

멀티미디어 시스템 Homework #3

▶ Laplacian filtering

▶ Due: 4월 29일(수요일) 까지

▶ 제출방법: LMS 과제제출에 HWP 보고서 파일 제출

- 파일: HWP 보고서 1개
- 보고서에 입력영상 3개, 결과영상 3x5=15개 출력, 그리고 source code 출력

▶ 과제내용

- 수업에서 배운 Laplacian filtering 을 C/C++/Java/Python 중에서 구현
- 영상 입력, 출력, 필터링은 OpenCV 사용
- 구현 순서

1) 입력영상을 read 한다.

2) 영상을 흑백으로 변환한다.

3) Gaussian smoothing 으로 영상을 blur 한다. (잡음도 제거)

`void cv::GaussianBlur();`

4) Blur된 영상에 Laplacian filtering 적용한다.

`void cv::Laplacian();`

* 주의점: Laplace 연산은 값이 255를 넘기때문에 영상 type을 interger 로 사용 바람. (float으로 하면 image display가 안되어 다시 변환필요)

멀티미디어 시스템 Homework #3

5) Edge를 찾기위해 Zero crossing 실행

- 첨부파일에 FindZeroCrossings() 함수 사용

* 주의점: 이 함수의 입력영상은 float type 이고, 출력은 unsigned char임

- 혹시, Zero crossing 함수결과가 잘 나오지 않으면 아래 코드 참조

```
#include "opencv2/imgproc/imgproc.hpp"
#include "opencv2/highgui/highgui.hpp"
#include <stdlib.h>
#include <stdio.h>

using namespace cv;

int main( int argc, char** argv )
{
    Mat src, gray, dst, abs_dst;
    src = imread( "lena.jpg" );

    /// Remove noise by blurring with a Gaussian filter
    GaussianBlur( src, src, Size(3,3), 0, 0, BORDER_DEFAULT );
    cvtColor( src, gray, CV_RGB2GRAY );

    /// Apply Laplace function
    Laplacian( gray, dst, CV_16S, 3, 1, 0, BORDER_DEFAULT );
    convertScaleAbs( dst, abs_dst );
    imshow( "result", abs_dst );

    waitKey(0);
    return 0;
}
```

6) 결과를 파일로 저장한다.

멀티미디어 시스템 Homework #3

- 입력영상은 3개
 - 1) Lena 영상
 - 2) 본인의 얼굴 영상
 - 3) 차선이 보이는 도로 영상
- Gaussian smoothing 적용 시 sigma (표준편차)를 1~5까지 변경하면서 Laplacian 결과를 만들기
`cv::GaussianBlur (InputArray src, OutputArray dst, Size ksize, double sigmaX, double sigmaY=0, int borderType=BORDER_DEFAULT)`
 - * 주의 사항 : GaussianBlur 함수에서 ksize는 filter의 크기, lena 처럼 작은 영상은 `Size(5,5)`, 만약 해상도가 1000 이상이면 `Size(5,5)` 또는 `Size(7,7)`, 해상도가 2000을 넘으면 `Size(7,7)` 또는 `Size(9,9)` 정도로 설정
 - * 너무 해상도 큰 영상 (2000이상) 은 굳이 하지 말 것. 해상도 줄여서 사용.
 - sigmaX 값은 Gaussian함수의 표준편차이며 1.0, 2.0, 3.0, 4.0, 5.0 으로 변경하면서 Laplacian 결과 만들기
 - 입력영상 1개당 5개 결과영상 저장
 - 결과를 image viewer 로 확인 후 보고서로 복사

멀티미디어 시스템 Homework #3

▶ 추가 고찰

- 1) 컬러영상을 그대로 Gaussian 과 Laplacian 적용하면 결과는?
- 2) 도로영상이 결과에서 직선 차선을 찾을 수 있는 방안은?
 - 직선 방정식을 이용.
 - 인터넷에 많은 내용이 있으니 찾아보자.