

Praktikum 3

Color Conversion

1. Grayscale

Program berikut ini mengubah dari color image menjadi grayscale image.

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

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

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

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

    // Fuction of processing image

    gray = cvCreateImage( cvSize(src->width, src->height), IPL_DEPTH_8U, 1 );
    cvCvtColor( src, gray, CV_BGR2GRAY );

    cvShowImage("Source", src);
    cvShowImage("Grayscale", gray);

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

Petunjuk praktikum:

- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.
 - cvCreateImage()
 - cvCvtColor()
- Jika yang diinginkan hanya menampilkan gambar grayscale saja, coba persingkat program tanpa menggunakan fungsi cvCreateImage() dan cvCvtColor().

2. Grayscale Threshold

Program berikut ini mengubah dari color image menjadi grayscale kemudian ke grayscale threshold (binary image).

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

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

    int threshold = 128, maxValue = 255;
    int thresholdType = CV_THRESH_BINARY;

    // Load an image
    src = cvLoadImage("image.jpg",1);
    gray = cvCreateImage( cvSize(src->width, src->height), IPL_DEPTH_8U, 1 );
    cvCvtColor( src, gray, CV_BGR2GRAY );
    grayThresh = cvCloneImage( gray );

    // Create a new window & display the image
    cvNamedWindow( "src", 1 ); cvShowImage( "src", src );
    cvNamedWindow( "gray", 1 ); cvShowImage( "gray", gray );

    // Fuction of processing image
    cvThreshold(gray, grayThresh, threshold, maxValue, thresholdType);
    cvNamedWindow( "grayThresh", 1 ); cvShowImage( "grayThresh", grayThresh );

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

Petunjuk praktikum:

- Ubah nilai variable threshold pada program di atas dalam range antara 0-255 kemudian amati perubahan pada gambar.
- Jelaskan konsep dasar grayscale threshold.
- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.
 - cvCloneImage()
 - cvThreshold()

3. Color Threshold

Program berikut ini mengubah dari color image menjadi color threshold.

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

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

    int threshold = 150, maxValue = 255;
    int thresholdType = CV_THRESH_BINARY;

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

    // Create a new window & display the image
    cvNamedWindow( "src", 1 ); cvShowImage( "src", src );

    // Fuction of processing image
    cvThreshold(src, colorThresh, threshold, maxValue, thresholdType);
    cvNamedWindow( "colorThresh", 1 ); cvShowImage( "colorThresh", colorThresh );

    // Wait for key to close the window
    cvWaitKey(0);
    cvDestroyWindow( "src" );
    cvDestroyWindow( "colorThresh" );
    cvReleaseImage( &src );
    cvReleaseImage( &colorThresh );

    return 0;
}
```

Petunjuk praktikum:

- Ubah nilai variable threshold pada program di atas dalam range antara 0-255 kemudian amati perubahan pada gambar.
- Jelaskan konsep dasar color threshold dan sebutkan perbedaannya dengan grayscale threshold.

4. Threshold by using sum_rgb()

Program berikut ini mengubah dari color image menjadi grayscale threshold (binary image).

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

void sum_rgb( IplImage* src, IplImage* dst )
{
    // Allocate image planes
    IplImage* r = cvCreateImage( cvGetSize(src), IPL_DEPTH_8U, 1 );
    IplImage* g = cvCreateImage( cvGetSize(src), IPL_DEPTH_8U, 1 );
    IplImage* b = cvCreateImage( cvGetSize(src), IPL_DEPTH_8U, 1 );

    // Split image onto the color planes
    cvSplit( src, r, g, b, NULL );

    IplImage* s = cvCreateImage( cvGetSize(src), IPL_DEPTH_8U, 1 );

    // Add equally weighted rgb values
    cvAddWeighted( r, 1./3., g, 1./3., 0.0, s );
    cvAddWeighted( s, 2./3., b, 1./3., 0.0, s );

    // Truncate values over 100
    cvThreshold( s, dst, 100, 100, CV_THRESH_BINARY );

    cvReleaseImage( &r );
    cvReleaseImage( &g );
    cvReleaseImage( &b );
    cvReleaseImage( &s );
}

int main( int argc, char** argv )
{
    const char* name = "Thresholding";
    cvNamedWindow( name, 1 );

    IplImage* src = cvLoadImage("image.jpg");
    IplImage* dst = cvCreateImage( cvGetSize(src), src->depth, 1 );
    sum_rgb( src, dst);

    cvShowImage(name, dst);

    while( 1 ){
        if( (cvWaitKey(10)&0x7f) == 27 )
            break;
    }

    cvDestroyWindow( name );
    cvReleaseImage( &src );
    cvReleaseImage( &dst );

    return 0;
}
```

Petunjuk praktikum:

- Jelaskan algoritma threshold pada contoh diatas dan sebutkan perbedaannya dengan program grayscale threshold sebelumnya.

5. Program: Adaptive Threshold

Program berikut ini mengubah dari color image menjadi grayscale kemudian merubahnya menjadi threshold image dengan menggunakan metode adaptive threshold (tanpa memasukkan nilai threshold).

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

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

    int threshold = 128, maxValue = 255;
    int thresholdType = CV_THRESH_BINARY;

    // Load an image
    src = cvLoadImage("image.jpg",1);
    gray = cvCreateImage( cvSize(src->width, src->height), IPL_DEPTH_8U, 1 );
    cvCvtColor( src, gray, CV_BGR2GRAY );
    adapThresh = cvCloneImage( gray );

    // Create a new window & display the image
    cvNamedWindow( "src", 1 ); cvShowImage( "src", src );
    cvNamedWindow( "gray", 1 ); cvShowImage( "gray", gray );

    // Fuction of processing image
    cvAdaptiveThreshold(
        gray,
        adapThresh,
        255,
        CV_ADAPTIVE_THRESH_MEAN_C,
        CV_THRESH_BINARY,
        3,
        5 );

    cvNamedWindow( "adapThresh", 1 ); cvShowImage( "adapThresh", adapThresh );

    // Wait for key to close the window
    cvWaitKey(0);
    cvDestroyWindow( "src" );
    cvDestroyWindow( "gray" );
    cvDestroyWindow( "grayThresh" );
    cvReleaseImage( &src );
    cvReleaseImage( &gray );
    cvReleaseImage( &adapThresh );
    return 0;
}
```

Petunjuk praktikum:

- Ubah nilai parameter pada cvAdaptiveThreshold pada program di atas dalam kemudian amati perubahan pada gambar.
- Jelaskan konsep dasar adaptive threshold dan sebutkan perbedaannya dengan threshold sebelumnya.
- Jelaskan fungsi berikut ini beserta dengan parameter yang ada di dalamnya.

○ cvAdaptiveThreshold()

Tugas: Converts image from one color space to another

Buatlah program dengan fungsi `cvCvtColor()` untuk menkonversi gambar dari

- RGB ke YCrCb
- RGB ke HSV
- RGB ke HLS
- RGB ke CIE L*a*b*
- RGB ke CIE L*u*v*