This doc contains

1. Histogram Equalization
2. Different types of Thresholding
3. Morphological Erode, Dilate

IplImage is the object for image. You can load an image from file or you can capture from camera. You can do operation on this IplImage object.

Import

```
import  com.googlecode.javacv.cpp.opencv_objdetect;
import com.googlecode.javacpp.Loader;

import static com.googlecode.javacv.cpp.opencv_core.*;
import static com.googlecode.javacv.cpp.opencv_objdetect.*;
import static com.googlecode.javacv.cpp.opencv_imgproc.*;
import static com.googlecode.javacv.cpp.opencv_highgui.*;
```

Image Sharp / Histogram Equalization

Method:

cvEqualizeHist(srcImg, dstImg);

Here,

srcImg = original image, must be grayscale

dstImg = result image will be found here, pass a blank gray image.

Example,

```
IplImage img = cvLoadImage("s.png");

//make a blank gray image
IplImage gray = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);

//convert to gray iamge
cvCvtColor(img, gray, CV_BGR2GRAY);

//make a blank gray image to hold the sharpen image
IplImage sharp = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);

//now do the equalization
cvEqualizeHist(gray, sharp);

//now sharp holds the sharpen image.
//lets show it.
CanvasFrame canvas = new CanvasFrame("MyImage");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);

canvas.showImage(sharp);
```

Output:

Fig: source image

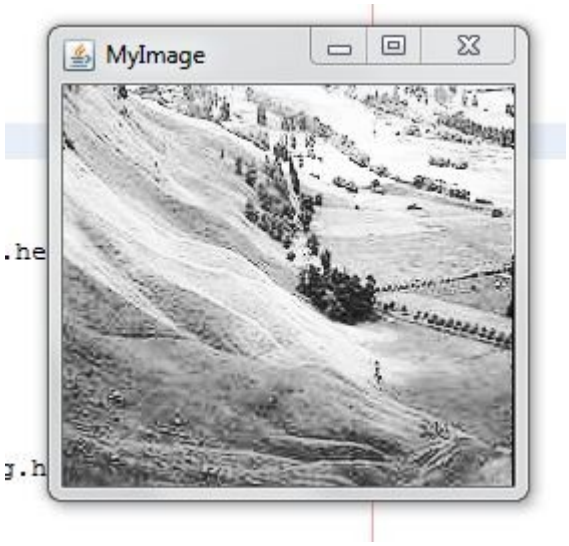


Fig: Histogram equalized image

Thresholding

Method_1

cvThreshold(src, dest, lowerValue, upperValue, flag);

Here,

src = the source image, it must be Grayscale

dest = the resultant will be found here, just pass a 1 channel blank image.

Flag =This define type of thresholding method. Some are

CV_THRESH_BINARY

CV_THRESH_BINARY_INV

CV_THRESH_OTSU

Lower value

upper value

example: Binary Threshold

```
IplImage img = cvLoadImage("108.jpg");

//make gray image
IplImage gray = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);

//convert to gray iamge
cvCvtColor(img, gray, CV_BGR2GRAY);

//create a blank 1 channel image to hold the
// threshold image.
IplImage th = IplImage.create(gray.width(),
```

```
gray.height(), IPL_DEPTH_8U, 1);

//now do the threshold.
cvThreshold(gray, th, 127, 255, CV_THRESH_BINARY);

//now th holds the thresholded image.
//lets show it.
CanvasFrame canvas = new CanvasFrame("MyImage");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);
canvas.showImage(th);
```

Output:

Fig: Input Image

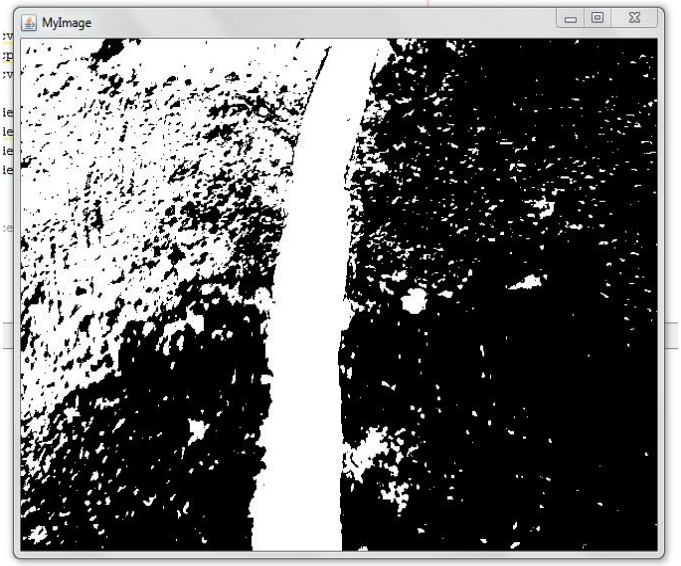
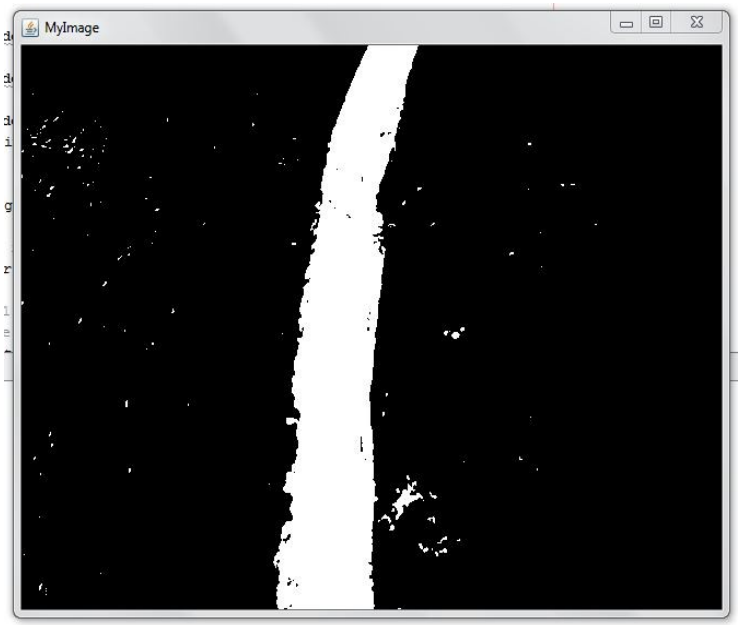


Fig: After Binary Threshold.

Example_2: threshold using OSTU

```
IplImage img = cvLoadImage("108.jpg");

//make gray image
IplImage gray = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);

//convert to gray iamge
cvCvtColor(img, gray, CV_BGR2GRAY);

//create a blank 1 channel image to hold the
// threshold image.
IplImage th = IplImage.create(gray.width(),
gray.height(), IPL_DEPTH_8U, 1);

//now do the threshold.
cvThreshold(gray, th, 127, 255, CV_THRESH_OTSU);

//now th holds the thresholded image.
//lets show it.
CanvasFrame canvas = new CanvasFrame("MyImage");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);
canvas.showImage(th);
```

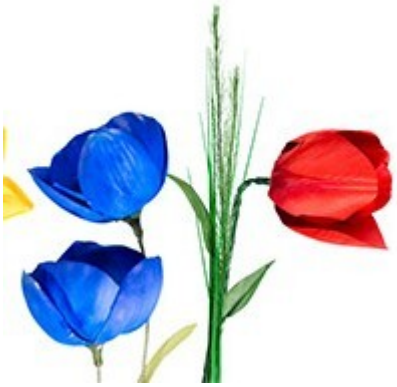
Output:



Method2: cvInRangeS

```
// color range of red like color
int hueLower = 160;
int hueUpper= 180;

IplImage img = cvLoadImage("r.jpg");

//make gray image
IplImage hsv = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 3);

//convert to hsv iamge
cvCvtColor(img, hsv, CV_BGR2HSV);

//create a blank 1 channel image to hold the
// threshold image.
IplImage th = IplImage.create(hsv.width(),
hsv.height(), 8, 1);

// cvScalar : ( H , S , V, A)
cvInRangeS(hsv, cvScalar(hueLower, 100, 100, 0),
cvScalar(hueUpper, 255, 255, 0), th);

cvReleaseImage(hsv);


//now th holds the thresholded image.
//lets show it.
CanvasFrame canvas = new CanvasFrame("MyImage");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);
canvas.showImage(th);
```

Output:



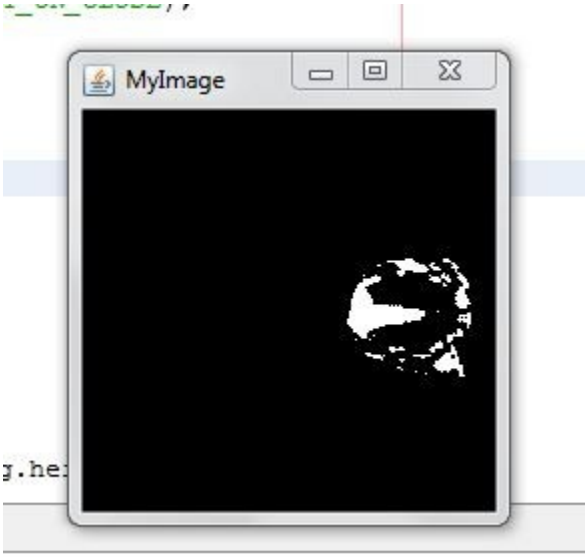


Fig: original image

Fig: only red colored obj detected

For green color, changing **int hueLower = 40;**

int hueUpper = 90;

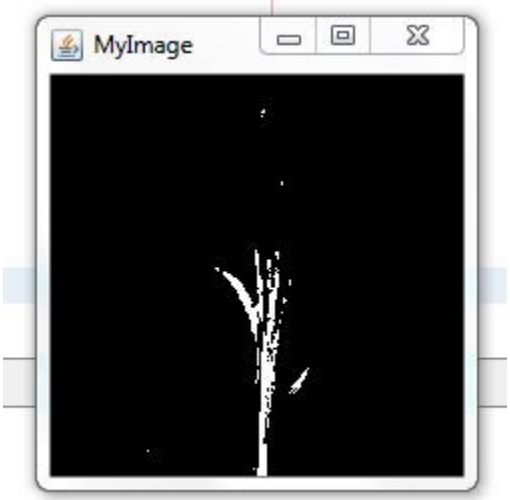


Fig: Green colored object detected

by changing,

int hueLower = 100;

int hueUpper = 120;

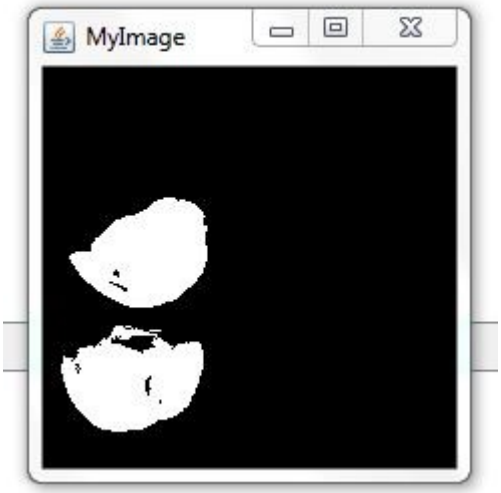


Fig: blue colored object detected.

Morphological **Erode**

Method

cvErode(th, dst, null, erodeCount);

Here,

th = source image

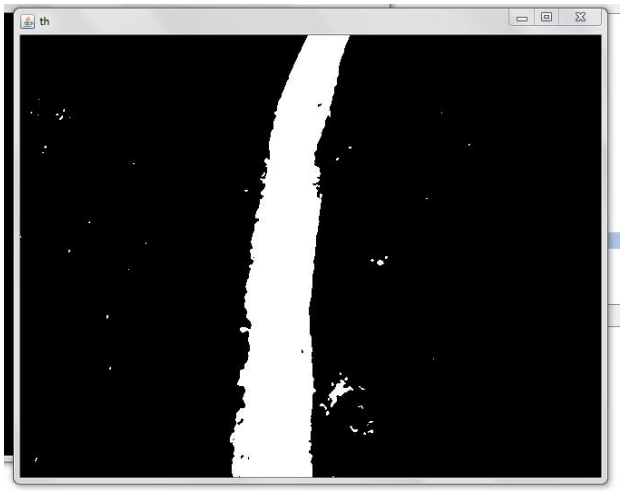
dst = the resultant will be found here, pass a blank image or the source, if source then change will be done on that source image.

ErodeCount = give it any odd value, for example 5

Example

```
CanvasFrame canvas = new CanvasFrame("image");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);


IplImage img = cvLoadImage("108.jpg");
cvSmooth(img, img, CV_GAUSSIAN, 3);


IplImage gray = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);  //make gray image
//convert to gray iamge
cvCvtColor(img, gray, CV_BGR2GRAY);


IplImage th = IplImage.create(gray.width(),
gray.height(), IPL_DEPTH_8U, 1);
cvThreshold(gray, th, 127, 255, CV_THRESH_OTSU);

CanvasFrame canvas2 = new CanvasFrame("th");
canvas2.showImage(th);

IplImage dst = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);  //make gray image


cvErode(th, dst, null, 5);
//cvDilate(dst, dst, null, 7);

canvas.showImage(dst);
```

Output:



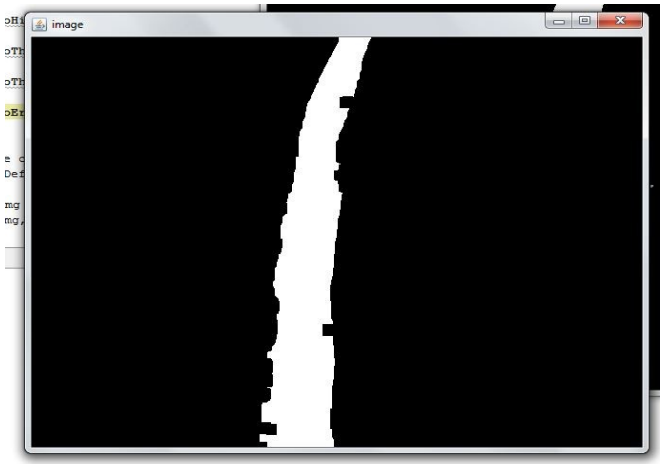


Fig: Thresholded image

Fig: after **erode**

Morphological **Dilate**

Method

cvDilate(dst, dst, null, 7);

same as erode.

Example,

```
CanvasFrame canvas = new CanvasFrame("image");

canvas.setDefaultCloseOperation(CanvasFrame.EXIT_ON_CLO
SE);


IplImage img = cvLoadImage("108.jpg");
cvSmooth(img, img, CV_GAUSSIAN, 3);
```

```
IplImage gray = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);  //make gray image
//convert to gray iamge
cvCvtColor(img, gray, CV_BGR2GRAY);


IplImage th = IplImage.create(gray.width(),
gray.height(), IPL_DEPTH_8U, 1);
cvThreshold(gray, th, 127, 255, CV_THRESH_OTSU);

CanvasFrame canvas2 = new CanvasFrame("th");
canvas2.showImage(th);

IplImage dst = IplImage.create(img.width(),
img.height(), IPL_DEPTH_8U, 1);  //make gray image


cvErode(th, dst, null, 5);
cvDilate(dst, dst, null, 7);

canvas.showImage(dst);
```

Output: