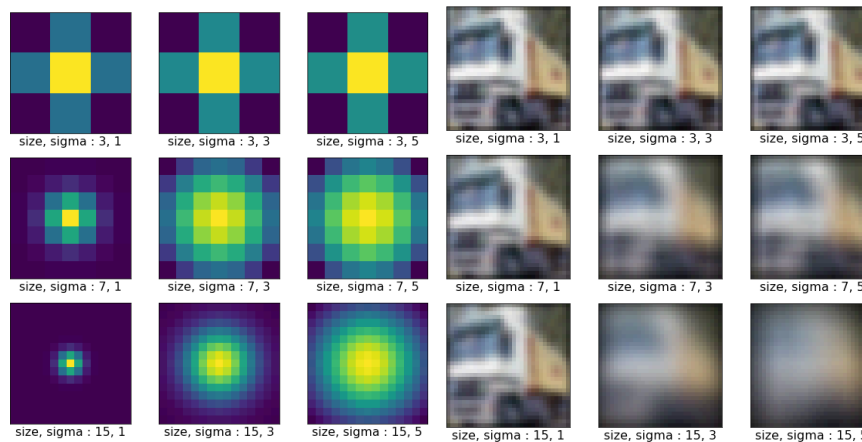


Short Analysis

가우시안 필터는 시그마 값이 작을수록 필터 중앙에 위치한 픽셀에 가중치를 많이 부여합니다. 시그마 값을 키울수록 하나의 픽셀에 유효한 영향을 줄 수 있는 중앙지점의 픽셀영역이 커집니다. 또한 필터 사이즈를 늘리는 것 역시 유사한 효과를 주는 것이 가능합니다. 그래서 시그마 값과 필터사이즈를 늘릴 경우 전체 이미지가 뿌옇게 변하는 모습을 확인할 수 있었습니다. 그리고 컨볼루션 연산 시 패딩을 넣었기 때문에 가장자리가 검게 변하는 것을 확인했습니다.

추가로 channelwise 로 convolution 을 세번 하는 것과, (w, h, in_ch=3, out_ch=3) 컨볼루션을 한번만 하는 것 중 어떤것이 효과적일지 궁금하여 테스트를 해 보았습니다. Channelwise 로 하는 것이 파라미터 용량이 9 배나 적어서 시간은 얼마나 유리할지 궁금했는데, 수행속도는 컨볼루션을 한번만 하는 것이 더 빨랐습니다. 그리고 추가로 강의시간에 배운내용인, 가우시안 필터를 여러번 거치면, 시그마가 더 큰 가우시안 필터를 씌운것과 결과가 비슷한 것을 실험을 통해 확인했습니다.

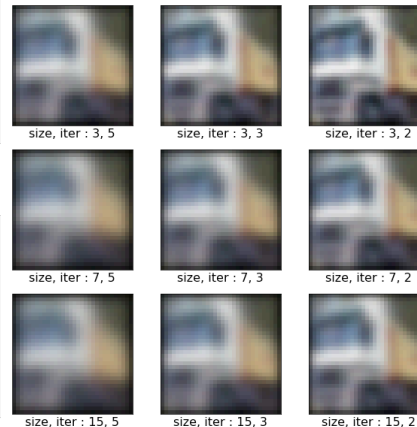
Results



a) Single Convolution

```
start_t = time.time()
rows = len(sizes)
cols = len(sigmas)
for r, size in enumerate(sizes):
    for c, sigma in enumerate(sigmas):
        idx = r*cols + c + 1
        filter = create_gaussian_filter(size, sigma)
        result = tf.nn.conv2d(images, filter, strides=[1, 1, 1, 1], padding="SAME")
end_t = time.time()
print(end_t - start_t)
```

2.8076422214508057



b) Three Channelwise Convolutions

```
start_t = time.time()
rows = len(sizes)
cols = len(sigmas)
for r, size in enumerate(sizes):
    for c, sigma in enumerate(sigmas):
        idx = r*cols + c + 1
        filter = create_gaussian_filter_channelwise(size, sigma)
        result_r = tf.nn.conv2d(images[:, :, :, 1], filter, strides=[1, 1, 1, 1], padding="SAME")
        result_g = tf.nn.conv2d(images[:, :, :, 2], filter, strides=[1, 1, 1, 1], padding="SAME")
        result_b = tf.nn.conv2d(images[:, :, :, 3], filter, strides=[1, 1, 1, 1], padding="SAME")
        result = np.concatenate((result_r, result_g, result_b), axis=3)
end_t = time.time()
print(end_t - start_t)
```

4.359415054321289