

멀티미디어 시스템 Homework #4

▶ 제목: DCT Lossy Compression

▶ Due: 6월 10일(수요일) 까지

▶ 제출방법: LMS 과제제출에 HWP, Word 또는 PDF 보고서 파일 제출

- 파일: HWP, PDF, Word 보고서 1개
- 보고서에 입력영상 2개, 결과영상 $2 \times 3 = 6$ 개 출력, 그리고 source code 출력

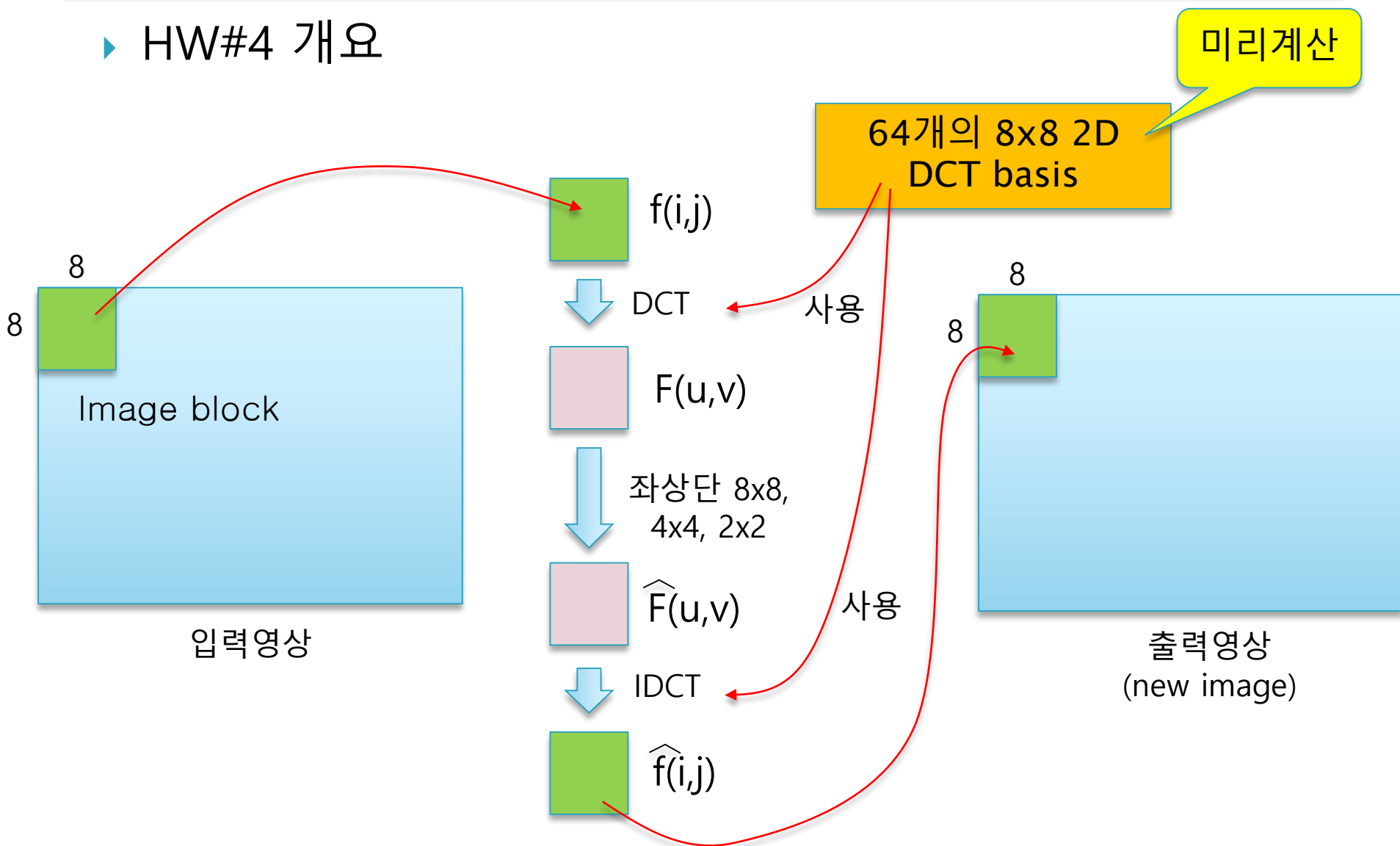
▶ 과제내용

- 수업에서 배운 DCT변환을 이용하여 영상의 lossy compression을 구현
- 영상 입력, 출력, 처리 등은 OpenCV 사용
- 구현 순서

- 1) 2D DCT에 사용될 8×8 DCT basis를 구한다. (전체 64개)
- 2) 입력영상을 read 하여 graylevel로 변환한다.
- 3) 영상의 8×8 영역(block)에 대하여 $f(i,j)$ 를 8×8 DCT basis 를 이용하여 8×8 주파수 공간 $F(u,v)$ 로 변환한다.
- 4) $F(u,v)$ 를 다시 IDCT로 8×8 영상을 다시 복원한다.
 - 이때, $F(u,v)$ 의 8×8 , 4×4 , 2×2 값들만 사용하여 비교
- 5) 영상 전체에 대하여 3)-4) 과정을 반복한다.
- 6) 입력 1장 영상에 대하여 복원된 영상 3장을 각각 저장한다.

멀티미디어 시스템 Homework #4

▶ HW#4 개요



멀티미디어 시스템 Homework #4

▶ 상세내용

1) 2D DCT에 사용될 8x8 DCT basis 구하기

- 2D basis 를 $B(i,j)_{uv}$ 라 하고 다음 식과 같이 표현됨

$$B(i,j)_{uv} = \frac{C(u)C(v)}{4} \cos \frac{(2i+1) \cdot u\pi}{16} \cdot \cos \frac{(2j+1) \cdot v\pi}{16}$$

- 여기서, $0 \leq i, j \leq 7$, 그리고 $C(\xi) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } \xi = 0, \\ 1 & \text{otherwise.} \end{cases}$
- 예를 들어 $B(i,j)_{00}$ 는 $u=0, v=0$ 이므로 i 와 j 를 이용하여 8x8 matrix를 만들어야함
- u 와 v 의 범위도 0에서 7까지이므로 2D DCT basis 는 모두 64개가 나옴
- 64개를 matrix를 각각 .txt 에 저장해도 되고 또는 프로그램내에서 array 에 저장함

멀티미디어 시스템 Homework #4

- 2) 너무 크지 않은 영상 (ex. Lena)을 read 하여 graylevel로 변환
- 3) 영상의 (0,0)부터 8x8 block 을 가져옴 -> $f(i,j)$
 - 8x8 크기의 $f(i,j)$ 에 $B(i,j)_{00}$ 를 element 끼리 곱하고 곱한 값들을 모두 더하면 $F(0,0)$ 가 됨
 - $f(i,j)$ 에 $B(i,j)_{10}$ 를 element 끼리 곱하고 곱한 값들을 모두 더하면 $F(1,0)$ 가 됨
 - 이 과정을 반복하여 $F(u,v)$ 를 모두 구함
 - 그러면 수업자료의 예와 같이 입력 8x8에 대한 $F(u,v)$ 가 계산됨을 확인할 수 있음 (아래 예와 같이 나오는지 확인)

imblock=

182	196	199	201	203	201	199	173
175	180	176	142	148	152	148	120
148	118	123	115	114	107	108	107
115	110	110	112	105	109	101	100
104	106	106	102	104	95	98	105
99	115	131	104	118	86	87	133
112	154	154	107	140	97	88	151
145	158	178	123	132	140	138	133



dctblock=1.0e+003*

1.0550	0.0517	0.0012	-0.0246	-0.0120	-0.0258	0.0120	0.0232
0.1136	0.0070	-0.0139	0.0432	-0.0061	0.0356	-0.0134	-0.0130
0.1956	0.0101	-0.0087	-0.0029	-0.0290	-0.0079	0.0009	0.0096
0.0359	-0.0243	-0.0156	-0.0208	0.0116	-0.0191	-0.0085	0.0005
0.0407	-0.0206	-0.0137	0.0171	-0.0143	0.0224	-0.0049	-0.0114
0.0072	-0.0136	-0.0076	-0.0119	0.0183	-0.0163	-0.0014	-0.0035
-0.0015	-0.0133	-0.0009	0.0013	0.0104	0.0161	0.0044	0.0011
-0.0068	-0.0028	0.0041	0.0011	0.0106	-0.0027	-0.0032	0.0016

멀티미디어 시스템 Homework #4

4) $F(u,v)$ 를 다시 IDCT로 8x8 영상을 다시 복원

- 위의 DCT과정을 다시 반복하되 $F(u,v)$ 에 basis의 element들을 곱하여 합하면 $f(i,j)$ 가 복원됨

$$\tilde{f}(i, j) = \sum_{u=0}^7 \sum_{v=0}^7 \frac{C(u)C(v)}{4} \cos \frac{(2i+1) \cdot u\pi}{16} \cdot \cos \frac{(2j+1) \cdot v\pi}{16} \cdot F(u, v)$$

- 아래 3가지 경우에 대하여 $f(i,j)$ 를 복원해 볼것

1) $F(u,v)$ 그대로 사용

2) $F(u,v)$ 좌상단 4x4는 그대로 두고 나머지는 0로 변경 사용

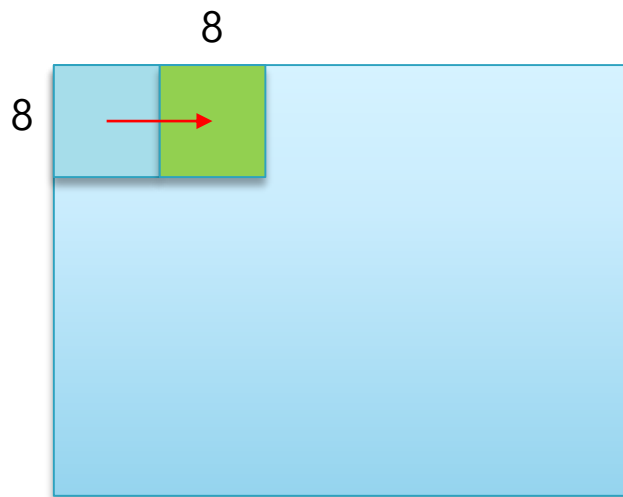
3) $F(u,v)$ 좌상단 2x2는 그대로 두고 나머지는 0로 변경 사용

- $F(u,v)$ 에 0이 많을수록 복원된 영상은 왜곡됨

멀티미디어 시스템 Homework #4

5) 위 과정을 영상의 모든 8x8 block 에 적용하여 영상 DCT 와 IDCT 수행

- 영상 block은 8 pixel 씩 건너뛰면서 수행



- 복원 결과영상을 마지막으로 file로 저장

멀티미디어 시스템 Homework #4

▶ 제출 할 것

- 1) Test 영상은 2장 (Lena 및 직접 찍은 영상)
 - 영상 크기는 x기준으로 1000이하
- 2) 각 test 영상마다 3개의 $f(i,j)$ 복원 결과
- 3) 8x8 DCT basis는 $v=0$ 일때 u 를 0~7까지 8개의 matrix를 보고서에 출력
- 4) 그리고 source code 출력