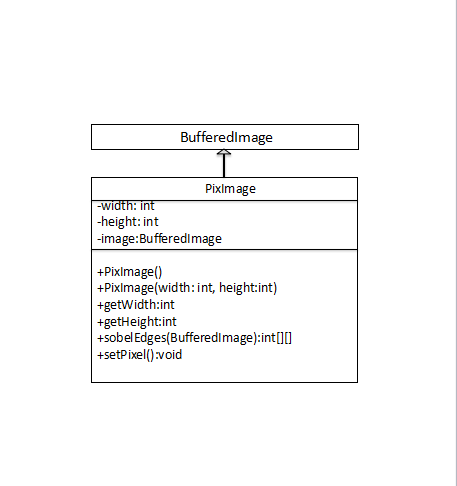
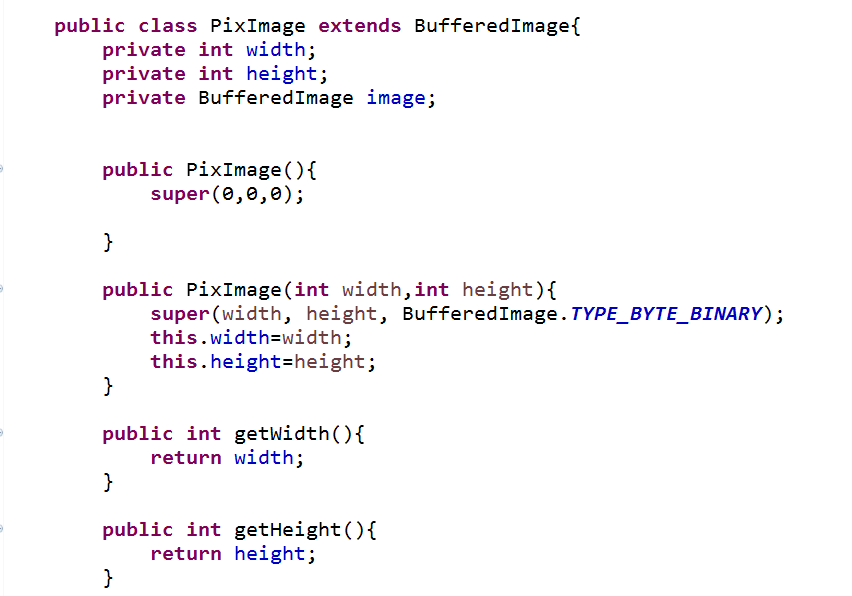
**Project 1: Implement Edge Detection Operation Report**

**Implement PixImage class**



1. Constructor



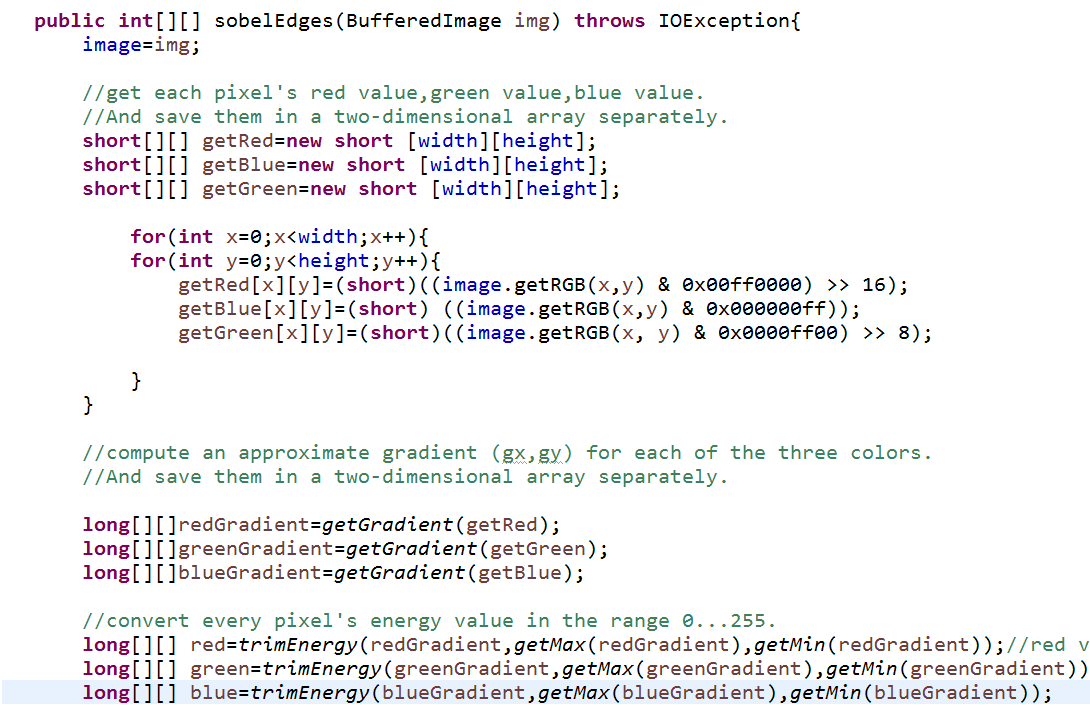
Concrete class PixImage extends BufferedImage.

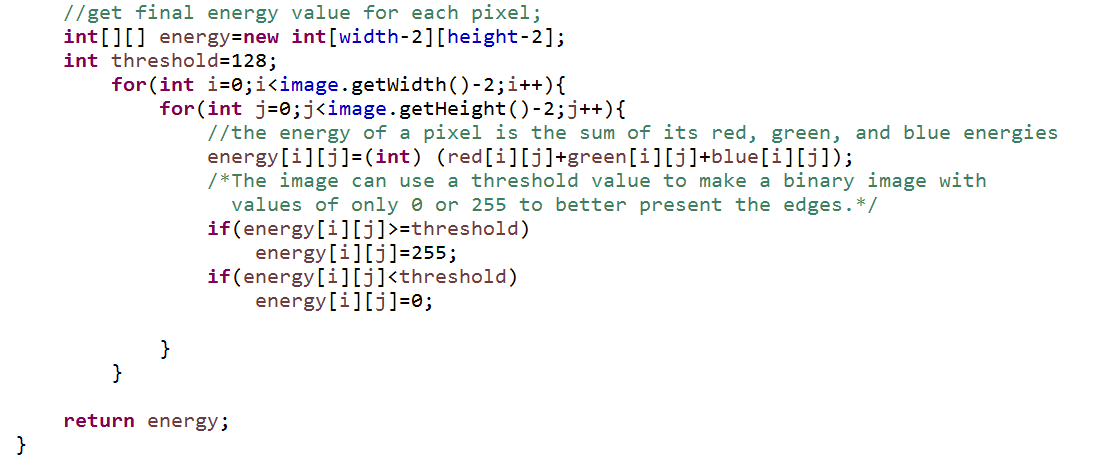
1.Three private data fields were defined:width,height,and image.

2.Two constructors were created, one is default constructor, the other takes two integers “width” and “height”as input,representing the width and height of the image,and returns an image of the specified size.

3.Two methods were created,getWidth() return width value;getHeight() return height value.

1. Detecting edges in an image





1. Get red,green,blue value of each pixel and save three color values in three two dimensional arrays separately.

A pixel is represented by a 4-byte (32 bit) integer, like so:

IMG_256

So use binary arithmetic to get the individual color components,like so:

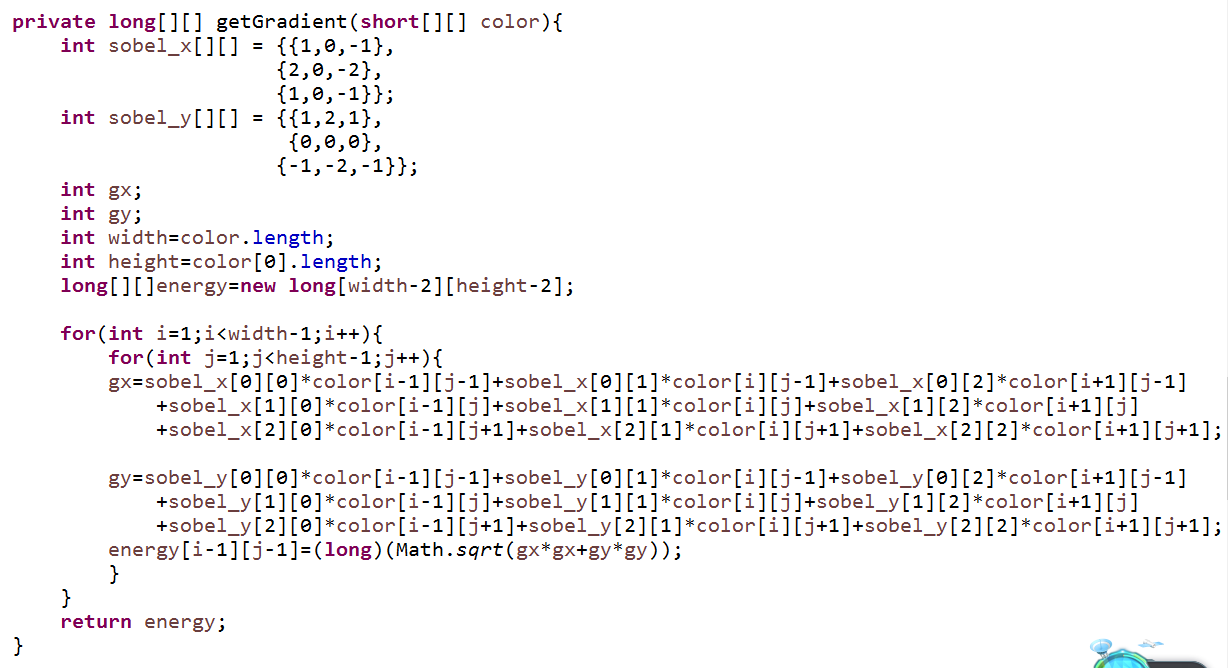
int rgb = getRGB(...);

int red = (rgb >> 16) & 0x000000FF;

int green = (rgb >>8 ) & 0x000000FF;

int blue = (rgb) & 0x000000FF;

1. Use getGradient() method to compute an approximate gradient for each pixel and get the energy.The red,green,and blue intensities are treated separately at first.See below for details
2. Use tirmEnergy() method to convert each gradient vector energy in the range 0...255 and return them in a long type two dimension array.See below for details
3. The energy of a pixel is the sum of its red,green,and blue energies. The final image use a threshold value to make a binary image with values of 0 or 255 to present the edges.Define the threshold value is 128,If the value is greater or equal to 128,it will be 255.If the value is smaller than 128,it will be 0.
4. Return new energy of pixels.



A private method getGradient() was created. it takes certain color values in a short type two dimensional array as parameter and return a long type two dimensional array to show the energy of gradient vectors.

1. Use following convolutions(see Figure 1) to compute gradient(gx, gy).

2.Define the energy of a gradient vector(gx, gy)to be IMG_256

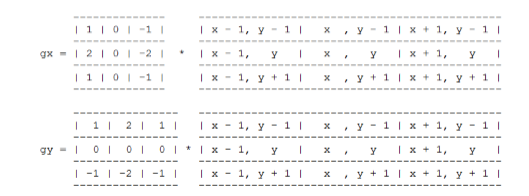
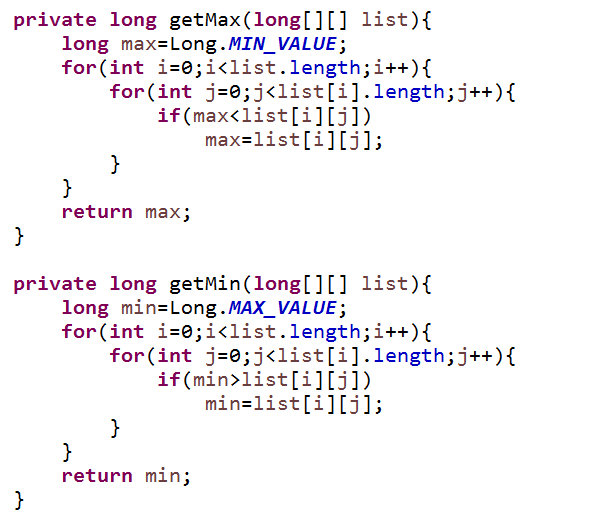
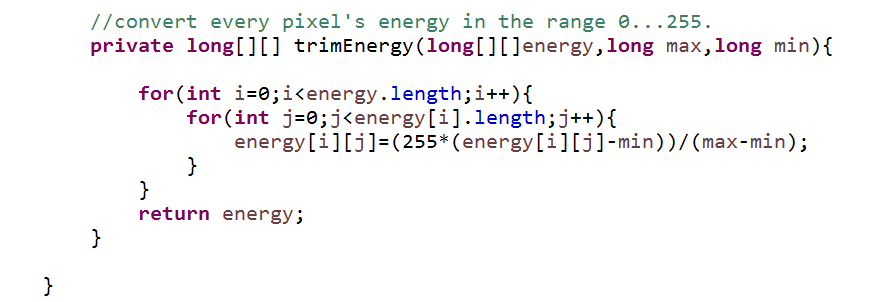


Figure 1:computing gradient of an image





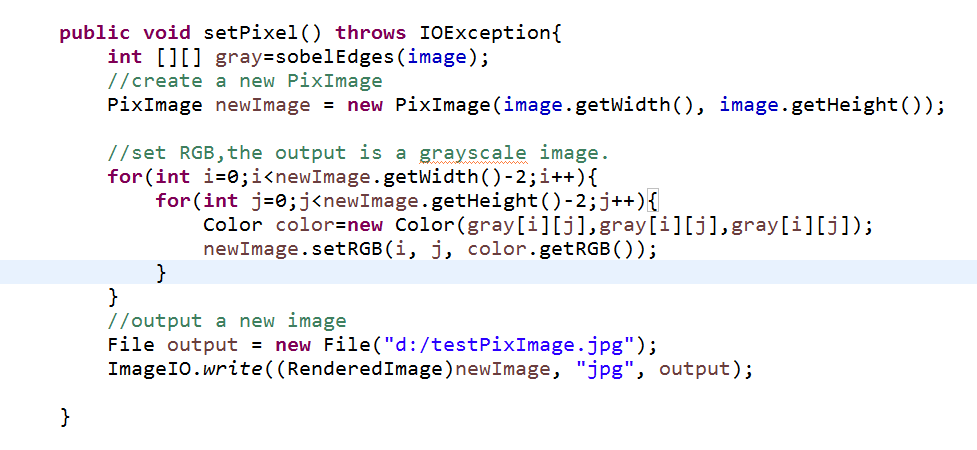
Two private methods getMax(),getMin() were created. The method getMax takes a two dimensional array as parameter return a long type max value .The method getMin return a min value.

Use these methods we can get the max gradient energy and the min gradient energy for red,green,and blue energy values.

The private method trimEnergy() takes long[][]energy,the max gradient energy,and the min gradient energy as parameters.It converts each gradient vector energy in the range 0...255 and return them in a long type two dimension array.

Gradient energy minus the min energy value can change gradient energy value equal or greater than 0;and then multiple 255 then divide (max-min) can change the max value to 255.

3.Changing the image’s pixels

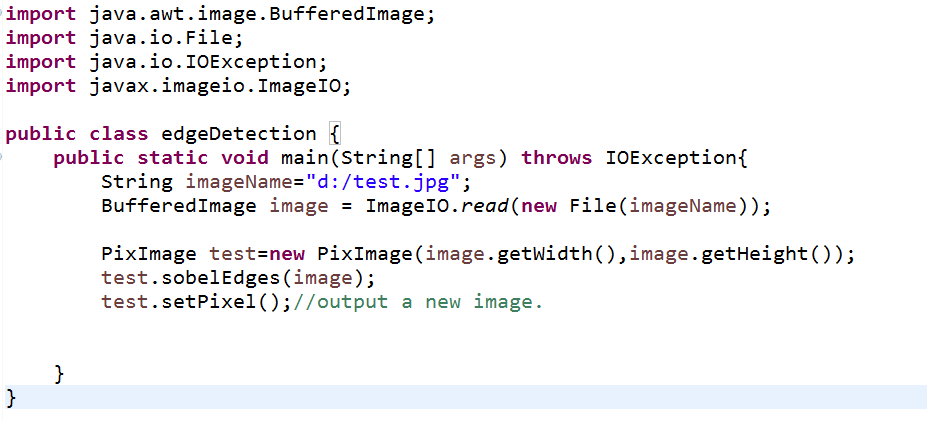


The method setPixel() construct a new PixImage depend on the original PixImage.

Set RGB to each pixel and the image is a grayscale image(red==green==blue for each pixel).

It update each pixel value and output a new Pixmage to show the result of edge detection.

**Test edge detection operation**



Input



Output

