Ben Hixon
CSCI 493.69: Computer Vision
Final Project Essay

Iris Recognition with Gabor Transforms

1. Theory

John Daugman patented the use of Gabor Wavelet transform to test if two pictures of eyes are the same. Given a picture of an iris, we can use edge detection, circular Hough transforms, or snakes to find the boundaries of the iris and pupil, resulting in a localized iris. From that, we can convert the image to polar coordinates and sample the image at a number of different points (r_i ,Θ_i). These represent the midpoints of patches of the iris, which is why Daugman calls it patchwise sampling. We turn these sampled points into a string of bits called the IrisCode representing the iris, and compare two IrisCodes by finding the fractional Hamming distance between them, which is the number of different bits divided by the length of the IrisCode. If the Hamming Distance for two irises is less than 0.3, we can say with high confidence that they are the same iris. However if it's greater than 0.3 than we can say with high confidence that the irises do not match, because they've 'passed' a test of statistical independence and their IrisCodes do in fact represent independent irises. This method of matching irises using a failure of a test of statistical independence appears to be the most commonly used today, at least according to Daugman.

The IrisCode representing an iris is a string of bits. To extract this code from the image, we use a Gabor wavelet transform to map each sampled patch to a quadrant of the complex plane. The Gabor transform is an integral of the form

$$G(r_0,\theta_0) = \iint_{\rho,\phi} I(\rho,\phi)e^{-i\omega(\theta_0-\phi)}e^{-(r_0-\rho)^2/\alpha^2-(\theta_0-\phi)^2/\beta^2}\rho d\rho d\phi$$

where I(ρ,φ) is the image. The complex exponential, called the 'Carrier', makes the integrand a sinusoid having a real and imaginary components., and depends on the parameter ω. This sinusoid is multiplied in the integrand by a Gaussian, called the 'Gaussian envelope,' which depends on the parameters α and β. For a sampled point (r_0,Θ_0), we can calculate G(r_i ,Θ_i) for a variety of parameters α, β, and ω to get different values. We integrate over the whole image, resulting in a complex number of the form A+Bi. If A+Bi is considered as a vector in the complex plane, then it is called a phasor, which is why Daugman refers to this method as mapping to a phasor. The useful information from the transform is what quadrant the phasor is in: each quadrant of the complex plane can be represented by two bits: the sign of the real part, A, and the sign of the imaginary part, B. For example, if after finding G we get an A+Bi where A>0 and B<0, then the result is in the fourth quadrant and we append (1,0) to our IrisCode.

To compare two IrisCodes, we just find the Hamming Distance by XOR'ing two IrisCodes. The HD is the number of different bits in the two codes, normalized by dividing by the length of the bitstring. If the HD is low, the two IrisCodes correspond to the same eye.

2. Code

For the coding portion of my project, I implemented the Gabor wavelet transform, the extraction of IrisCodes, and the computation of the fractional Hamming Distance test to compare two eye images. I used the Image.cpp, Image.h, and Pgm.cpp files working with *.pgm files that we've been using through the semester. I was unable to find two different clear iris images for the same eye, so I calculated them for two different eyes and took the Hamming Distance between them. I did not code the localisation, but rather used as input images where all but the iris had been cropped or blacked out, though the localisation could easily be found using methods we've discussed in class.

While the Gabor transform can be efficiently calculated, I simply coded Daugman's formula directly, using the following c++ code for the integrand:

```
double envelope = exp( -1 * (r0-rho)*(r0-rho)/(alpha*alpha) - (theta0-
      phi)*(theta0-phi)/(beta*beta) );
double t = -1 * omega * (theta0 - phi ) ;
complex<double> carrier ( cos(t), sin(t) );
g_integrand = ival * carrier * envelope;
```

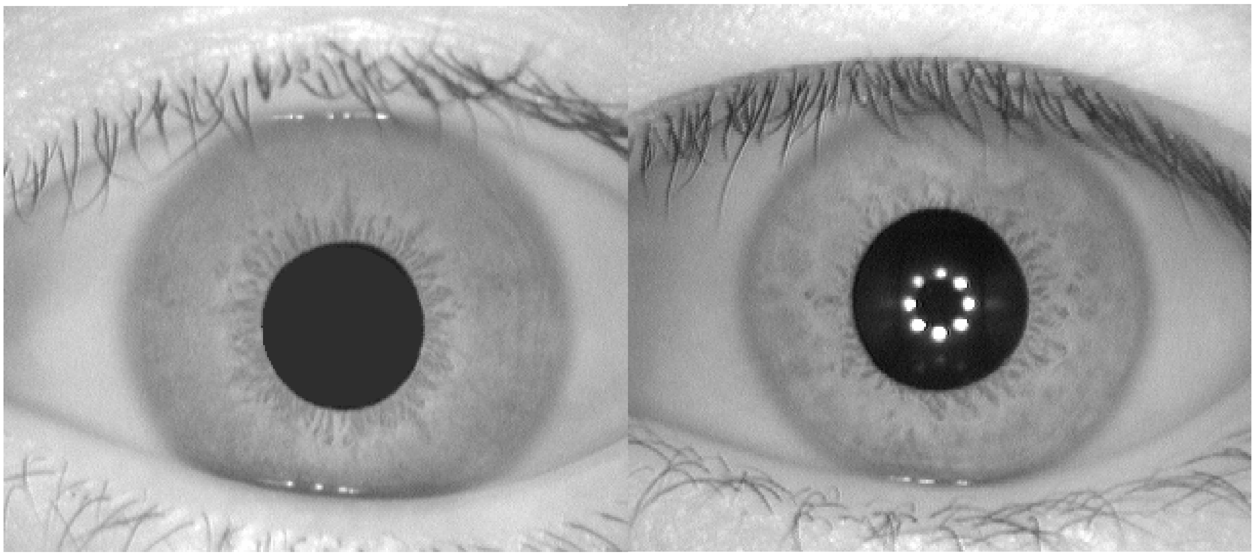
so that the integrand `g_integrand` is given by the product of the pixel value `ival`, the complex carrier function `carrier`, and the gaussian envelope `envelope` evaluated at given values for the sampled points r0 and theta0 and the Gabor parameters alpha, beta, and omega. The most difficult part was the image point sampling to find (r_0,Θ_0), since Daugman does not discuss how exactly he samples the iris, just that the sample patches are evenly distributed, so I tried to evenly distribute over the image. I also did not let the window size alpha and beta vary, only letting the value for omega vary from 0 to 2π. Again, Daugman did not specify exactly how he let these parameters vary; I suspect that he just played with the Gabor parameters until he got the steepest histogram possible for hamming distances between two iriscodes. If I had a database of iris images I could do the same with this code. To get the actual IrisCode bits, I just tested the sign of the real and imaginary components of the result of the Gabor transform:

```
G = Gabor(im, theta0, r0, center, omega, alpha, beta);
if( real(G) > 0 ){ ic.push_back(1); }
else{ ic.push_back(0); }
if( imag(G) > 0 ){ ic.push_back(1); }
else{ ic.push_back(0); }
```

where G is a complex<double>. Calculating the fractional Hamming Distance is easy; after storing the two iriscodes in two different integer vectors ic1 and ic2, I just find the number of different bits.

3. Results

For the following iris images



which I downloaded from Daugman's website (I manually localised the iris in each first), I found a Hamming Distance of 0.647, which means that they are not the same eye. Here's a screenshot of the program running with some of the command-prompt output:

```
eniac.geo.hunter.cuny.edu - PuTTY
-bash-3.2$ g++ -o iris iris.cpp Image.cpp Pgm.cpp
-bash-3.2$ ./iris eye1.pgm eye2.pgm
Read eye1.pgm and eye2.pgm
Calculating IrisCode for eye1.pgm
Center is 79.1765,109.063
Gabor of bitpair 500 at patch (r0,th0,omega)=(64.1993,0.0651011,3) is (-6427.12,19930.3)
Gabor of bitpair 1000 at patch (r0,th0,omega)=(73.3443,-1.30786,0) is (2.19653e+06,0)
Calculating IrisCode for eye1.pgm
Center is 101.57,112.376
Gabor of bitpair 500 at patch (r0,th0,omega)=(67.9387,2.1577,3) is (-993.597,-42516.6)
Gabor of bitpair 1000 at patch (r0,th0,omega)=(70.0903,-2.87551,0) is (2.29579e+06,0)
Gabor of bitpair 1500 at patch (r0,th0,omega)=(94.4488,-2.16177,3.75) is (123819,-402448)
int HD = 1018
HDf = 0.647248

Fractional Hamming Distance of eye1.pgm and eye2.pgm is: 0.647248
-bash-3.2$
```

The iriscodes for the two images are saved in text files, "imagename.pgm.iriscode.txt", included with this essay.