```
/***********************************************************************/
/*    Author name: Rui Chang
/*    Email: rchang@g.clemson.edu
/*    604 project04
/***********************************************************************/
```

## 1.Description

1.Read image and store it in pixels,and store its channels.
2.If argc is 3 or 4.It is to use filter in .filt file. Read filtdata and store it as filter.
If argc is 6 or 7 and argc[1] is "–g" , generate gaborfilter.
3.Display the image.
4. If type key "c", convolve image using filter and change display image to this convolved image. Type "r" to display original image.
5.If type key "w", write the displaying image to the last parameter you type.
6.If type key "q", quit.

## 2.How to run code

1.make it .
2.If you want to use filter in .filt file , type : (output image is not necessary)
./filt [filter_name.filt] [input image] [output image]
eg. ./filt box.filt square.png s.png
3.If you want to use gaber filter, type: (output image is not necessary)
./filt –g [theta] [sigma] [period] [input image] [output image]
eg. ./filt –g 45 4 8 square.png s.png
4.Type "c" to display convolved image. Type "r" to display original image.
5.Type "w" to write displaying image to [output image].
6.Type "q" to quit.

## 3.functions

1. bool read()
2. bool readKernel(char* filename)
3. void flipFilt(double ** kernel)
4. void filtImage(unsigned char * image,double ** kernel)
5. double** gaberFilt( double theta , int sigma , double period )
6. bool write()
7. void displayImage()
8. void resetPixels(unsigned char* srcPixels,unsigned char* desPixels)
9. void handleKey(unsigned char key,int x,int y)

## 4.Normalization method and Boundary Mechanism

1. Normalization method

there is a possibility that the result pixel value is out of boundary, so if it is larger than 255 , convert it to 255, if it is less than 0 , convert it to 0.

```
int result = (sum / factor);
result = result>255? 255: result;
result = result<0? 0: result;
image[i*xres*channels+ j*channels + k] = result;
```

2.Boundary mechanisim

If filter is outside of the image pixel, set the weight to 0.

in function: void filtImage(unsigned char * image,double ** kernel)

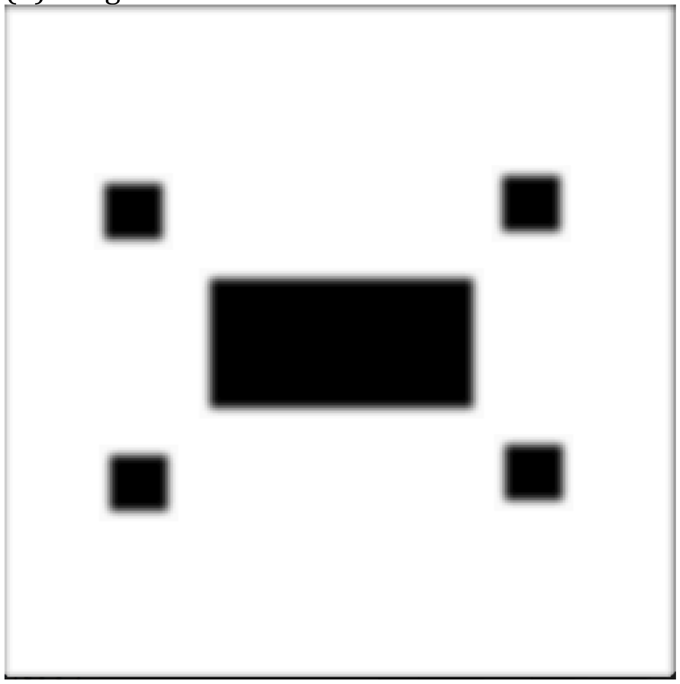
Code is as following:

```
if(edgeH>=0 && edgeH < yres && edgeW >=0 && edgeW < xres)
        sum += kernel[y][x]*tempPixels[(i+y-N/2)*xres*channels
+(j+x-N/2)*channels + k];
}
```

## 5.Results

1.squares

(1)using bell9.filt

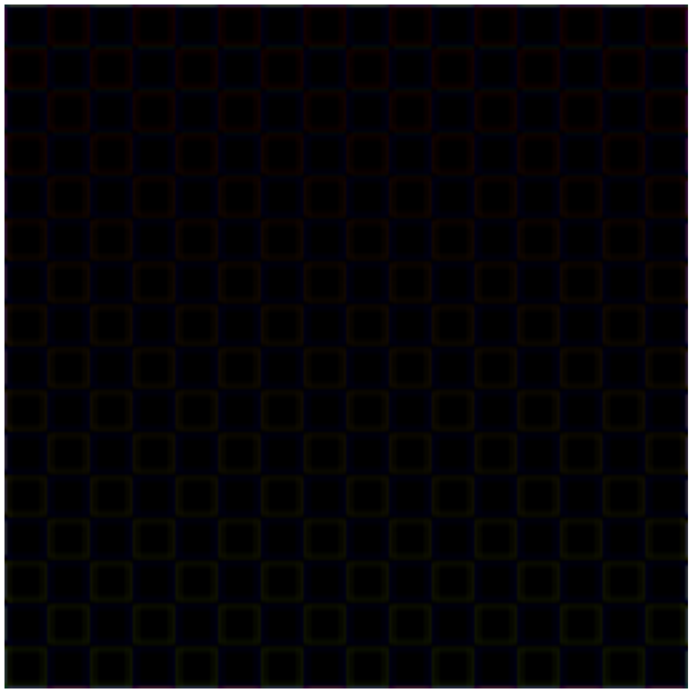


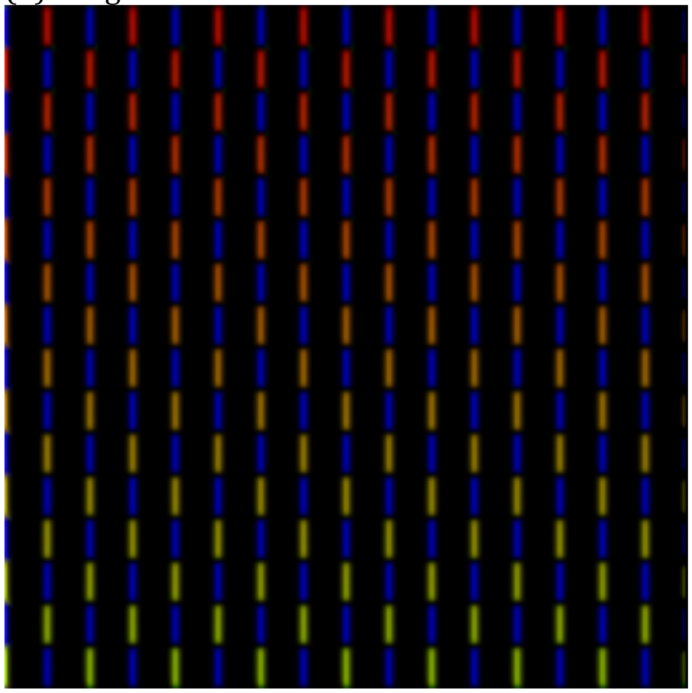
(2)using box9.filt

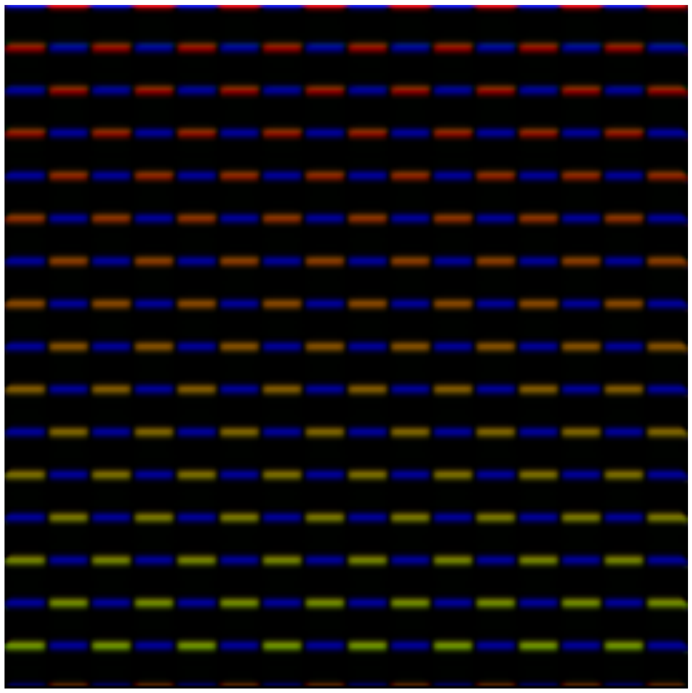
(3)using tent9.filt

2.checkers.png

(1)using hp.filt

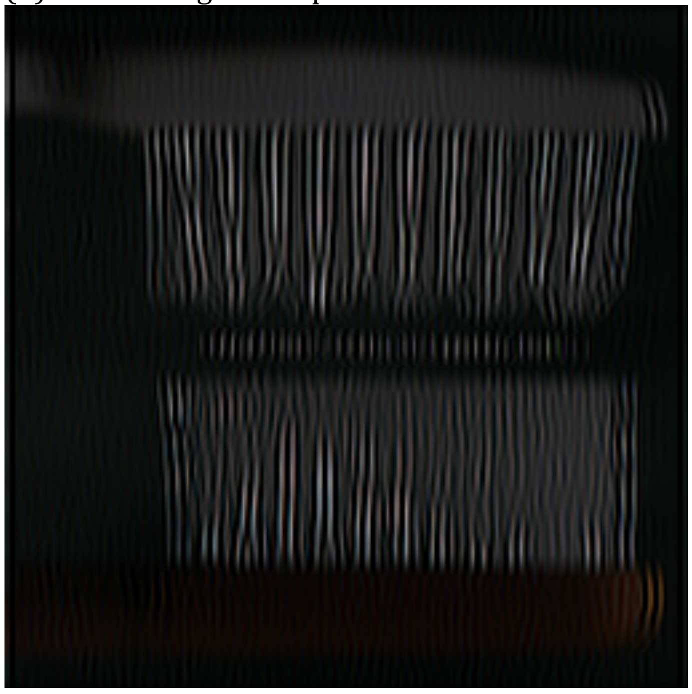

(2)using sobol-horiz.filt


(3)using sobol-vert.filt



3. gabor filter

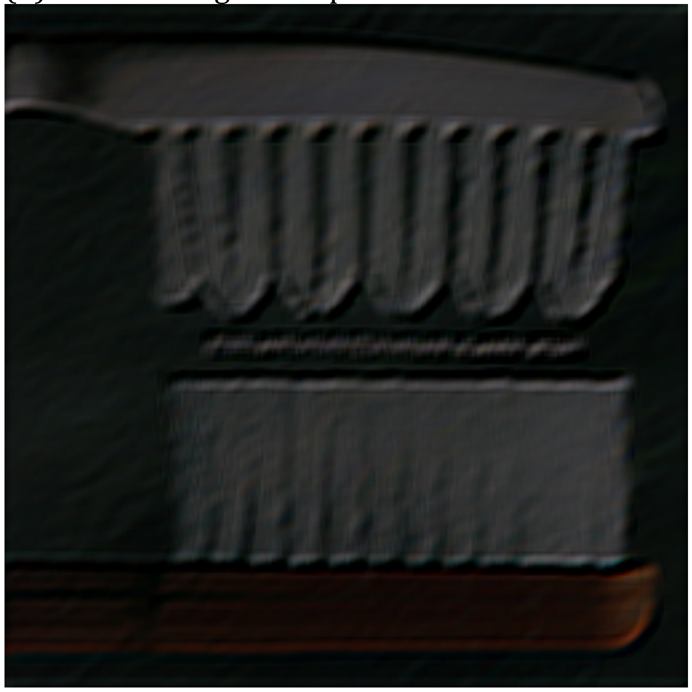
(1)theta = 0 sigma = 4 period = 4



new filter is:
0.367878,-0.00109375,-0.535261,0.000468136,0.606531,0.000468136,-0.535261,-0.00109375,0.367878,
0.457831,-0.0013612,-0.666143,0.000582605,0.75484,0.000582605,-0.666143,-0.0013612,0.457831,
0.535259,-0.0015914,-0.7788,0.000681134,0.882497,0.000681134,-0.7788,-0.0015914,0.535259,

0.587867,-0.00174781,-0.855344,0.00074808,0.969233,0.00074808,-0.855344,-0.00174781,0.587867,
0.606528,-0.0018033,-0.882496,0.000771826,1,0.000771826,-0.882496,-0.0018033,0.606528,
0.587867,-0.00174781,-0.855344,0.00074808,0.969233,0.00074808,-0.855344,-0.00174781,0.587867,
0.535259,-0.0015914,-0.7788,0.000681134,0.882497,0.000681134,-0.7788,-0.0015914,0.535259,
0.457831,-0.0013612,-0.666143,0.000582605,0.75484,0.000582605,-0.666143,-0.0013612,0.457831,
0.367878,-0.00109375,-0.535261,0.000468136,0.606531,0.000468136,-0.535261,-0.00109375,0.367878,

(2)theta = 45 sigma = 4 period = 8



new filter is:
-0.140218,-0.329518,-0.501886,-0.586902,-0.540831,-0.373594,-0.146011,0.0621155,0.191806,
-0.399169,-0.566969,-0.633787,-0.54745,-0.316783,-0.0151461,0.254291,0.410342,0.429387,
-0.528748,-0.561351,-0.433215,-0.151028,0.205378,0.515773,0.679251,0.662896,0.509158,
-0.399282,-0.240046,0.0667188,0.442493,0.760933,0.908898,0.844852,0.616274,0.326699,
-0.0476936,0.247215,0.598983,0.887983,1,0.887983,0.598983,0.247215,-0.0476936,

0.326699,0.616274,0.844852,0.908898,0.760933,0.442493,0.0667188,-0.240046,-
0.399282,
0.509158,0.662896,0.679251,0.515773,0.205378,-0.151028,-0.433215,-0.561351,-
0.528748,
0.429387,0.410342,0.254291,-0.0151461,-0.316783,-0.54745,-0.633787,-
0.566969,-0.399169,
0.191806,0.0621155,-0.146011,-0.373594,-0.540831,-0.586902,-0.501886,-
0.329518,-0.140218,