

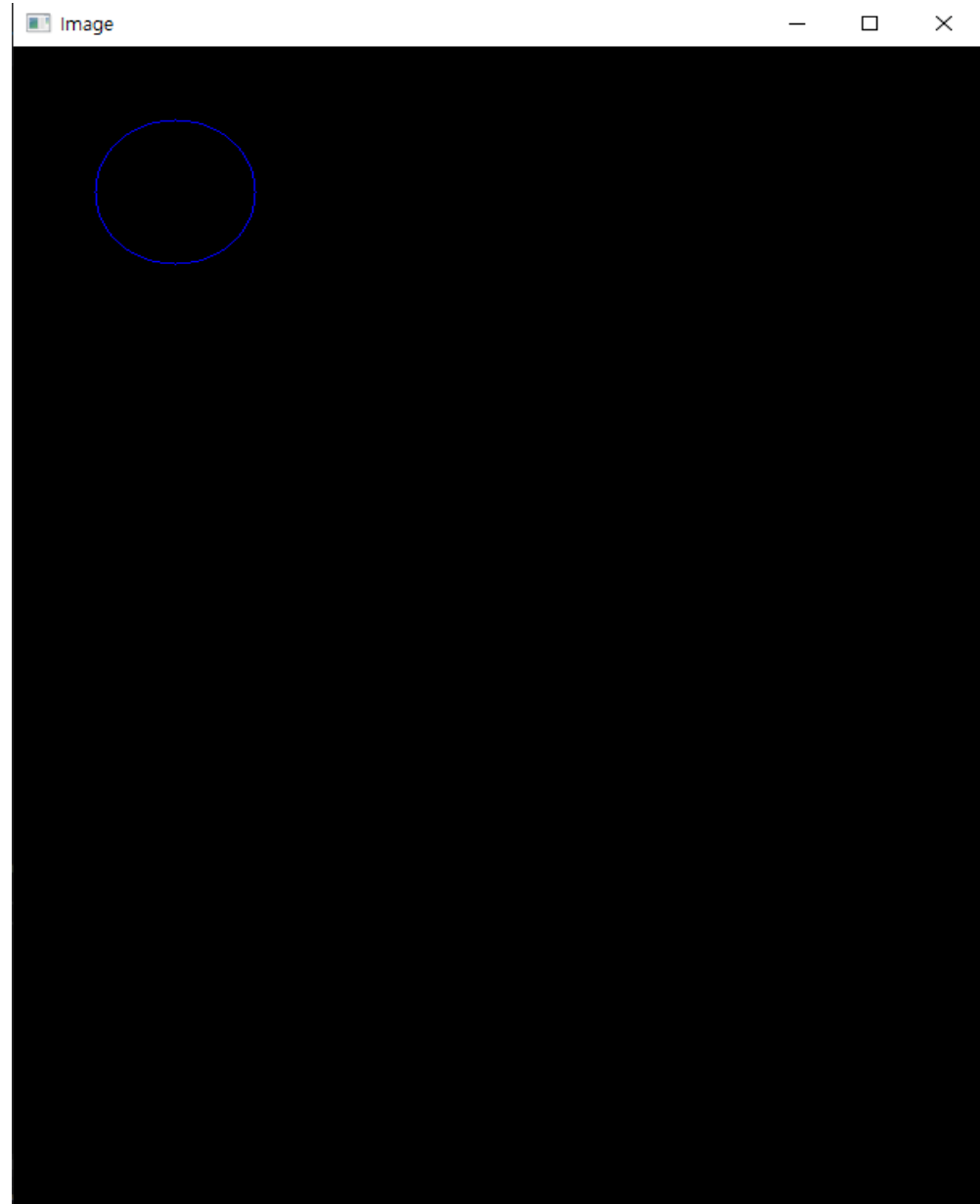
영상처리 실제 - 2주차 과제

p. 03-OpenCV의 기초(1) - p.40 - 1번

```
//HW1
#include
using namespace cv;
int main()
{
    Scalar blue(255, 0, 0); //색상선언
    Mat image = Mat(800, 600, CV_8UC3, Scalar(0, 0, 0));

    circle(image, Point(100, 100), 50, blue);
    imshow("Image", image);
    waitKey(0);

    return 0;
}
```



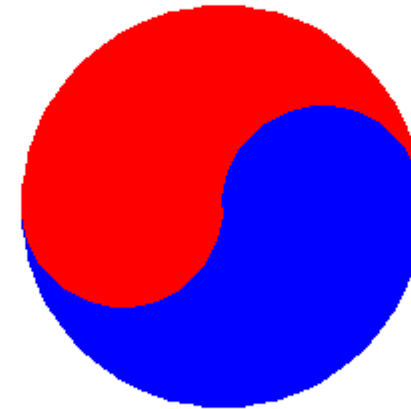
p. 03-OpenCV의 기초(1) - p.40 - 2번

```
//HW2
#include 1
Mat image = Mat(400, 600, CV_8UC3, Scalar(255, 255, 255));
int nDiameter = 200;
Scalar blue(255, 0, 0), red(0, 0, 255); // 색상
Point center(image.cols / 2, image.rows / 2); // 태극 중심 좌표
Size circle_Big(nDiameter / 2, nDiameter / 2), circle_small(nDiameter / 4, nDiameter / 4); // 태극 문양을 그리기 위한 두개의 원

// 빨강과 파란색의 반원을 아래위로 그린다.
ellipse(image, center, circle_Big, 0, 0, 180, blue, -1);
ellipse(image, center, circle_Big, 0, 180, 360, red, -1);
// 작은 반원을 아래 위로 그려준다.
ellipse(image, Point(center.x - circle_small.width, center.y), circle_small, 0, 0, 180, red, -1);
ellipse(image, Point(center.x + circle_small.width, center.y), circle_small, 0, 180, 360, blue, -1);

imshow("Image", image);
waitKey(0);
#endif
```

Image



p. 03-OpenCV의 기초(2) - p.35 - 1,2번

```
//3장 HW1
#include <opencv2/opencv.hpp>
using namespace cv;

int main()
{
    if 1
    {
        img_hw1 = Mat(800, 600, CV_8UC1, Scalar(120));
        namedWindow("hw1", WINDOW_AUTOSIZE);
        imshow("hw1", img_hw1);
        setMouseCallback("hw1", hwDraw, &img_hw1);
        createTrackbar("Line", "hw1", &nLineThickness, 10, on_trackbar_hw1);
        createTrackbar("Radius", "hw1", &nRadius, 200, on_trackbar_hw1);

        waitKey(0);
    }
    #endif
}
```

```
void hwDraw(int event, int x, int y, int flags, void* param)
{
    if (event == EVENT_LBUTTONDOWN)
    {
        Mat img = *(Mat*)(param);
        if (nLineThickness < 1)
        {
            if (nRadius < 10)
            {
                circle(img, Point(x, y), 10, Scalar(0, 255, 0), 1);
            }
            else
            {
                circle(img, Point(x, y), nRadius, Scalar(0, 255, 0), 1);
            }
        }
        else
        {
            if (nRadius < 10)
            {
                circle(img, Point(x, y), 10, Scalar(0, 255, 0), nLineThickness);
            }
            else
            {
                circle(img, Point(x, y), nRadius, Scalar(0, 255, 0), nLineThickness);
            }
        }
        imshow("hw1", img); // 영상이 변경되면 다시 표시한다
    }
    else if (event == EVENT_RBUTTONDOWN)
    {
        Mat img = *(Mat*)(param);
        Point<int> pt1(x, y);
        Rect rect(pt1, Size(100, 100));
        if (nLineThickness <= 0)
        {
            rectangle(img, rect, blue, 2); //사각형 그리기
        }
        else
        {
            rectangle(img, rect, blue, nLineThickness); //사각형 그리기
        }
        imshow("hw1", img); // 영상이 변경되면 다시 표시한다
    }
    else if (event == EVENT_LBUTTONUP)
    {
    }
    else if (event == EVENT_RBUTTONUP)
    {
    }
}
```

