



Anti-aliasing

과목명	디지털신호처리
담당교수	심동규 교수님
학과	컴퓨터정보공학부
학년	3학년
학번	2019202009
이름	서여지
제출일	2021.11.30 (화)

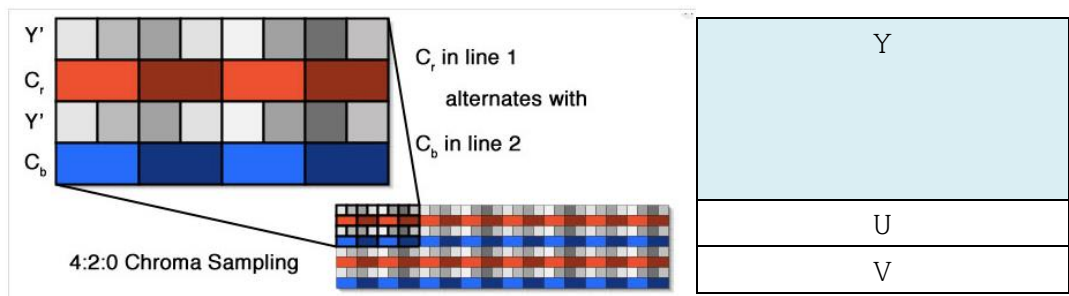
1. 과제 개요

이미지를 downsampling 한 뒤 upsampling 하는 프로그램을 작성한다. anti-aliasing filtering을 수행하여 antialiasing을 하지 않은 경우와 비교한다. 테스트 이미지는 832*480, 1920*1080의 해상도를 갖고, Y,U,V 채널로 구성되어 있다. Y채널은 한 픽셀당 하나의 값을 갖고, U와 V채널은 4개의 픽셀이 하나의 값을 갖는다. Y채널의 정보에 대해 antialiasing filter를 적용하고, 이미지의 크기를 반으로 downsampling한 뒤 다시 원래의 크기로 upsampling하는 프로그램을 작성한다.

2. 과제 수행 방법 (이론과 구현)

- YUV420 이미지 형식

이번 과제에서 사용하는 이미지 형식은 저번 3차과제에서 사용했던 RGB를 이용한 이미지와 다르게 Y,U,V 채널을 사용한다. Y채널은 명암 정보, U채널은 파란색 성분, V채널은 붉은색 성분의 값을 갖는다. 각 채널의 값의 비율에 따라 YUV444, YUV422, YUV420 등의 종류가 있다. YUV420의 경우 1픽셀에 해당하는 정보를 12bit를 사용하여 저장한다. Y채널의 명암 정보는 각 픽셀당 1byte의 정보를 갖고, U와 V채널은 4개의 픽셀에서 각각 1byte의 값을 공유한다.

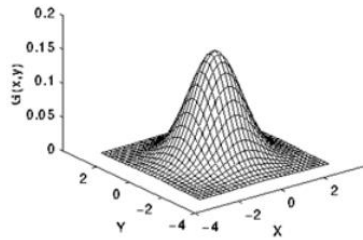


i

- Antialiasing

나이퀴스트의 샘플링 이론에서 어떤 신호를 샘플링 주파수 F_s 로 샘플링 했을 때, 이 신호는 최대 $1/2F_s$ 의 주파수를 갖는 신호로 복원될 수 있다. 따라서 원본 신호의 최대 주파수 F_{max} 의 두 배에 해당하는 주파수 $2F_{max}$ 가 F_s 보다 큰 경우에 샘플링 된 신호는 손상이 발생한다. 이때 발생한 손상을 aliasing이라 한다. aliasing을 방지하기 위해 low pass filter를 이용하여 원본 신호의 높은 주파수를 제거한다. 이번 과제에서는 Gaussian filter를 이용하

여 antialiasing을 수행한다. 과제에서 사용된 Gaussian filter는 5*5 크기의 window를 사용한다. 2차원 가우시안 분포의 식에서 분산을 1로 하여 필터 계수를 미리 계산하여 사용하였다.



$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp \frac{-(x^2+y^2)}{2\sigma^2}$$

ii

- Downsampling

샘플링 주파수를 감소시켜서 그 결과 이미지의 해상도를 줄이는 과정이다. 이번 과제에서는 너비와 높이를 각각 1/2 비율로 downsampling하여 원본 이미지에서 4개의 픽셀이 해상도가 줄어든 이미지에서 하나의 픽셀로 표현된다. 아래 그림에서 1,1에 해당하는 phase의 값을 선택하여 저장하는 방법으로 downsampling하였다.

(0,0)	(0,1)
(1,0)	(1,1)

- Upsampling

downsampling과 반대로 이미지의 해상도를 높이는 과정이다. 이번 과제에서는 6 tap DCT-IF를 이용하여 가로, 세로, 대각선에 대하여 각각 보간하였다. 미리 계산된 DCT-IF 계수를 6개의 값에 곱하고 합산하는 것으로 값을 보간한다.

- 구현방법

- YUV 채널 분리

YUV420 형식은 이미지의 한 픽셀당 12bit을 사용하여 정보를 저장한다. 4개의 픽셀에서 하나의 값을 공유하는 U,V 채널에 대해서는 별도의 처리를 하지 않고, Y채널의 값만 antialiasing, downsampling, upsampling 처리를 수행하였다. U,V 채널의 값은 복원된 Y채널 정보의 뒤에 복사하여 저장하였다.

- 프로그램 구조

프로그램은 파일을 읽는 부분, antialiasing, downsampling, upsampling을 수행하는 부분, MSE 계산과 그 결과를 출력하는 부분, 결과에 U,V채널의 값을 복사하고 파일에 쓰는 부분으로 나뉘어진다. input 디렉토리의 입력 파일을 읽어서 결과를 output 디렉토리에 파일로 저장한다. 각 파일명과 그 내용은 다음과 같다.

input	originalFile	입력파일명	입력영상
output	antialiasedFile	a_	가우시안 필터 적용 결과
	f_downsampledFile	df_	필터 적용 후 다운샘플링 결과
	downsampledFile	d_	필터 적용 없이 다운샘플링 결과
	f_reconstructedFile	rf_	필터 적용 후 복원 결과
	reconstructedFile	r_	필터 적용 없이 복원한 결과

◆ antialiasing

Y채널의 모든 픽셀에 대해 5*5 크기의 미리 계산한 필터계수와 곱을 계산하여 가우시안 필터를 적용한다. 이미지 범위를 벗어나는 값은 이미지의 테두리 값을 이용하여 padding하였다.

(0,0)	(0,0)	(0,1)
(0,0)	(0,0)	(0,1)
(1,0)	(1,0)	(1,1)

```
int filter[5][5] = { {1,4,7,4,1},
                     {4,16,26,16,4},
                     {7,26,41,26,7},
                     {4,16,26,16,4},
                     {1,4,7,4,1} };
```

◆ downsampling

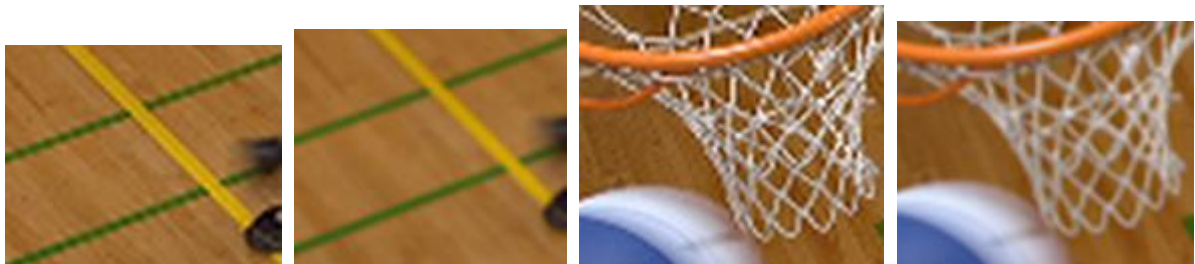
원본 크기와 같은 크기의 배열에 0과 이미지 값을 저장한다. 값이 저장되는 픽셀은 (1,1)에 해당하는 phase를 갖는다.

◆ upsampling

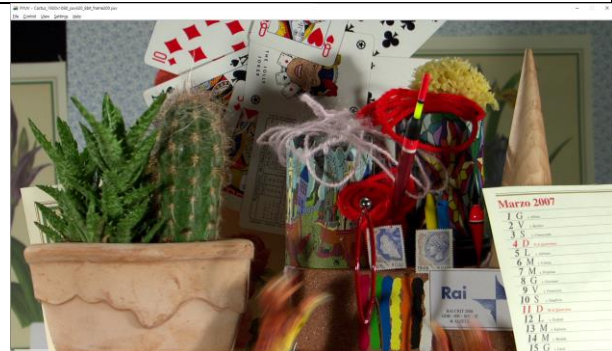
미리 계산된 6tab DCT 필터의 계수를 이용하여 horizontal, vertical, diagonal 방향으로 각각 upsampling한다. antialiasing과정과 마찬가지로 이미지 범위를 벗어나는 값은 이미지의 테두리 값을 이용하여 padding한다.

3. 결과

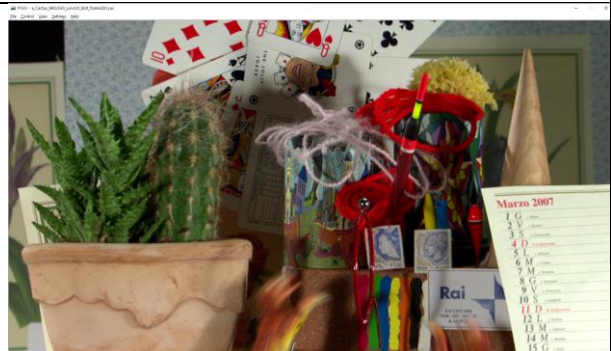




originalFile



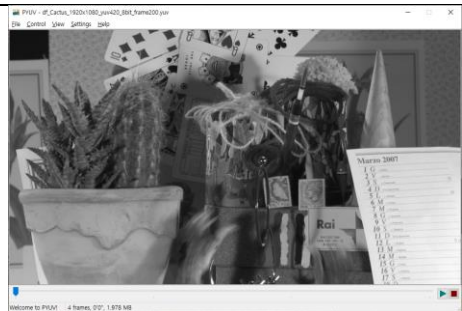
antialiasedFile



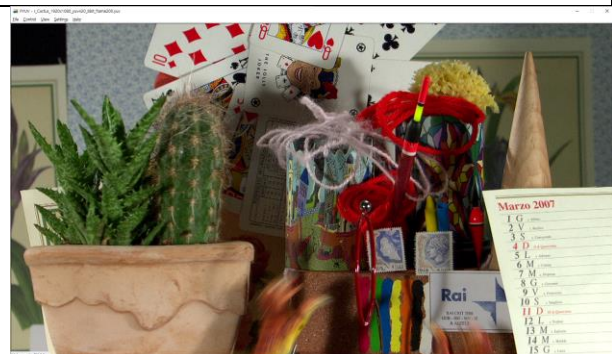
downsampledFile



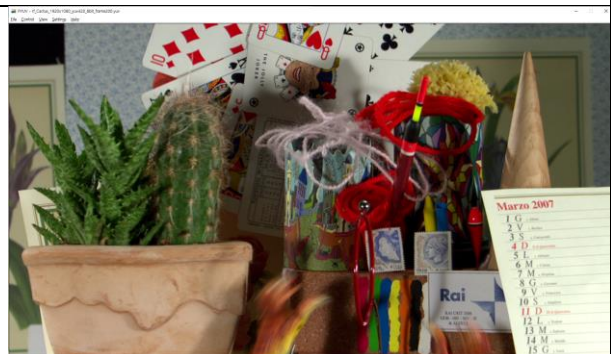
f_downsampledFile

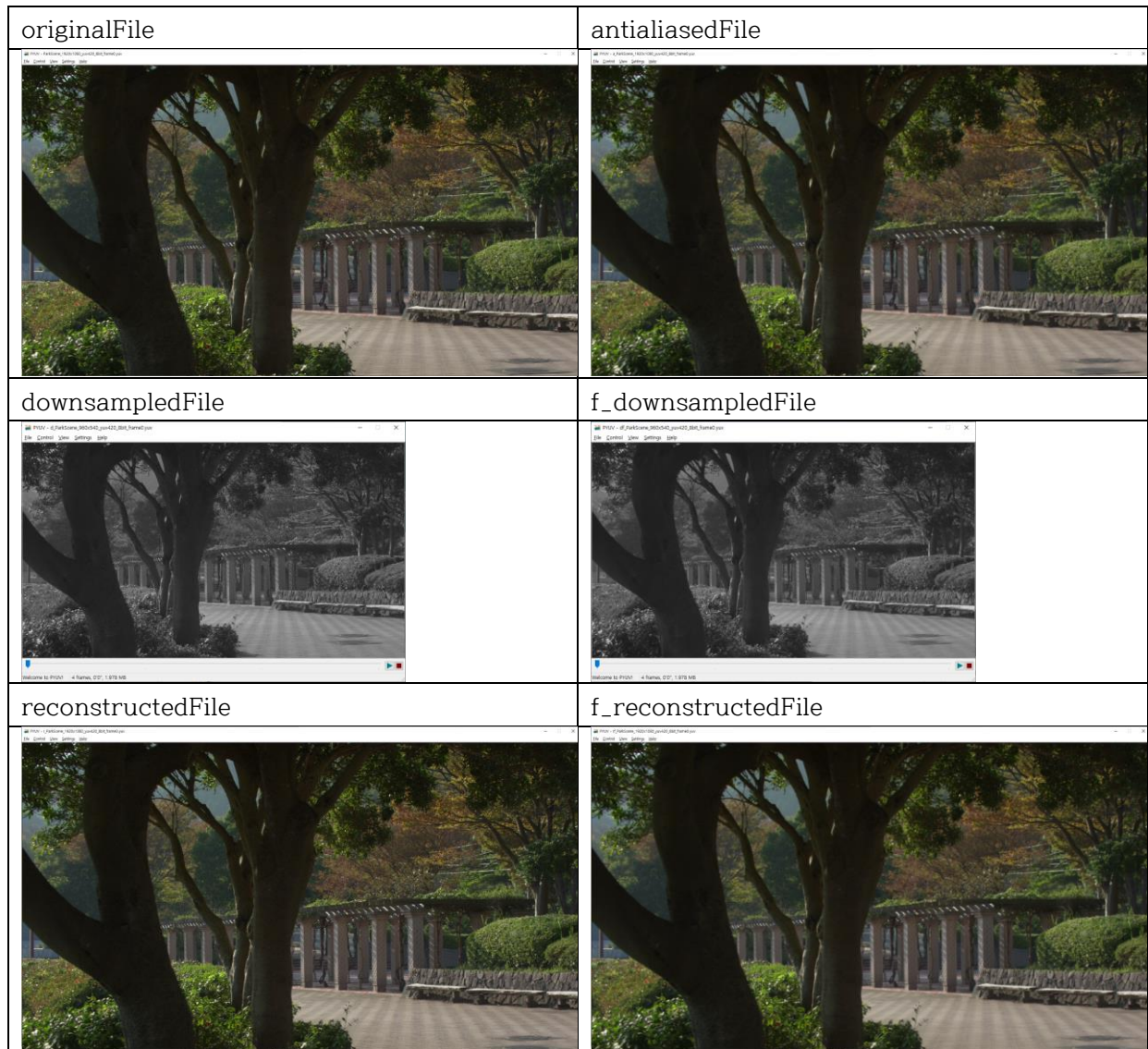


reconstructedFile



f_reconstructedFile





복원 결과를 눈으로 비교했을 때, 가우시안 필터를 적용한 이미지는 값이 급격하게 변화하는 부분에서 특히 흐리게 나타난 것을 확인할 수 있으며 필터를 적용하지 않은 이미지에서 aliasing에 의해 나타난 계단현상이 나타나지 않았다.

입력파일명	해상도	MSE: 필터없음	MSE: 필터적용
PartyScene_832x480_yuv420_8bit_frame120	832x480	295.072	243.272
BasketballDrill_832x480_yuv420_8bit_frame0	832x480	63.594	63.2402
Cactus_1920x1080_yuv420_8bit_frame200	1920x1080	37.8574	34.579
ParkScene_1920x1080_yuv420_8bit_frame0	1920x1080	19.911	25.8066

몇 가지 이미지의 antialiasing 수행 여부에 따른 MSE값은 위의 표와 같이 나타났다. PartyScene과 같이 육안으로도 큰 차이를 주며 MSE에서도 값의 차이가 큰 경우도 있었고, BasketballDrill과 같이 육안으로 차이가 느껴지지만 MSE는 크게 차이가 나지 않는 경우도 나타났다. 반면 ParkScene의 경우는 antialiasing필터를 적용한 결과 MSE가 더 크게 나타났다. 이미지를 비교했을 때 더 흐릿하게 보였다.

4. 고찰

anti-aliasing 과정을 통해 downsampling할 때 발생하는 aliasing을 피하는 과제를 진행하였다. 이번 과제에서 사용한 가우시안 필터는 상대적으로 적은 고주파 성분을 제거하고, 주파수 영역에서 에너지가 패스밴드로 모이도록 최적화한다. 그 결과 복원된 이미지에서 계단현상으로 나타나는 aliasing을 피한 것을 확인할 수 있었다. 신호 처리 결과를 눈으로 확인할 수 있는 기회가 되었다.

ⁱ YUV420 <https://imagej.tistory.com/150>

ⁱⁱ 가우시안 분포 <https://programmer.ink/think/opencv-learning-basic-image-manipulation-smooth-and-blur.html>