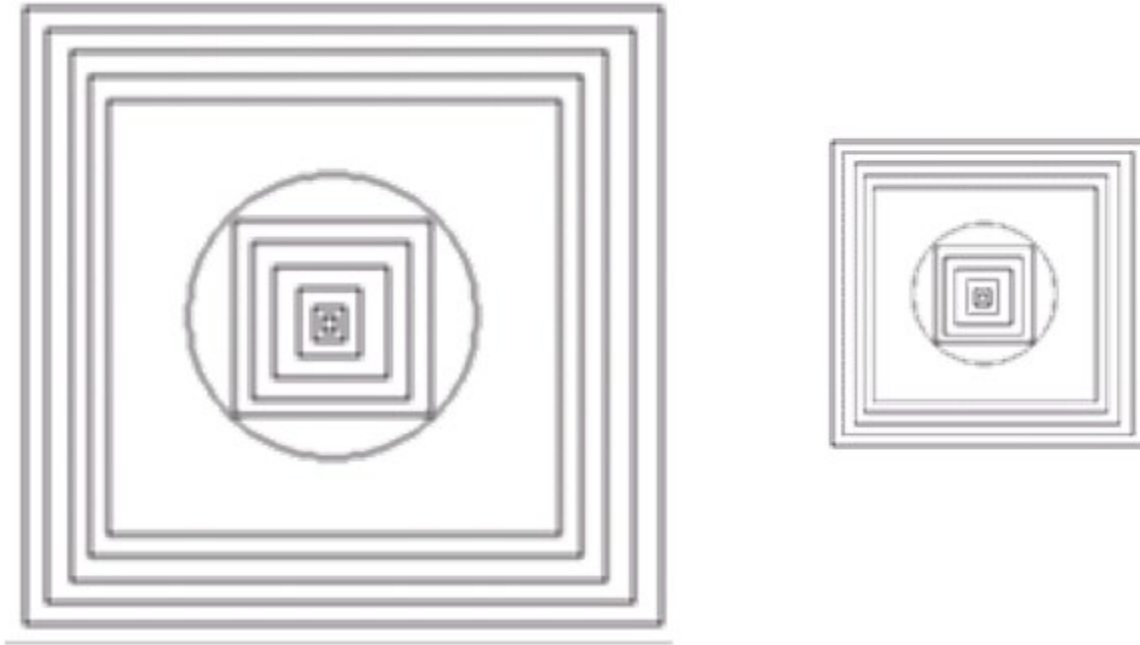
서브샘플링을 이용한 축소의 문제점

- 영상의 세밀한 정보를 잃어버릴 수 있음



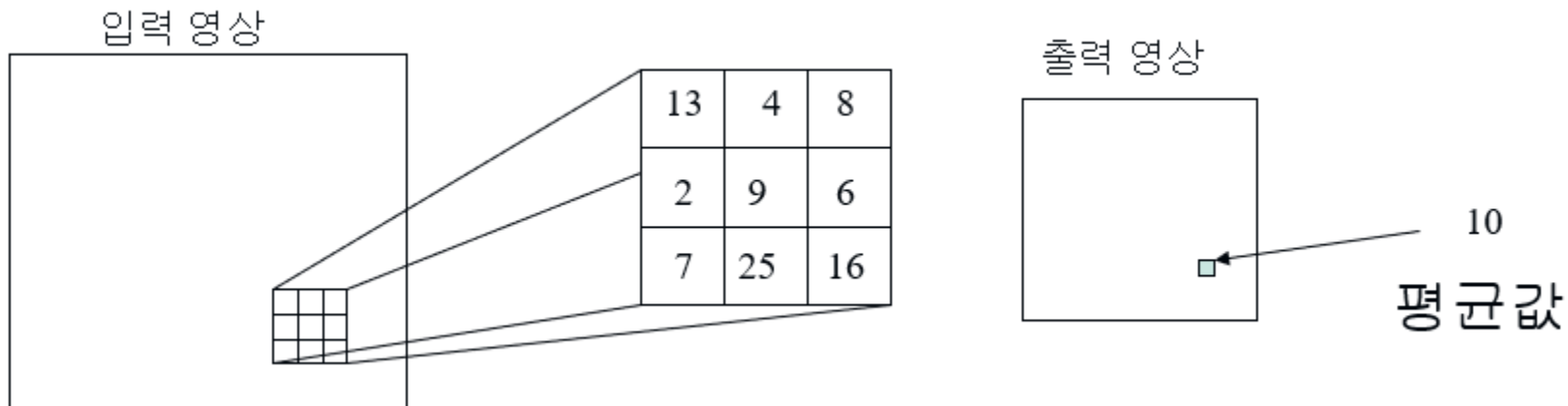
해결책

- 영상을 흐리게 한 후에 서브샘플링을 수행

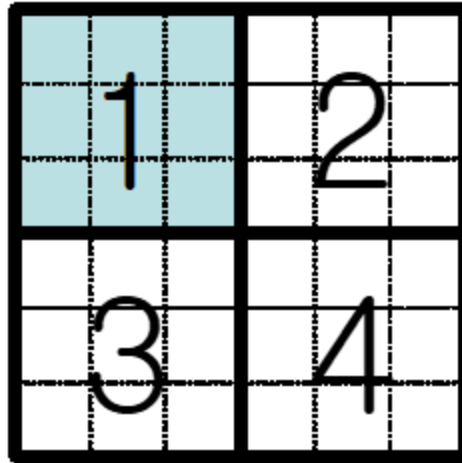


평균값 필터링을 이용한 축소

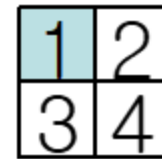
- 블록을 평균값으로 대체



평균값 필터링을 이용한 축소 예 (1/2)

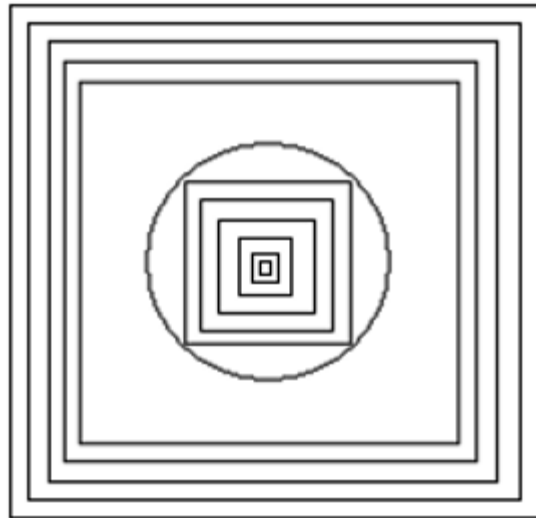


입력 영상



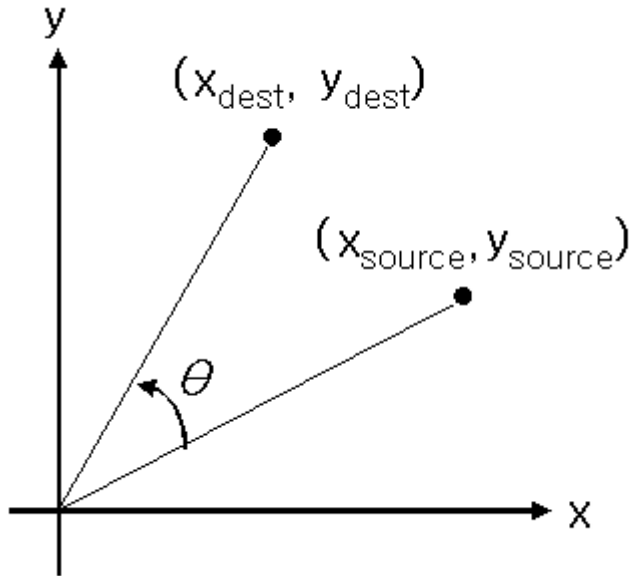
출력 영상

평균값 필터링을 이용한 축소 예 (2/2)



회전

- 원점을 중심으로 점 (x_{source}, y_{source}) 를 반시계 방향으로 θ 만큼 회전하여 점 (x_{dest}, y_{dest}) 를 얻음



$$x_{dest} = x_{source} \cos \theta - y_{source} \sin \theta$$

$$y_{dest} = x_{source} \sin \theta + y_{source} \cos \theta$$

$$\begin{bmatrix} x_{dest} \\ y_{dest} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_{source} \\ y_{source} \end{bmatrix}$$

회전의 고려 사항 (1/4)

- 전방향 사상을 이용한 회전
 - 출력 영상에서 픽셀 값을 할당 받지 못한 빈 곳이 발생함



전방향 사상 이용

회전의 고려 사항 (1/4)

- 역방향 사상을 이용하여 문제 해결
 - 원점을 중심으로 점 (x_{dest}, y_{dest}) 를 시계 방향으로 θ 만큼 회전하여 점 (x_{source}, y_{source}) 를 얻음
 - 출력 영상에서 픽셀 값을 할당 받지 못한 빈 곳이 발생함

$$x_{source} = y_{dest} \sin \theta + x_{dest} \cos \theta$$

$$y_{source} = y_{dest} \cos \theta - x_{dest} \sin \theta$$

$$\begin{bmatrix} x_{source} \\ y_{source} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_{dest} \\ y_{dest} \end{bmatrix}$$

회전의 고려 사항 (2/4)

- 회전의 중심

- 앞에서 사용한 수식은 원점을 기준으로 회전하는 식임
- 결과 영상의 표시 범위가 입력 영상과 동일할 경우, 회전에 의해 표시 범위를 벗어나서 보이지 않게 되는 부분이 많이 발생함
- 영상의 중심점을 기준으로 회전함으로써 문제를 해결



입력 영상



원점 기준 회전



영상의 중심점 기준 회전

회전의 고려 사항 (2/4)

- 영상의 중심점 (C_x, C_y)를 기준으로 하는 회전

$$x_{source} = (y_{dest} - C_y) \sin \theta + (x_{dest} - C_x) \cos \theta + C_x$$

$$y_{source} = (y_{dest} - C_y) \cos \theta - (x_{dest} - C_x) \sin \theta + C_y$$

$$\begin{bmatrix} x_{source} \\ y_{source} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_{dest} - C_x \\ y_{dest} - C_y \end{bmatrix} + \begin{bmatrix} C_x \\ C_y \end{bmatrix}$$

회전의 고려 사항 (3/4)

- 화면 좌표계와 수학 좌표계의 차이
 - 화면 좌표계에서는 좌측 상단이 원점임
 - 수학 좌표계에서는 좌측 하단이 원점임
- 앞에서 사용한 수식은 수학 좌표계를 이용한 수식이기 때문에 영상이 반시계 방향으로 회전하지 않고 시계 방향으로 회전함.
 - 화면 좌표를 수학 좌표로 변환한 다음에 회전 시키고, 그 결과를 다시 화면 좌표로 변환하여 문제를 해결함

회전의 고려 사항 (3/4)

- 좌표계 변환과 회전 결과
 - 반시계 방향으로 30도 회전



입력 영상



좌표계 변환을 하지 않은 경우



좌표계 변환을 한 경우

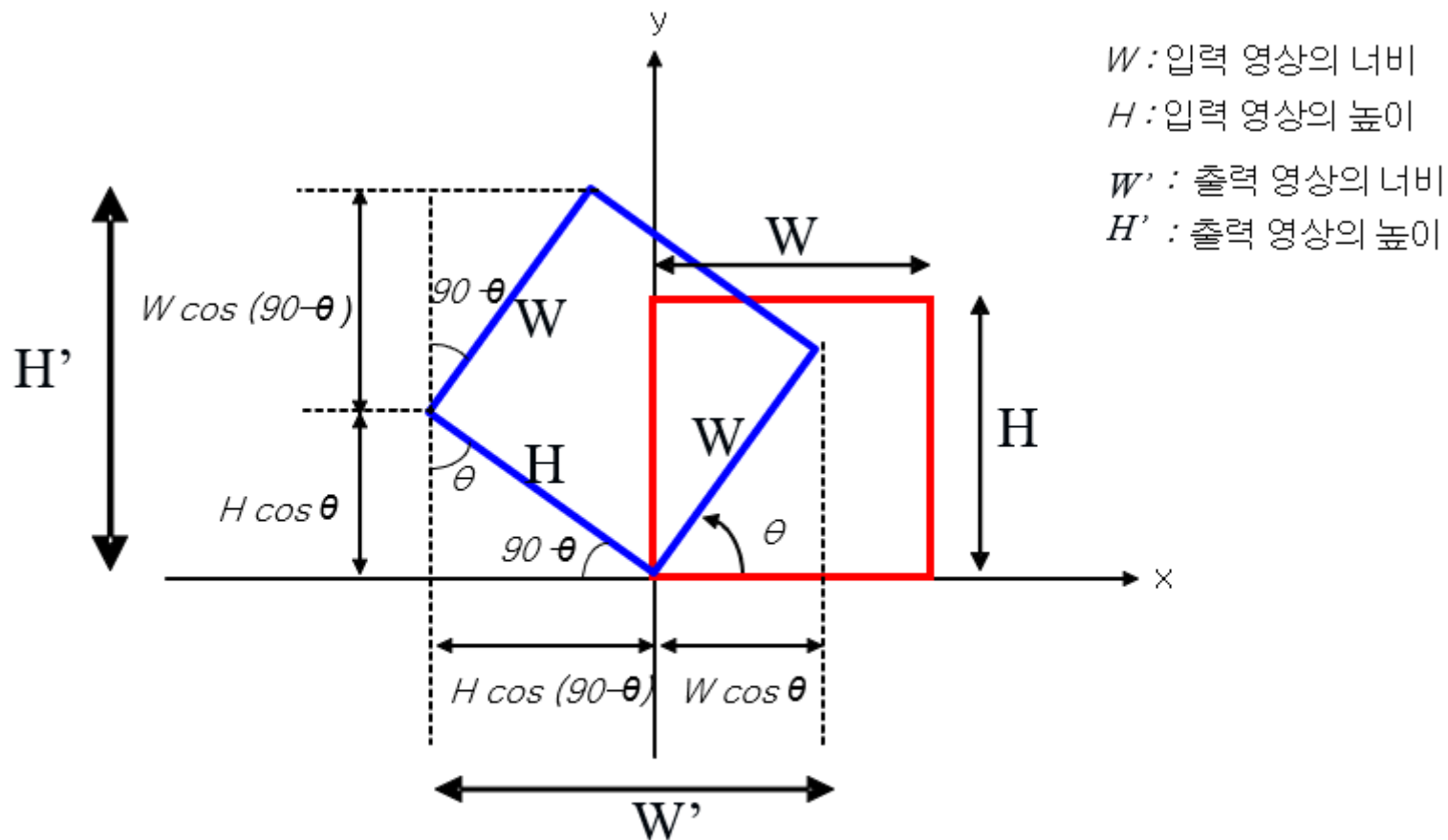
회전의 고려 사항 (4/4)

- 출력 영상의 크기

- 입력 영상과 출력 영상의 크기를 같게 할 경우, 출력 영상에서 잘려 나가는 부분이 발생함
- 출력 영상의 크기를 미리 계산하여 역방향 사상을 적용해야 함
- 출력 영상의 크기는 회전 각도에 따라서 달라짐

회전의 고려 사항 (4/4)

- 출력 영상의 크기 계산 방법



$$W' = H \cos(90-\theta) + W \cos \theta$$
$$H' = H \cos \theta + W \cos(90-\theta)$$

회전의 고려 사항 (4/4)

- 출력 영상의 크기를 고려한 회전



입력 영상



출력 영상

대칭



입력 영상



좌우 대칭



상하 대칭

흑백 영상의 대칭 변환의 구현

- 좌우 대칭

```
for (y = 0; y < imageHeight; y++)  
    for (x = 0; x < imageWidth; x++)  
        resultImg[y][x] = inputImg[y][imageWidth - 1 - x];
```

- 상하 대칭

```
for (y = 0; y < imageHeight; y++)  
    for (x = 0; x < imageWidth; x++)  
        resultImg[imageHeight - 1 - y][x] = inputImg[y][x];
```

질문

Q&A