

# Praktikum 4

## Image Filtering

### 1. Grayscale Smoothing

Program berikut ini merubah image color menjadi grayscale kemudian diberikan fungsi smoothing untuk memperhalus gambar.

```
#include <cv.h> //main OpenCV header
#include <highgui.h> //GUI header

int main()
{
    // Declare a new IplImage pointer
    IplImage* gray;
    IplImage* dst;

    // Load an image
    gray = cvLoadImage("image.jpg",0);

    // Create a new window & display the image
    cvNamedWindow("Grayscale", 1);
    cvMoveWindow("Grayscale", 300, 100);
    cvNamedWindow("Destination", 1);
    cvMoveWindow("Destination", 500, 100);

    // Fuction of processing image
    dst = cvCreateImage( cvSize(gray->width, gray->height), IPL_DEPTH_8U, 1 );

    cvSmooth(
        gray, dst,
        CV_GAUSSIAN,
        5, 1, 0, 0
    );

    cvShowImage("Grayscale", gray);
    cvShowImage("Destination", dst);

    // Wait for key to close the window
    cvWaitKey(0);
    cvDestroyWindow( "Grayscale" );
    cvReleaseImage( &gray );
    cvDestroyWindow( "Destination" );
    cvReleaseImage( &dst );
    return 0;
}
```

Petunjuk praktikum:

- Ubah tipe smooting pada cvSmooth() dengan tipe smooting yang lainnya (CV\_BLUR\_NO\_SCALE, CV\_BLUR, CV\_GAUSSIAN, CV\_MEDIAN, CV\_BILATERAL) jangan lupa untuk mengubah parameter di dalamnya.
- Jelaskan konsep smooting dan perbedaan masing-masing tipe smooting diatas.

## 2. Color Smoothing with Trackbar

Program berikut ini menggunakan trackbar untuk memilih fungsi *filtering* pada *image color* untuk memperhalus gambar.

```
#include <cv.h>
#include <highgui.h>

int g_switch_value = 0;
int filterInt = 0;
int lastfilterInt = -1;

void switch_callback( int position ){
    filterInt = position;
}

int main( int argc, char** argv )
{
    const char* name = "Filters Window";
    IplImage* img = cvLoadImage( "image.jpg" );
    IplImage* out = cvCreateImage( cvGetSize(img), IPL_DEPTH_8U, 3 );

    cvNamedWindow( name, 1 );
    cvShowImage(name, out);

    // Create trackbar
    cvCreateTrackbar( "FILTER", name, &g_switch_value, 3, switch_callback );

    while( 1 ) {
        switch( filterInt ){
            case 0:
                cvSmooth( img, out, CV_BLUR, 1, 1 ); // Do nothing
                break;
            case 1:
                cvSmooth( img, out, CV_BLUR, 7, 7 );
                break;
            case 2:
                cvSmooth( img, out, CV_GAUSSIAN, 7, 7 );
                break;
            case 3:
                cvSmooth( img, out, CV_MEDIAN, 7, 7 );
                break;
        }
        if(filterInt != lastfilterInt){
            cvShowImage(name, out);
            lastfilterInt = filterInt;
        }
        if( cvWaitKey( 15 ) == 27 )
            break;
    }

    cvReleaseImage( &img );
    cvReleaseImage( &out );
    cvDestroyWindow( name );

    return 0;
}
```

Petunjuk praktikum:

- Jelaskan perbedaan program color smooting dengan grayscale smooting pada program sebelumnya.
- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.
  - o cvSmooth()

### 3. PyrDown (Gaussian Pyramid Down)

Program berikut ini menggunakan fungsi PyrDown dengan melakukan downsampling pada dekomposisi (pemecahan) Gaussian pyramid. Pertama dilakukan konvolusi pada image dengan filter tertentu. Kemudian dilakukan downsampling pada gambar dengan menghapus baris dan and kolom yang genap.

```
#include <cv.h> //main OpenCV header
#include <highgui.h> //GUI header

int main()
{
    // Declare a new IplImage pointer
    IplImage* src;
    IplImage* dst;

    // Load an image
    src = cvLoadImage("image.jpg",1);

    // Create a new window & display the image
    cvNamedWindow("Source", 1);
    cvMoveWindow("Source", 100, 100);
    cvNamedWindow("Destination", 1);
    cvMoveWindow("Destination", 500, 100);

    // Destination should have 2x smaller width and height than the source
    dst = cvCreateImage( cvSize(src->width/2, src->height/2), IPL_DEPTH_8U, 3);

    // Fuction of processing image
    cvPyrDown( src, dst, CV_GAUSSIAN_5x5 );

    cvShowImage("Source", src);
    cvShowImage("Destination", dst);

    // Wait for key to close the window
    cvWaitKey(0);
    cvDestroyWindow( "Source" );
    cvReleaseImage( &src );
    cvDestroyWindow( "Destination" );
    cvReleaseImage( &dst );
    return 0;
}
```

Petunjuk praktikum:

- Jelaskan konsep Gaussian pyramid down (PyrDown) pada program diatas.
- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.
  - cvPyrDown()

## 4. PyrUp (Gaussian Pyramid Up)

Program berikut ini menggunakan fungsi PyrUp dengan melakukan up-sampling pada pemecahan Gaussian pyramid. Pertama dilakukan up-sampling pada gambar dengan memasukkan nilai kosong pada baris dan and kolom yang genap. Kemudian dilakukan konvolusi pada image dengan filter “multiplied by 4” untuk interpolasi (penambahan). Jadi gambar lebih besar 4 kali dari gambar asal.

```
#include <cv.h> //main OpenCV header
#include <highgui.h> //GUI header

int main()
{
    // Declare a new IplImage pointer
    IplImage* src;
    IplImage* dst;

    // Load an image
    src = cvLoadImage("image.jpg",1);

    // Create a new window & display the image
    cvNamedWindow("Source", 1);
    cvMoveWindow("Source", 100, 100);
    cvNamedWindow("Destination", 1);
    cvMoveWindow("Destination", 500, 100);

    // Destination should have 2x smaller width and height than the source
    dst = cvCreateImage( cvSize(src->width*2, src->height*2), IPL_DEPTH_8U, 3 );

    // Fuction of processing image
    cvPyrUp( src, dst, CV_GAUSSIAN_5x5 );

    cvShowImage("Source", src);
    cvShowImage("Destination", dst);

    // Wait for key to close the window
    cvWaitKey(0);
    cvDestroyWindow( "Source" );
    cvReleaseImage( &src );
    cvDestroyWindow( "Destination" );
    cvReleaseImage( &dst );
    return 0;
}
```

Petunjuk praktikum:

- Jelaskan konsep Gaussian pyramid up (PyrUp) pada program diatas.
- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.

- cvPyrDown()

## Tugas: Filter2D with Kernel

Cobalah fungsi berikut ini:

- `cvCopyMakeBorder()`
- `cvIntegral()`

Buatlah program menggunakan fungsi `cvFilter2D()` untuk mengkonvolusi image dengan kernel / matriks transformasi image yang dibuat sendiri.