Iris recognition using the JAVAVis Library

V.Larrumbe Hidalgo, L.Martin García, M.Taboada Lorenzo

Abstract --**This project has been created to develop a biometric identification system through a man's iris using a computer to perform the processing of the pictures. To develop this application, and to differentiate the project from others who have already implemented, we have used the image processing library JAVAVis and JAVA as a programming language.**

Keywords - - **biometric, image, iris, wavelet**

I.INTRODUCTION

Biometrics is based on the premise that each individual is unique and has distinctive physical traits or behaviours, which can be used to identify or validate him/her. The Iris human structure is only one per person and it forms a very complex system unchanged during any person's life.

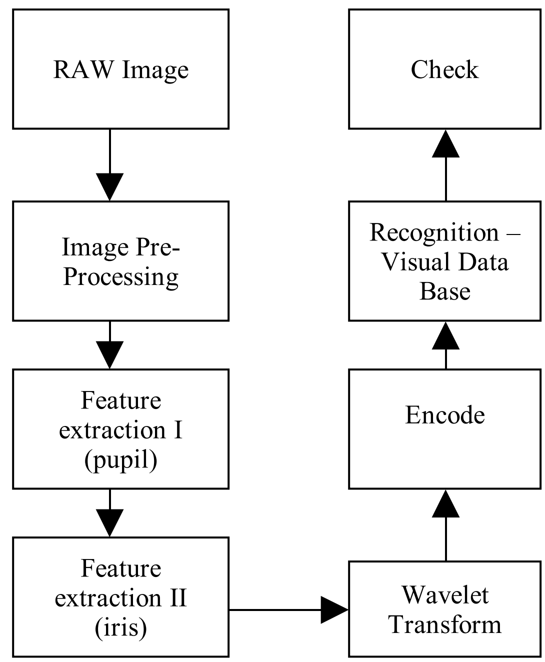
For several years the human iris has been using for user authentication. To this, an image capture is necessary. It must be transform into black-white ones, in an environment properly lighted, using a high-resolution camera.

Usually this is done by looking through the a fixed camera lens, a person simply positioned in front of the camera and the system captures the iris image, this image is processed to extract Biometry optical iris patterns, which in turn are subject to mathematical algorithms to produce a sufficient amount of data to identify ambiguously of the individual concerned. That is compared with another sample taken earlier and stored in the database system, so that if both match the user is considered authenticated successfully, the probability of a false acceptance is the lowest of all biometric models.

We start from the idea that in a future the application could be used by capturing images through a webcam, but now and ought to technical limitations, images prosecuted are .JPG files previously entered into the computer, which must have a high resolution to achieve a smooth running of the program.
The image, of which we are referring to, is digitally processed to obtain a code that characterized it. They are used in the treatment pre- processing algorithms to enhance image quality and thus, extract higher quality information and therefore more accurate.

The code that characterizes an iris image is obtained by applying Haar wavelet transform type. The iris code is composed of 366 bits and thus obtained 1.5031x10110 different codes. Evidence of encryption algorithms have been conducted on internet images.

Phases for the recognition of the image:



II.PHASES OF THE IMAGE PROCESSING

1. Capture the image.
2. Pre-processing image with the help of JAVAVis library.
3. Removing the area of interest.
4. Haar wavelet application.
5. Compare with codes from the database.

I. Capturing the image.

At the beginning of the project and facing to a future that might be applicable and useful in a real way, there was talk of obtaining images with pictures taken directly to a person's iris with a webcam in real time, but the technical features that were necessary, and those that we did not have, what prevented. The final solution was to derived images from previously saved files on the computer with the extension .jpg

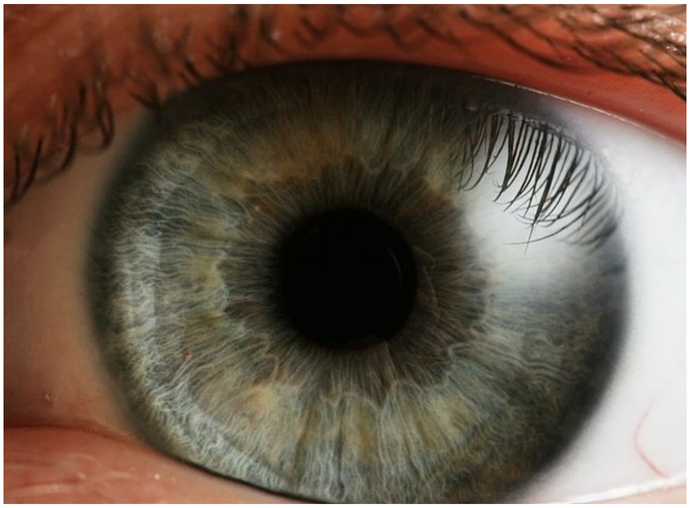


Fig 1. Photo of an eye in JPG format

II.Pre-processing image with the help of JAVAVis library.

Once we have the eye image in a format in which we can work, we begin the most important phases in the processing of the image.

The first one is to give the image, obtained in color, in a 256 gray scale. This process is made in order to, in further phases, be able to stand out the most characteristic features of the image. For that, we use the function given by the java library in which the project is based on, FColorToGray. This function takes as input parameters the image that is going to be processed and the type of gray image, that is, if the image is byte, bit o real type. For our case, the image is in bytes. As output parameter, we will obtain the image converted in a gray scale.

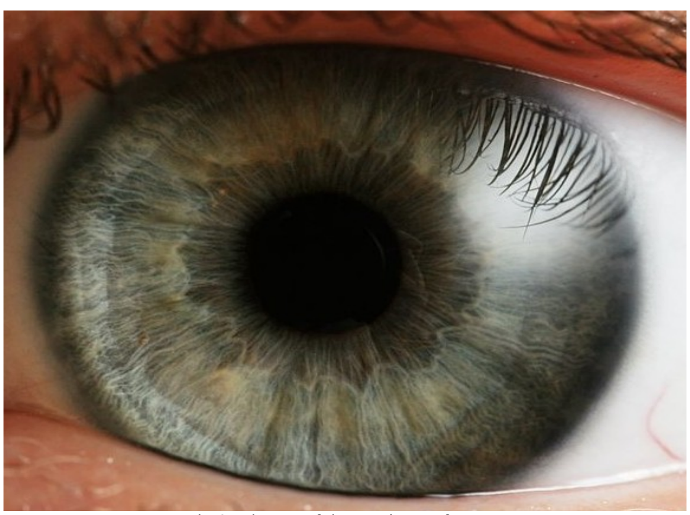


Fig 2. Picture of the eye in JIP format

We do this with the help of a simple conversion:

JIPImage jipImage = jipToolKit.getColorImage(imagenJPG)

Next, it will be necessary to apply a new filter called Median Filter. This filter is used to reduce the noise existing in the digital images, naming noise to any pixel value of an image that does not concern exactly with the reality. This happens most of the time when acquiring an image using an electronic capturing set, so noise acquired at the transmission stretch, due to interferences or errors in the bit information transmission, is needed to be added to that type of noise.

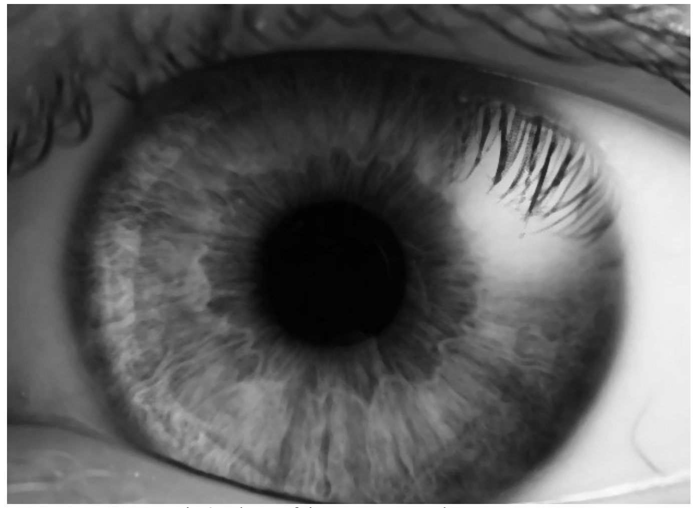


Fig 3. Photo of the eye converted to gray.

JIPImage imageGray =fColorToGray.processImg(jipImage)

Therefore, it is necessary the use of the FSmoothMedian function, returning the image with the edges of the two most prominent areas, even more emphasized, for them to be easier to be distinguished.

Now is when the first of the two binarizations made in the image is executed. The aim of the binarization is first, to distinguish the edges of the pupil so the values applied at the binarization exist exactly to emphasize in obtaining it, and second. To obtain the candidate circles that will be the pupil.

Binarization consist in delimiting through two values, the pixels that will have to be transformed in 0 (white) or 1 (black). These values have to be among 0 and 256, and are not inserted randomly because they are different for each image. To get those values it will be necessary to make the image histogram which is a function that represents graphically the intensity of the image pixel (gray intensity).

The obtained values are recuperated by means of the “observation” of the limit points where the values of the image are launched. When introducing these values in the binarization function, the image will get transformed into a black and white image.

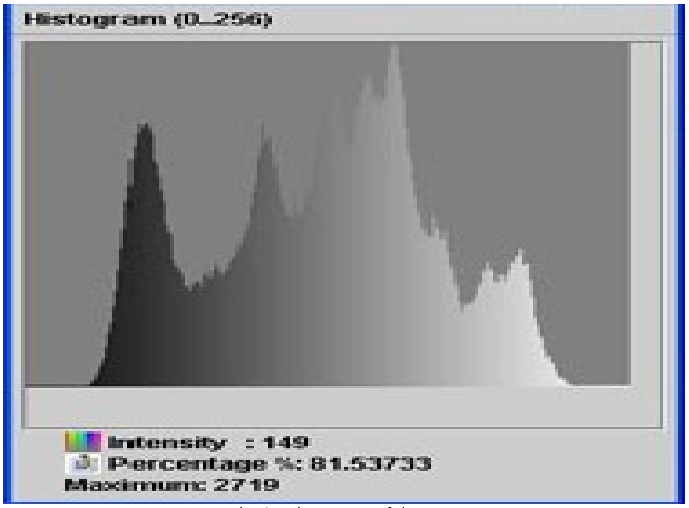


Fig 4. Histogram of the eye

Fig 5. Eye with the medium filter

```
FSmoothMedian fsmoothMedian = new FSmoothMedian()
JIPImage imgFiltroMediania =
fsmoothMedian.processImg(imageGray)
```

Now we can obtain the limits of the pupil well defined, because they are the ones that have been emphasized at the binarization. The eye pupil that is being processed will get totally white at the end.

To execute the binarization, the java library implements the FBinarice function, that gets a byte image and it is converted in a binary one (with 1 and 0 type values). The pixels which gray tones are among the given values, will get the value of "1", and the others, the value of "0".

Focusing again in the object of this first binarization, that is to find the limits under the pupil is positioned, we have to keep in mind that this binarization, having launched the function, will have obtained a "0" en each pixel of the pupil. Because this can occurs in some deformed characteristic of the iris, we will understand that the pupil will be determined because, between the possible candidate circles, it will be the one with more radius, made entirely of white distinctives (100% white).

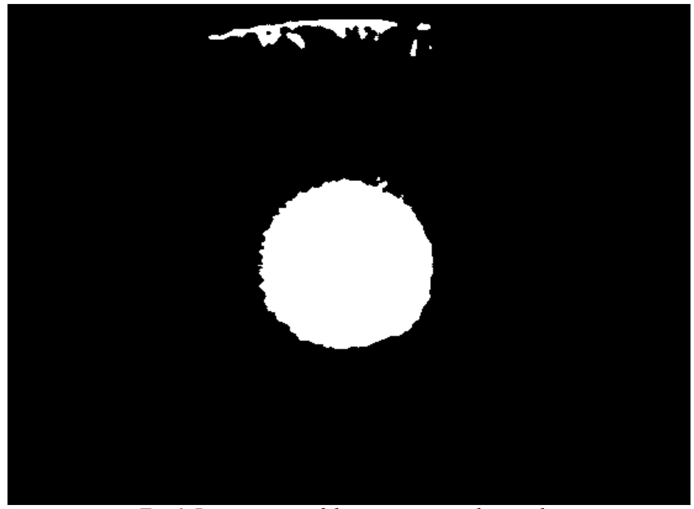


Fig 6. Binarization of the image to get the pupil.

```
FBinarize fBinarize = new FBinarize();
fBinarize.setParamValue("u1", min);
fBinarize.setParamValue("u2", max);
```

Also, we have talked about "candidate circles". Candidate circles are obtained from the image through the FHoughCirc function. This function obtains the number of circumferences of a bit type of the image using the Hough Transform for circumferences. It returns a Poly image with the "draw" of the image and two output parameters. In first place ncir, that is the number of circumferences found, and in second place, the circumstances, that are just a circumference type objects vector which contains their centres and their radius.

The input parameters are the minimum and the maximum possible radius of the circumference and the necessary number of votes to carry out the condition of being circumference. In our case, this number of votes is 32, which coincides with the number that the function takes by defect.

Fig 7 . Image Canny of the iris.

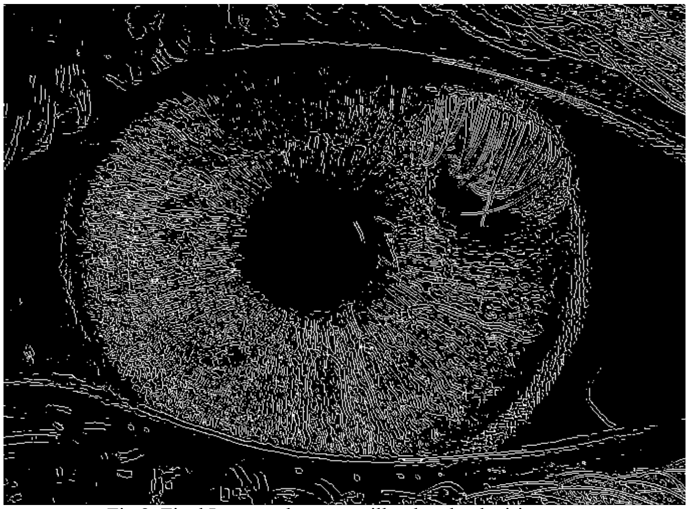


Fig 8. Final Image, where we will only take the iris part.

Considering that once made all these filters we will have the limits of the pupil, it is important to say that in an human eye, the “x” coordinate of the pupil centre coincides with the “x” coordinate of the iris centre, but doesn’t occurs the same with the “y” coordinates.

It seems clear that the next step is to be able to distinguish the iris limits. For that, we make a new binarization to the initial image (once have made the colour to gray change filters and the median) with the difference that now we previously apply a new filter, the Canny filter.

Canny is a filter made to detect edges, which maintains vital criteria for the case that concern us. The first of them is the reliability in detecting points of outline.

Therefore, the image that will process in the binarization will have even more emphasized edges. As having said previously, binarization is a process that is made after obtaining the threshold values limited by the hestogram, but this time we will focus on values that stand out exactly in the opposite way that we wanted in the last binarization, in other words, we want the pupil becomes totally black, and enhance the important points, therefore characteristics of the iris.

Once made the image transformation to a bite image, we will use again the FHoughCirc function, in order to obtain the candidate circles, now, keeping in mind the data thatwe already know about the “x” coordinate of the iris´ central point of the circle.

As well as with the pupil, with the iris the criteria is the same. It will be chosen the candidate circle which follows the condition of having the same “x” central coordinate as the pupil and the larger radius.

Once the two circles in which we are interested are localized and delimited, we take out the important circle that identifies a person, the one which corresponds to the iris, and the one whose top area limits with the obtained data in the last round, and the bottom area limit with the data obtained from the circle that belongs to the pupil.

III.Removing the area of interest

It is here where, we obtain "the donuts", namely, the iris.

At this point there the rest of the picture is not necessary for anything.

To obtain the information that it is wanted, namely, the characteristics that make each individual can be identified ambiguously through him/hers iris, it is necessary that the image will be process. Now there is a function that will allow us to stretch the iris.

*FRectifyOmnidir fRectifyOmnidir = **new** FRectifyOmnidir();*
JIPImage imgRec = fRectifyOmnidir.processImg(image);

Finally it is obtained as a result, an image that represents the context of the iris.

*FRectifyOmnidir fRectifyOmnidir = **new** FRectifyOmnidir();*
JIPImage imgRec = fRectifyOmnidir.processImg(image);

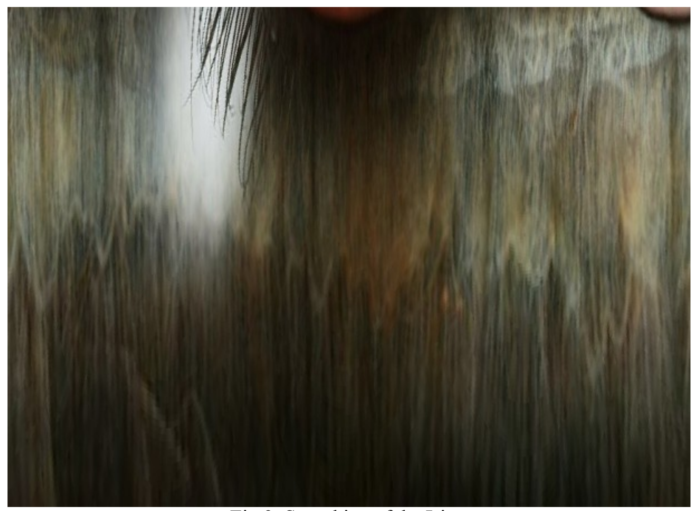


Fig 9. Stretching of the Iris.

Finally it is obtained as a result, an image that represents the context of the iris.

IV.Haar wavelet transform

With the image properly treated, it is just need to get that information in the form of securities registered values, in some way.

This is point where it is necessary to apply a Haar wavelet transform. These transformations performed consecutive iterations to achieve the level of detail required in each case. To obtain accurate information from the iris, there are needed five iterations performed as shown in the figure below:

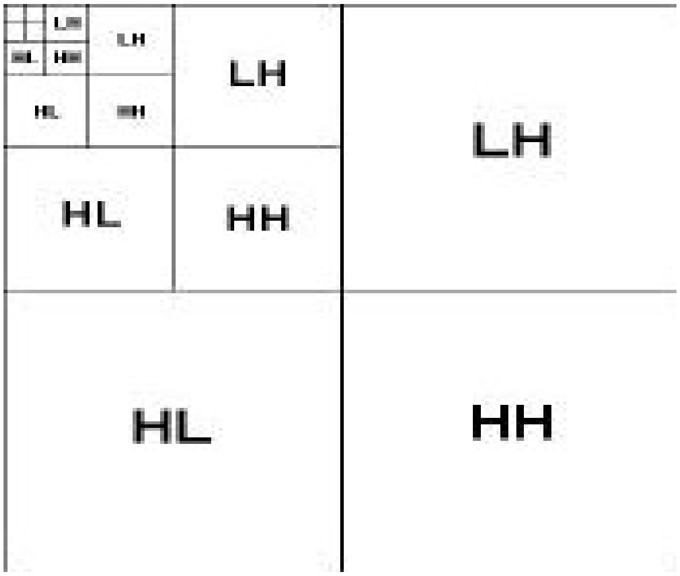


Fig 10.iterations performed to obtain accurate information from the iris.

It will give as a result some values processed as a form of vector and the really necessary ones are the first six positions. These positions correspond to 4HL, 4LH, 4HH, 5HL, 5LH, and 5HH, which are major points of distinction from the human iris.

In other words, the wavelet transform manages to assigned the image an univocal code carrying the biggest bites in each of the transformations upwards and towards the left and situating the smaller bites downwards and to the right. Therefore, we achieve that the distinctive characteristics of the person are found in the smallest cube located in the chart above.

In a very general way, the wavelet transform of a function f(t) is the decomposition of F(t) in an a set of functions $\psi_{s,\tau}$ (*t*), which form a base and are called the "wavelets". The wavelet transform is defined as:

$$W_f(s,\tau)=\int f(t)\psi^*_{s,\tau}(t)\,dt.$$

The wavelets are generated from the transfer y and scale change of another wavelet ψ (t) function, named the "mother wavelet" and it is defined as:

$$\psi_{s,\tau}(t)=\frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right)$$

Where *"s"* is the scale factor and τ is the transfer factor.

V.Compare codes with database information

This procedure is no different from any other comparison. It simply consist in taking each of those values obtained after the whole procedure and compare them with those registered in a database made for the project where they will find the data. In the database are recorded fields identification of the individual, the data to compare (4HL, 4LH, 4HH, 5HL, 5LH, and 5HH) and another field to indicate whether the eye is the left of the person or the right ones, because as we said earlier, the irises are not equal and therefore, it could be an incorrect identification.

Should you find the values obtained from processing in the database, it will be returned with the correct identification data of the person to whom it belongs, otherwise, it will be returned a message informing on the Non-identification of the individual, who is not in the register.

III.PROBLEMS

During the development of the application, we have found many problems, that have made us delay more of what we had expected we would:

- Schedules compatibility problems to bring the group together.
- We tried to make use of photos taken with the camera, but they wouldn't work in the application, having finally to make use of photos downloaded from the Internet.
- It has been complicated to find enough information, due to there is no much material available about iris processing.
- At the time of executing the application from Eclipse, the total time of performance is nearly 30 minutes, making us waste so much time each time we needed to change something of the code.

IV.CONCLUSION

The recent advantages of the information technology and the increase of the security requests have entailed a fast development of biometric – technique – based personal identification smart systems. Biometric techniques use characteristic o physiological behaviours of each in his own individual to be identified.

Iris recognition is a very recent technique in the area of the personal identification and it is considered as one of the most reliable ways of biometrics.

The importance of iris lies mainly in the individual character for each person, being this on of the main advantages of its use in the area of the business security and bastion of its exit and propagation along numerous countries around the world. In the same way, it emphasizes as well its non – invasive character, known as the capacity to capture the biometrical features without the necessity of using a way that comes in contact physically with the analyzed subject.

Nowadays, several organizations use iris not just in the criminal investigation but as well in the control to restricted areas, frontier control, employee identification, financial security and other fields still today becoming.

REFERENCES

[1] A. Jain, R. Bolle & S. Pankanti, "Biometrics: Personal Identification in a Networked Society", Eds. Kluwer, 1999.

[2] T. Mansfield, G. Kelly, D. Chandler & J. Kane "Biometric Product Testing Final Report", Issue1.0, National Physical Laboratory of UK, 2001.

[3] A. Mansfield, J. Wayman, "Best Practice Standards for Testing and Reporting on Biometric Device Performance", National Physical Laboratory of UK, 2002.

[4] D. Yingzi, I. Robert, E. Delores, W. Thad & C. Chein-I, "A One-Dimensional Approach for Iris Identification", Electrical Engineering Department, United States Naval Academy, Annapolis, MD 21402 & Dept. of Computer Science and Electrical Engineering, Univ. of Maryland, Baltimore County, Baltimore, MD 21228.

[5] J. Daugman, "How Iris Recognition Works", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, No. 1, pag. 21-30, Jan. 2004.

[6] J.Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 15, pp. 1148-1161, 1993.

[7] J. M. Ali, & A.E Hassanien, "An Iris Recognition System to Enhance E-Security, Advanced Modeling and Optimization", vol. 5, pp. 93-104.

[8] A. Muroò, & J. Pospíil. "The Human Iris Structure and its Usages", Department of Experimental Physics, Natural Science Faculty of Palacký University, Svobody, República Checa, Mar. 2000.

[9] J. Daugman, "Biometric Personal Identications System Based on Iris Analysis", U.S. Patent, No. 5291560. Washington, U.S. Goverment 1994.

[10] R. Wildes, "Automated, Non-Invasive Iris Recognition System and Method", U.S. Patent, No. 5751836. Washington, U.S. Goverment 1998.

[11] M. Li,, T. Tieniu, Y. Wang & D. Zhang, "Personal Identification Based on Iris Texture Analysis" , IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 25, No. 12, pág 1519-1533, Dec. 2003.

[12] R. Almeyda, P. Lucerna, "Reconocimiento de Iris", 2004, Available:

http://iie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2004/recon_iris/index.htm