

영상처리 실습 보고서

9주차: Discrete cosine transform (DCT)

학번	201802170
이름	하 상 호

1. 과제의 내용

DCT 수행 및 mask 시각화

2. 과제의 해결 방법

과제의 내용을 해결하기 위해 어떠한 방법을 사용했는지, 자세하게 기술한다.

```
for v_ in range(v):
    for u_ in range(u):
        mask = block * np.cos(((2 * x + 1) * u_ * np.pi) / (2 * n)) * np.cos(((2 * y + 1) * v_ * np.pi) / (2 * n))

        temp1 = np.cos(((2 * x + 1) * u_ * np.pi) / (2 * n)) * np.cos(((2 * y + 1) * v_ * np.pi) / (2 * n))

        dct_mask_list.append(temp1)

        dst[v_, u_] = C(u_, n) * C(v_, n) * np.sum(mask)

return dst, dct_mask_list
```

공간도메인에서 주파수 도메인으로 바꿀 때 JPEG에서는 DCT를 이용해서 바꾼다.
DCT는 Discrete cosine transform 의 약자로 손실 압축하는 과정이다.
본 과정을 수행하기 위해 공식을 이용한다.

$$F(u, v) = C(u)C(v) \sum_{y=0}^{n-1} \sum_{x=0}^{n-1} f(x, y) \cos\left(\frac{(2x+1)u\pi}{2n}\right) \cos\left(\frac{(2y+1)v\pi}{2n}\right)$$
$$C(w) = \begin{cases} \sqrt{1/n} & \text{if } w = 0 \\ \sqrt{2/n} & \text{otherwise} \end{cases}$$

`def dct_block(block, n=8):` 함수에서 위 과정을 구현한 것이 위의 코드 사진이다.

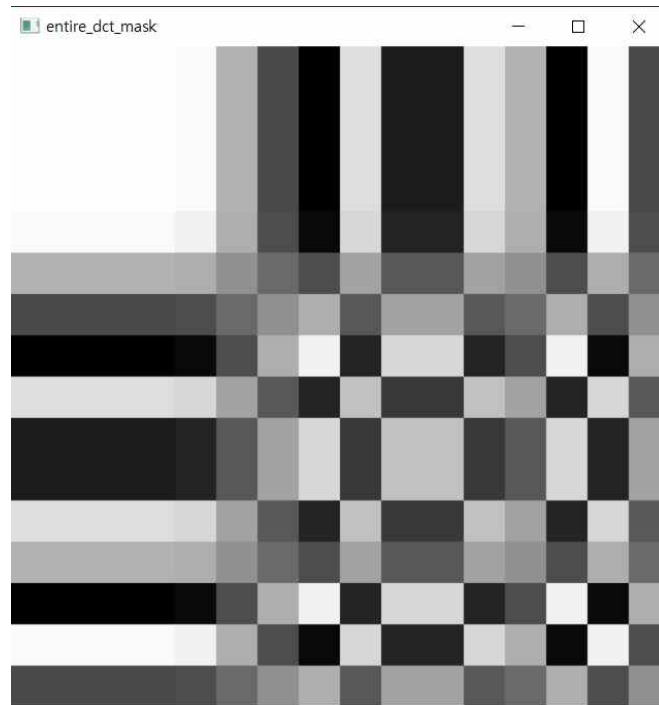
사진의 시각성을 위해 block을 제거 하였을 때 과제의 해당하는 올바른 사진이 나온 것을 확인 할 수 있었다.

또한 block을 추가 하여 src 값과 dct 의 값이 올바르게 나온 것을 확인 할 수 있었다.

여기서 block 은 $f(x,y)$ 에 해당하며, $dst[v_, u_]$ 는 $C(u)C(v)$ 시그마 (값들의 합) 을 나타 낸다.

3. 결과물

결과물이 잘 보이도록 화면을 캡처해 보고서에 올린다.



```
C:\Users\USER\anaconda3\python.exe "C:/Users/USER/Desktop/cnu/2022_1호
[[0.53731403 0.47336688 0.50497433 1.
 [0.60318716 0.71475572 0.60758565 0.663362
 [0.62448659 0.62549253 0.51641143 0.
 [0.80377561 0.1795282 0.3361258 0.77435916]]
[[ 2.2412 0.0466 0.2621 0.0086]
 [ 0.2491 -0.3091 -0.102 -0.1901]
 [ 0.0635 -0.3806 0.5487 -0.0029]
 [-0.2116 0.2147 -0.2189 0.0621]]
[[255 255 255 255 244 173 71 0 216 28 28 216 173 0 244 71]
 [255 255 255 255 244 173 71 0 216 28 28 216 173 0 244 71]
 [255 255 255 255 244 173 71 0 216 28 28 216 173 0 244 71]
 [255 255 255 255 244 173 71 0 216 28 28 216 173 0 244 71]
 [244 244 244 244 235 169 75 9 209 35 35 209 169 9 235 75]
 [173 173 173 173 169 141 103 75 158 86 86 158 141 75 169 103]
 [ 71 71 71 71 75 103 141 169 86 158 158 86 103 169 75 141]
 [ 0 0 0 0 9 75 169 235 35 209 209 35 75 235 9 169]
 [216 216 216 216 209 158 86 35 188 56 56 188 158 35 209 86]
 [ 28 28 28 28 35 86 158 209 56 188 188 56 86 209 35 158]
 [ 28 28 28 28 35 86 158 209 56 188 188 56 86 209 35 158]
 [216 216 216 216 209 158 86 35 188 56 56 188 158 35 209 86]
 [173 173 173 173 169 141 103 75 158 86 86 158 141 75 169 103]
 [ 0 0 0 0 9 75 169 235 35 209 209 35 75 235 9 169]
 [244 244 244 244 235 169 75 9 209 35 35 209 169 9 235 75]
 [ 71 71 71 71 75 103 141 169 86 158 158 86 103 169 75 141]]
```