

# 디지털 신호처리 실습과제

---

제출마감일: 14주차 월요일 [5/31(월), 2021]

제출 내용: 보고서(결과물 포함/해설), simulation source (c/c++/matlab), 결과 그림파일  
보고서 및 simulation source[보고서 포함유무와 관계없이 별도 첨부], 결과 그림 파일들[마찬가지로 별도 첨부]를 압축하여 하나의 파일로 LMS에 업로드하기 바랍니다. 보고서에는 진행 과정 및 결과와 그에 대한 해석 및 설명이 충분히 포함되어야 합니다.

Delay 감점: 하루당 20%, 제출일 5일[만 4일] 이상 경과시 불인정 [0점 처리].

---

\* 첨부된 레나 이미지 (256x256 픽셀, grayscale)는 간단한 DSP 영상처리 및 과제를 위해서 널리 이용되는 이미지이다. 이에 다양한 FIR 기반 LPF/HPF를 적용한 영상 처리를 진행해보고자 한다 (LPF – Smoothing, HPF – Sharpening). 단 구현시 각 이미지에 적용할 FIR 필터는 내장함수가 아닌 for 문 등을 이용하여 구현해야 한다. 또한 1, 2번에서의 각각의 적용 결과에 대한 해석을 자유롭게 진행하고 (예: 교재의 그림 6-13과 같은 공간축 크기 비교, 푸리에 변환(fft)을 이용한 주파수 분석: 분석시에는 내장함수를 이용해도 무방), 해당 해석 및 설계 과정 (예: 차수는 왜 해당 값들로 결정하였는지), 결과, 코드 등을 보고서에 충분히 기술하라. 다른 유사한 효과를 낼 수 있는 FIR 필터 등을 찾거나 개발해서 적용할 수 있으면 그를 적용한 결과들을 추가적으로 포함하여도 좋다.

1. [FIR LPF Filtering] 원본 이미지에 대해 이동 평균 필터 + 지연 연산기를 이용한 LPF를 원본 이미지의 모든 행 (또는 열)에 적용한 결과물을 구한다. 이 때 두 개의 LPF를 고려하여 진행하며, 각 LPF의 결과(변환 이미지)가 충분히 서로 다르도록 필터의 차수를 스스로 결정하여 진행한다.

2. [FIR HPF Filtering] 원본 이미지에 대해 다음의 HPF를 적용한 결과물을 두 개 구한다. 이 때 두 개의 양의 정수 b 값을 고려하며, 결과가 충분히 달라질 수 있도록 b 값을 스스로 결정하여 진행한다.

$$\begin{aligned} y[m,n] &= 3x[m,n] - (x[m+1,n] + x[m-1,n] + x[m,n+1] + x[m,n-1]) \\ &\quad + \frac{1}{4}(x[m+1,n+1] + x[m+1,n-1] + x[m-1,n+1] + x[m-1,n-1]) \\ z[m,n] &= x[m,n] + by[m,n] \end{aligned}$$

y: 입력 x에 대한 edge-finding filter의 결과물 (y-x 관계식은 일반적인 에지 검출 (이미지 밝기가 급격히 변하는 가장자리 찾기)을 위한 FIR 필터의 차분 방정식)

z: edge-finding filter를 이용한 입력 x에 대한 sharpening 필터의 식

---

## [프로젝트 해설 1: 이미지 파일 다루기 (MATLAB)]

0. 영상 픽셀에서 인덱스 범위를 벗어난 경우는 if 문 등을 통해 범위를 벗어나는 항들만 제외한다. (예: 1차원 벡터에서 3개의 이동합 구할 때 출력  $y(2) = x(2) + x(1)$ 로  $x(0)$ 을 제외)

1. MATLAB에서 아래 명령을 통해 작업 폴더내의 원본 이미지 파일을 불러온다. (작업 폴더: MATLAB 메인 창내 왼쪽 윈도우 현재 폴더(current folder)). 이 때  $x$ ,  $map$ 의 변수명은 바뀔 수 있다. ( $x$ 는 입력신호,  $map$ 은 파일의 색상 정보를 나타냄)

```
[x, map] = imread('lena.png');
```

2. 입력신호  $x$ 는 2차원 배열의 형태로 얻어진다. 이에 대한 처리 과정을 진행하여 출력  $y$ 를 구할 때, 사전에 출력 값을 담을 변수형을 uint8로 지정하여 진행하라.

예)  $y = \text{uint8}(\text{zeros}(\text{size}(x)))$ ;

3. 필터링된 최종 출력이  $y$ 일 때, 이를 이용한 변환 이미지 생성은 다음과 같이 진행한다. 파일명은 바뀔 수 있으나, 확장자는 jpg로 진행한다.

```
imwrite(y, map, 'lena.jpg');
```