영상처리 실제 7주차 실습_형태학적 처리

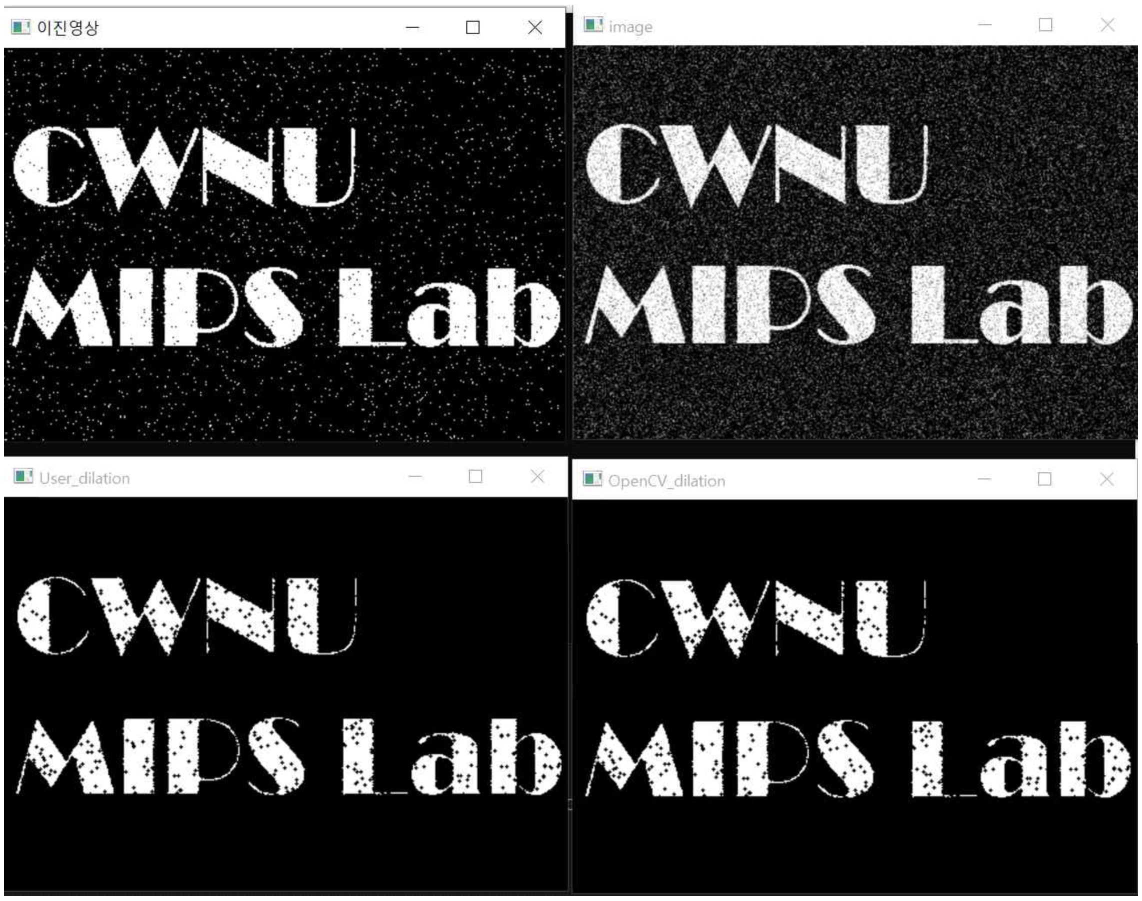
2023254015 장욱진

```
#include <opencv2/opencv.hpp>

using namespace cv;
using namespace std;

bool check_match(Mat img, Point start, Mat mask, int mode = 0)
{
	for (int u = 0; u < mask.rows; u++)
	{
		for (int v = 0; v < mask.cols; v++)
		{
			Point pt(v, u);
			int m = mask.at<uchar>(pt);
			int p = img.at<uchar>(start + pt);

			bool ch = (p == 255);
			if (m == 1 && ch == mode)
				return false;
		}
	}
	return true;
}

void erosion(Mat img, Mat dst, Mat mask)
{
	dst - Mat(img.size(), CV_8U, Scalar(0));
	if (mask.empty()) mask = Mat(3, 3, CV_8UC1, Scalar(1));

	Point h_m = mask.size() / 2;
	for (int i = h_m.y; i < img.rows - h_m.y; i++)
	{
		for (int j = h_m.x; j < img.cols - h_m.x; j++)
		{
			Point start = Point(j, i) - h_m;
			bool check = check_match(img, start, mask, 0);
			dst.at<uchar>(i, j) = (check) ? 255 : 0;
		}
	}
}

void dilation(Mat img, Mat& dst, Mat mask) {
	dst = Mat(img.size(), CV_8U, Scalar(0));
	if (mask.empty()) mask = Mat(3, 3, CV_8UC1, Scalar(0));

	Point h_m = mask.size() / 2;

	for (int i = h_m.y; i < img.rows - h_m.y; i++)
	{
		for (int j = h_m.x; j < img.cols - h_m.x; j++)
		{
			Point start = Point(j, i) - h_m;
			bool check = check_match(img, start, mask, 1);
			dst.at<uchar>(i, j) = (check) ? 0 : 255;

		}

	}
}

void opening(Mat img, Mat& dst, Mat mask)
{
	Mat tmp;
	erosion(img, tmp, mask);
	dilation(tmp, dst, mask);
}
```

```cpp
void closing(Mat img, Mat& dst, Mat mask)
{
	Mat tmp;
	dilation(img, tmp, mask);
	erosion(tmp, dst, mask);
}

void page6()
{
	Mat image = imread("../image/morph_test1.jpg", 0);
	CV_Assert(image.data);
	Mat th_img, dst1, dst2;
	threshold(image, th_img, 128, 255, THRESH_BINARY);

	uchar data[] = { 0,1,0,
					1,1,1,
					0,1,0 };

	Mat mask(3, 3, CV_8UC1, data);

	erosion(th_img, dst1, (Mat)mask);
	morphologyEx(th_img, dst2, MORPH_ERODE, mask);

	imshow("image", image), imshow("이진영상", th_img);
	imshow("User_dilation", dst1), imshow("OpenCV_dilation", dst2);

	waitKey();
}

void page10()
{
	Mat image = imread("../image/morph_test1.jpg", 0);
	CV_Assert(image.data);
	Mat th_img, dst1, dst2;
	threshold(image, th_img, 128, 255, THRESH_BINARY);

	Matx <uchar, 3, 3> mask;
	mask << 0, 1, 0,
		1, 1, 1,
		0, 1, 0;

	mask << 0, 1, 0, 1, 1, 1, 0, 1, 1;

	dilation(th_img, dst1, (Mat)mask);

	morphologyEx(th_img, dst2, MORPH_DILATE, mask);

	imshow("image", image), imshow("User_dilation", dst1);
	waitKey();
}

void page18()
{
	Mat image = imread("../image/morph_test1.jpg", 0);
	CV_Assert(image.data);
	Mat th_img, dst1, dst2, dst3, dst4;
	threshold(image, th_img, 128, 255, THRESH_BINARY);

	Matx <uchar, 3, 3> mask;
	mask << 0, 1, 0,
		1, 1, 1,
		0, 1, 0;

	opening(th_img, dst1, (Mat)mask);
	closing(th_img, dst2, (Mat)mask);
	morphologyEx(th_img, dst3, MORPH_OPEN, mask);
	morphologyEx(th_img, dst4, MORPH_CLOSE, mask);

	imshow("User_opening", dst1), imshow("User_closing", dst2);
	imshow("Opencv_opening", dst3), imshow("Opencv_closing", dst4);
```

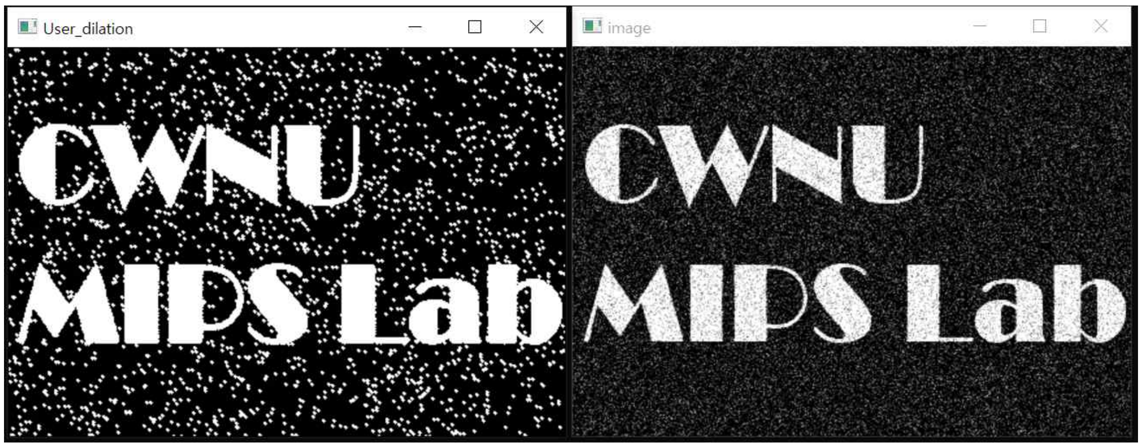
```
		waitKey();
}

void page23()
{
		while (1)
		{
				int no;
				cout << "차량 영상 번호(0:종료) : ";
				cin >> no;
				if (no == 0)break;

				string fname = format("./test_car/%02d.jpg", no);
				Mat image = imread(fname, 1);

				if (image.empty())
				{
						cout << to_string(no) + "번 영상 파일이 없습니다." << endl;
						continue;
				}

				Mat gray, sobel, th_img, morph;
				Mat kernel(5, 25, CV_8UC1, Scalar(1));
				cvtColor(image, gray, COLOR_BGR2GRAY);

				blur(gray, gray, Size(5, 5));
				Sobel(gray, gray, CV_8U, 1, 0, 3);

				threshold(gray, th_img, 120, 255, THRESH_BINARY);
				morphologyEx(th_img, morph, MORPH_CLOSE, kernel);

				imshow("image", image);
				imshow("이진영상", th_img), imshow("열림연산", morph);
				waitKey(0);
		}
}

void page27()
{
		Mat img = imread("./letterb.png", CV_LOAD_IMAGE_GRAYSCALE);
		threshold(img, img, 127, 255, cv::THRESH_BINARY);
		imshow("src", img);
		Mat skel(img.size(), CV_8UC1, Scalar(0));
		Mat element = getStructuringElement(MORPH_CROSS, Size(3, 3));
		Mat temp, eroded;
		do
		{
				erode(img, eroded, element);
				dilate(eroded, temp, element);
				subtract(img, temp, temp);
				bitwise_or(skel, temp, skel);
				eroded.copyTo(img);
		} while ((countNonZero(img) != 0));
		imshow("result", skel);
		waitKey(0);
}

int main()
{
		page6();
		page10();
		page18();
		page23();
		page27();
}
```

결과화면



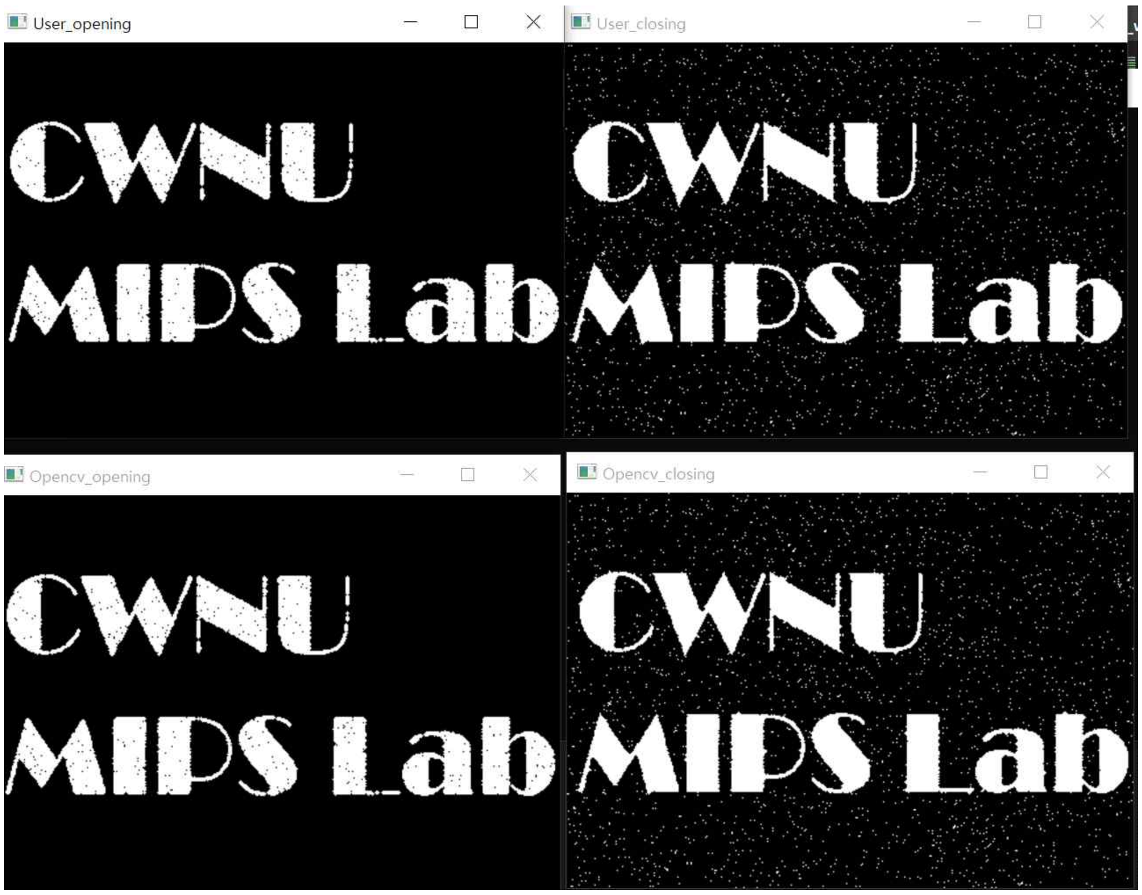
이진영상
image
User_dilation
OpenCV_dilation

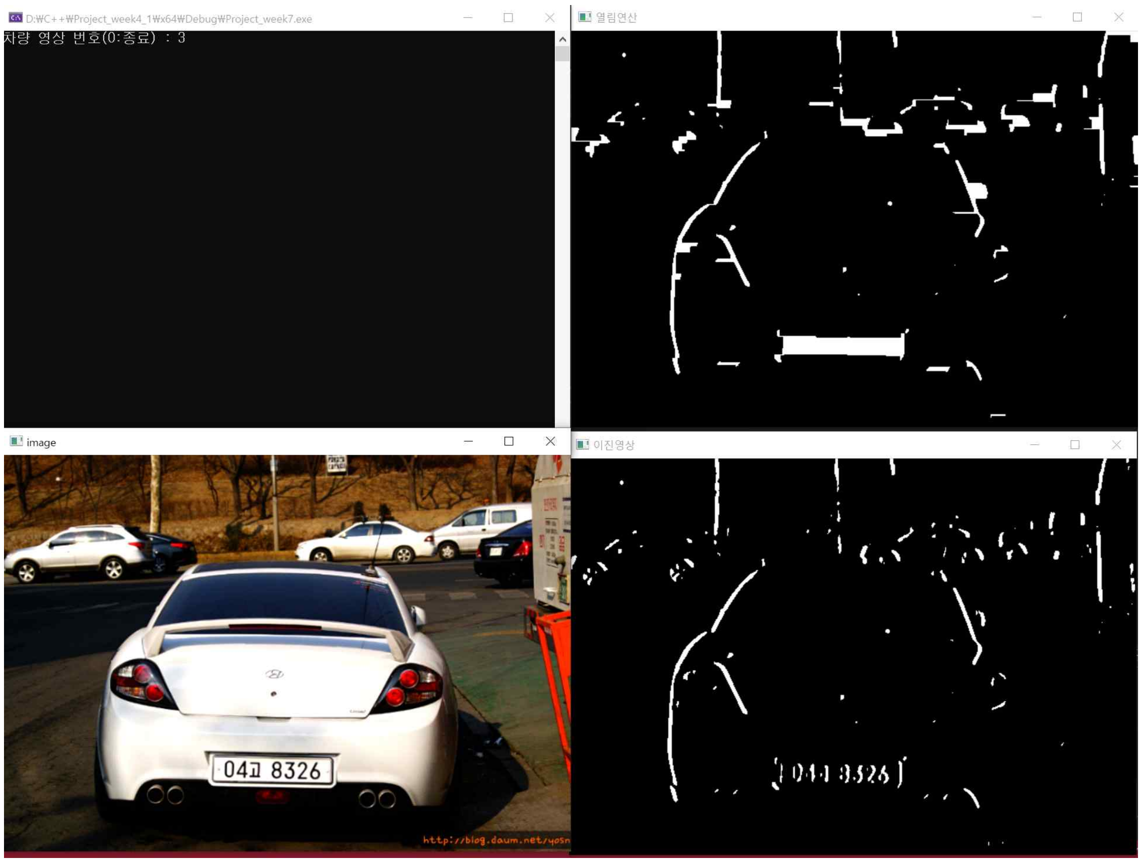
<page6 결과화면>

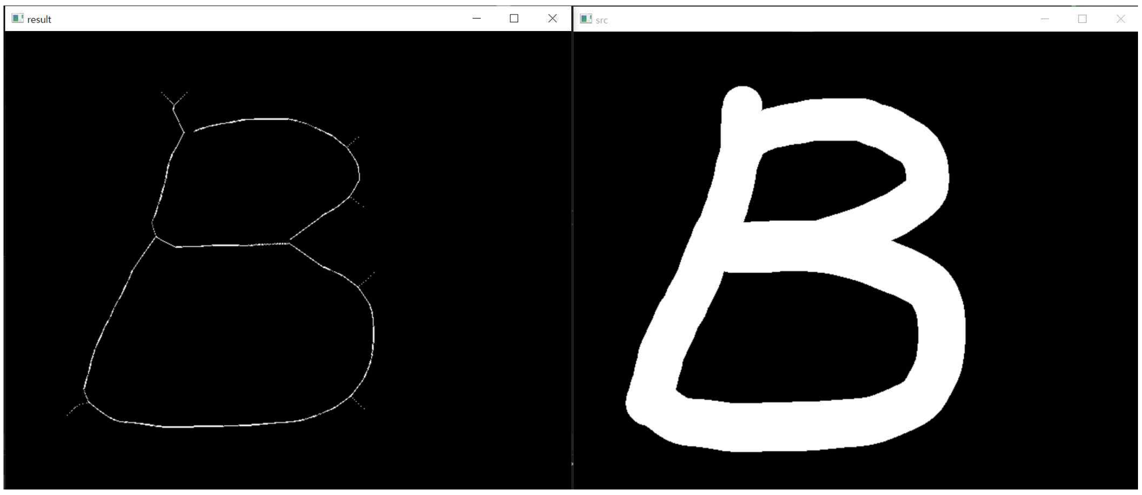


User_dilation
image

<page10 결과화면>


User_opening
User_closing
Opencv_opening
Opencv_closing
CWNU
MIPS Lab
<page18 결과화면>


D:₩C++₩Project_week4_1₩x64₩Debug₩Project_week7.exe
열림연산
image
이진영상
```
차량 영상 번호(0:종료) : 3
```
<page23 결과화면>


result
src

<page27 결과화면>