ISSN:2229-6107

INTERNATIONAL JOURNAL OF PURE AND APPLIED SCIENCE & TECHNOLOGY

E-mail :
editor.ijpast@gmail.com
editor.ijpast@.in

www.ijpast.in

INTERNATIONAL JOURNAL OF PURE AND APPLIED SCIENCE & TECHNOLOGY IJPAST

ISSN 2229-6107 www.ijpast.in

Vol 13,Issue 2.Dec 2022

A Novel Approach for IRIS Recognition through Machine Learning Techniques

M.Rohitha Sreya, Nandita Biradar, P.Manjusri Harshini, P.Chandana Lakshmi,

Guide:Dr.kanaka Durga Returi,

Abstract:

In this study, we demonstrate the viability and utility of several machine learning approaches for iris detection using photos taken with a smartphone camera. In the first step, the iris is located using the well-known Doughman's approach, and the eyelids are hidden using a clever edge detection technique. The retrieved iris area is then normalised in a previously unexplored manner by establishing an adjustable threshold. The next step is to use Har wavelets to deconstruct the normalised picture into the feature vectors. An equalisation of the histogram is conducted to improve the precision of the categorization. The collected feature vectors are then used to train a variety of classifiers, resulting in an accuracy of around 99.7% during the training phase and 97% during the testing phase. Finally, the suggested strategy is compared to others that have been used on the same dataset, and it is shown to be superior in most cases.

Key Work:

Eyelash removal, smartphone iris scanners, machine learning, and the visible spectrum.

INTRODUCTION

The iris of the human eye is famous for being one-of-a-kind, fixed, and unmoving [1]. Researchers in fields such as bioinformatics, cryptography, computational intelligence, etc., have found iris identification to be an interesting topic to study. There have been several effective strategies used so far. These methods may be split into two groups: those that use machine learning and those that don't. Iris pictures taken by an NIR (Near infrared) camera are employed in these methods because they allow for excellent texture visibility even in highly pigmented areas [2]. This means that the extracted iris area has more precise data, indicating higher recognition probabilities. However, the aforementioned camera's extensive setup is problematic, particularly when portability and ease of use are factors. However, camera-equipped cell phones are now affordable for almost everybody. The sole drawback is that, unlike NIR cameras, they record pictures in the visible light spectrum, which results in less-detailed iris photos. So, therefore, the question arises, "Are they good enough for iris recognition?" There are several researches that provide supportive answers to this topic [2,8].

One important point to note is that none of the aforementioned methods really put in any work to prove that machine learning methods will work or even be useful with the iris datasets collected by smartphones. This is significant since extremely excellent results have been obtained using machine learning approaches on datasets taken with NIR cameras [3]. The issue of whether or not ML approaches may be used in this context remains unsolved, since iris photos captured in visible light are anticipated to yield comparatively fewer

Department:CSE

Malla Reddy College for Engineering and Women , Hyderabad , Telangana.

information. In contrast, the work by Raja et al. [4] employed the Sparse Reconstruction Classifier in conjunction with K-means clustering, yielding a negligible proportion of false-positive results (EER) (Equal Error Rate). That is to say, it provides a strong hint, but it does not evaluate other machine learning methods or provide any more details about their potential usefulness. We expand previous research on the use of machine learning methods to iris detection by analysing photographs of the eye taken with a smartphone in the visible light spectrum. To do this, we create an all-encompassing segmentation and feature extraction method and experiment with feeding the same derived features into several classifiers. At last, we evaluate the feasibility of applying machine learning methods to databases that were acquired using smartphones by comparing the classification accuracy of the trained classifiers.

CONNECTED TEXTS

UBIRISv1 [5], UBIRISv2 [6], MICHE [7], etc. are publicly accessible datasets that include iris pictures in the visible light spectrum, and they have been used in a number of different research projects. Provence et al. [8] explored the difficulties of iris detection when dealing with unconstrained iris pictures in visible light. Errors are most likely caused by blurry iris pictures and separate segmentation and noise identification processes. Best lighting setups for visible light iris pictures were investigated by Santos et al. [9]. Using deep sparse filtering on the visible spectrum iris dataset VSSIRIS, Biolabs, Raja et al. [10] found very encouraging results (EER 2%). In a related but distinct research, Gronkiewicz et al. [2] used a newly constructed dataset to conclude that iris photos taken with a mobile phone provide enough visibility of iris texture features for all levels of pigmentation.

They defended the photos' compatibility with existing iris recognition technologies as Veri Eye [11], MIRLIN [12], OSIRIS [13], Agricore [14], and so on. The accuracy these algorithms provided for the dataset was more than 95%. The use of machine learning methods in iris recognition has also shown promising results. Researchers De Marsico al. [3] examined many iris identification machine learning methods. Specifically, the NIR camera pictures that make up the CASIA-Iris [15] dataset was employed in these analyses. When comparing methods, the combination of Support Vector Machines and Hamming distance achieved 99% accuracy by Rai and Yadav [16].

Dataset (Third)

This study made use of the dataset created by Trokielewicz and co-workers [2]. All the photos were taken using an iPhone 5s (8 megapixels, f/2.2), and there was a total of 70 people in the study. Two sessions yielded a final collection of around 3192 photographs. Eyes were scanned using these photographs. More importantly, as far as we are aware, no machine learning techniques have ever been applied to this dataset.

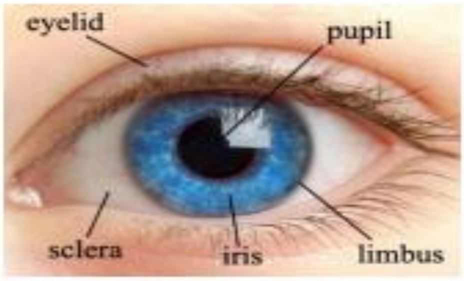

Fig. 1 The typical components in an eye image [20].

METHODOLOGY

Pre-Processing of Images

The RGB photos included in the data set were the only kind available. In order to go further, we had to transform it into a mono channel. It was determined that the best iris pattern could be seen using the red channel during conversion because the wavelengths associated with red light (closest to near infrared) are the longest in our visible spectrum [2].

Detection of the Iris

Initially, the tried-and-true Doughman's Integro-differential operator is used to extract the iris area. This is the

definition of the integro-differential operator in [17].

$$max_{(r,x_o,y_0)} \left| G_\sigma(r) * \frac{d}{dx} \oint_{x_o,y_0} \frac{I(x,y)}{2\pi r} ds \right| \quad (1)$$

If we replace I (x, y) with the eye image, r with the search radius, G(r) with the Gaussian smoothing function, and s with the shape of the circle centred at (xo, y0), we get the circle with a radius of r and its centre at (xo, y0). Changing the radius and centre x and y location of the circular contour, the operator seeks the circular route where the highest change in intensity occurs. It all starts with the iris border being concentrated there since that's where the largest gradient is. Pupillary boundaries are then identified by a thorough search. The Gaussian smoothing function was set at variance (= 0.5). The picture was reduced in size to determine its coordinates (r, xo, y0), which allowed for a quicker runtime. The original image's coordinates and radius were derived by scaling these values. This technician was able to pinpoint the iris area under normal conditions. However, it may fail regionally if reflection is present. Thus, we adaptively executed the search for a chosen set of locations within the iris area with pupil radius varied from 10% to 90% than that of the iris, for fine search rather than brute force. Thus, we were able to precisely locate the iris in every photograph of an eye in our database.

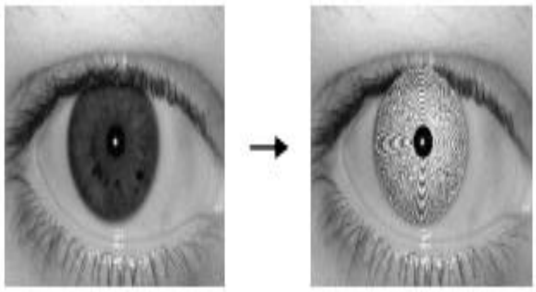

Fig. 2 Iris localization

Reduced Eyelid Movement

The iris's visible section isn't perfectly round. To some extent, it is hidden by the eyelids, which must be hidden. Here, we used a method based on Mask's [18]. The whole search area was split into two halves, the top eyelid and the lower eyelid. The search area's breadth is determined only by the distance between the iris's and pupil's radii. Canny edge detection was used first, then the gamma correction and hysteresis thresholding were used. Finally, the upper and lower eyelid line were extracted by radon transformation of the edge picture.

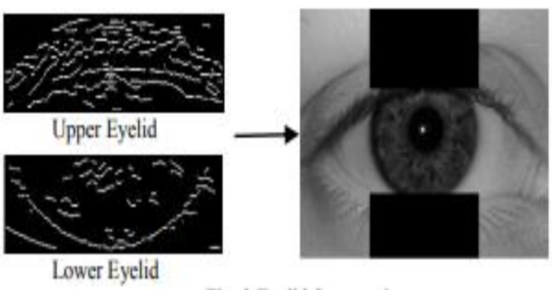

Fig. 3 Eyelid Suppression

Normalization

The iris has been effectively segmented, and the eyelids have been hidden. For further processing, we must now convert it to fixed dimensions. When doing so, we relied on Doughman's [17] homogeneous rubber sheet model, which has shown to be rather useful. Each iris-localized point is transformed into a set of polar coordinates (r,) in the homogeneous rubber sheet model, where r is on the interval [0, 1] and is an angle in the range [0, 2].

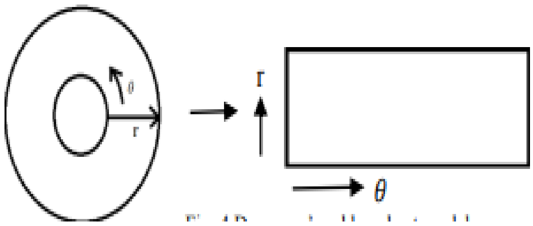

Fig. 4 Doughman's rubber sheet model

The remapping can be modelled as

$$I(x(r,\theta), y(r,\theta)) \rightarrow I(r,\theta) \qquad (2)$$

With,

$$x(r,\theta) = (1-r)x_p(\theta) + r x_l(\theta) \qquad (3)$$

$$y(r,\theta) = (1-r)y_p(\theta) + r y_l(\theta) \qquad (4)$$

Where I (x, y) is the iris region, (x, y) are the original Cartesian coordinates, (r, θ) are the corresponding normalized polar coordinates, (xp, yp) and (xi, yi) are the centre coordinates of pupil and iris boundary along the θ direction.

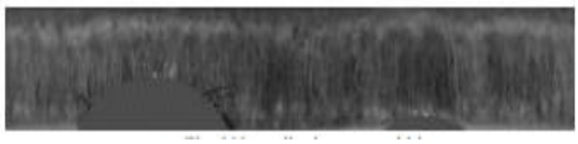

Fig. 5 Normalized segmented iris

Lashes Are Cut Off

Although we removed eyelashes as part of noise cancellation, we did so after normalisation. Since eyelashes are so visually distinct from one photograph to the next, it was no easy task to devise a technique to achieve this. Establishing a threshold was the most transparent strategy available. However, there is no assurance that every picture will benefit from using a strict threshold setting. The iris area was darker in some pictures than others. Since the threshold value is different for each picture, we needed to create an adaptive method to determine it. We achieved this by inspecting the iris image's histogram after it was scaled and normalised. Histogram analysis helped us zero down on the eyelashes' pixel values, since they tend to be the image's darkest areas. These pixel values were utilised as cut-offs for the occlusion detection process. The identified pixels were first reset to a value of 0, and then their contents were restored using data from neighbouring pixels that were not obscured.

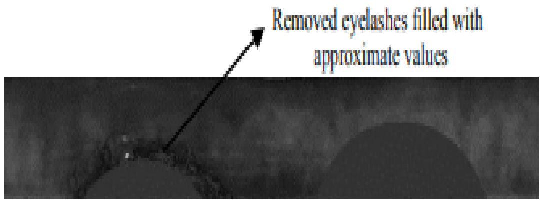

Fig. 6 Eyelash removal

Balance the Histogram

After the iris area has been segmented, normalised, and noise removed, the necessary texture and intensity information must be collected in order to train a classifier. The photos were first normalised, and then their histograms were equalised. This is because, as shown by the histogram analysis of the normalised picture, the image intensities are concentrated in a narrow area, making discrimination by the classifier more difficult. We observed that a 2% increase in identification and training accuracy was achieved using histogram equalisation.

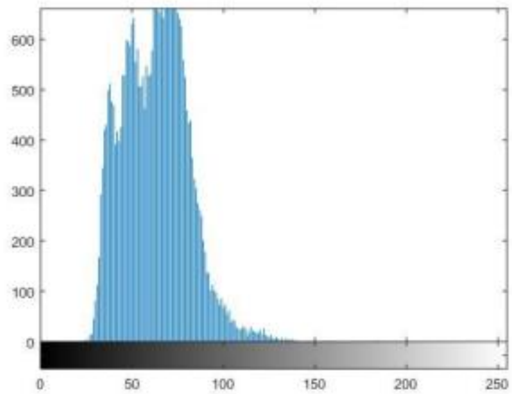

Fig. 7 Before histogram equalization

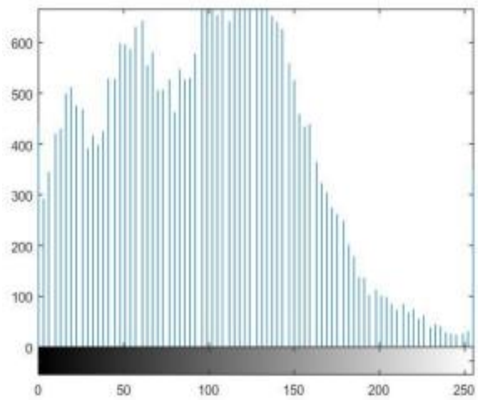

Fig. 8 After histogram equalization.

Separation of Distinct Features

There are many different kinds of intricate patterns on a normal iris, including arcing ligaments, furrows, ridges, crypts, rings, corona, freckles, and a zigzag collarette. It's a huge challenge to extract these intricate patterns. To that end, we choose to use the picture itself as a training data source. We have a standard 64 x 512-pixel picture at this time. We cannot use this volume of data for training the classifier without significantly increasing the time required for the process. In light of this, we need to expand our efforts. However, data loss is a potential risk while growing. Wavelet decomposition provides a solution because its frequency data is spatially confined, allowing characteristics with the same resolution to be compared and matched. It is already common knowledge that a 2-dimensional wavelet transformation applies to a picture and breaks it down into four parts (LL, LH, HL, and HH). The LL is sometimes referred to as an image approximation. In this case, LH indicates the horizontal detail of the picture, HL the vertical, and HH the diagonal. The LL coefficients hold the greatest power and information. Those are the ideals we have settled on. Two steps of wavelet decomposition are shown in Figure 9 for the iris picture. After 3 iterations of wavelet decomposition, we were able to get the picture suitable for training. Coefficients from LL3 were used, which had 512 features (8 x 64). Specifically, Har wavelets were employed for the decomposition. Dimensionally, the retrieved feature vector was 2D, with a size of 8 x 64. In order to train the classifier, it was first transformed into a 512-row 1D vector (Fig- 10).

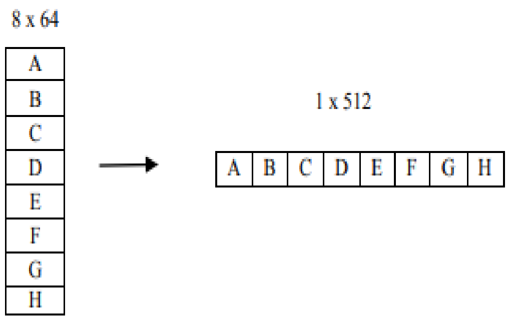

Fig. 10 1-d feature vector of size 512

Training Classifier

In the given database, we had eye images of 70 people. For training the model we took 5 images for each person and rest of the images were kept aside for testing the classifier. For training we used 5-fold cross validation method so that each image in the training set can be tested once against the others. We tried several classifiers and among them support vector machines, knearest neighbour, linear discriminants etc. showed great promise. The results are summarized in the next section.

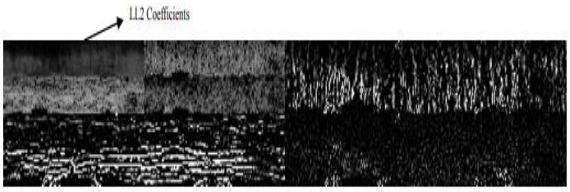

Fig. 9 Wavelet decomposition after 2 stages

Results

For training and testing several classifiers were used. Starting from decision trees, discriminant analysis, support vector machines, RUS Boosted trees, K-nearest-neighbours, Subspace KNN etc. The following table summarizes the best performed classifier accuracy:

TABLE I

CLASSIFIER ACCURACY

Classifier	Train Accuracy (%)	Test Accuracy (%)
SVM (Linear Kernel)	99.1	96.46
SVM (Quadratic)	**99.7**	**97**
KNN	99.4	95.1
LDA	99.4	94.28

Based on this information, it's clear that Support Vector Machines provide the greatest outcomes. The k-nearest-neighbours method also does quite well. Its accuracy is lower than that of SVMs, but training and testing it takes a fraction of the time. Linear Discriminant Classifier has a similar problem (LDA). Receiver Operating Characteristics (ROC) curve matrix acquired after best model training.

Here is an SVM with a quadratic kernel:

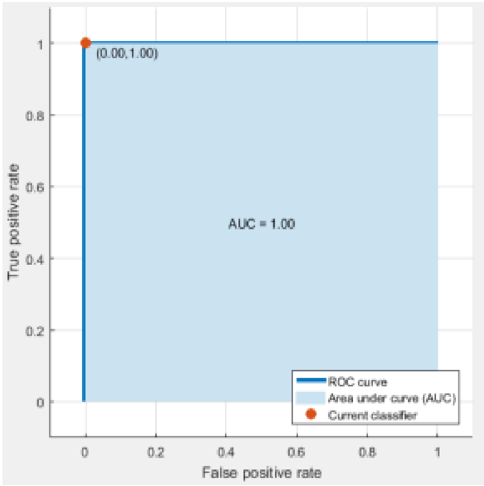

Fig. 11 ROC curve for SVM model

It is obvious from the outcome that our very first question regarding the viability and applicability of machine learning approaches on iris identification from smartphone collected visible light photos is effectively addressed by this study. Now if we compare our strategy and its outcomes with the other approaches that is previously used on the same dataset, we discover that our approach actually shows a tremendous potential.

TABLE II

DIFFERENT APPROACHES' RESULT ON SAME DATASET

Classifier	Accuracy (%)
VeriEye[11]	94.57[2]
MIRLIN[12]	95.63[2]
OSIRIS[13]	95.25[2]
IriCore[14]	**99.67**[2]
Proposed Method	**97**

Sources of Errors

The Major Source of errors in our findings is failure in segmentation of eye images. Despite our efforts, there were

Fig. 12 Failure in correct segmentation

one or two such images whose segmentation could not be done properly. And these are the images who were falsely labelled. Another source of errors are eye images with extremely dark pigments. As a result, it becomes very difficult to extract distinct information from them. Some other noise sources might be blurred images or images with excessive eyelid/eyelash occlusion.

CONCLUSION

In this study a machine learning based solution on iris detection from smartphone shot photos is suggested. With the findings above, this article successfully proved that in case of smartphone obtained visible spectrum iris photos, the machine learning algorithms are similarly as excellent as the other ones, in some instances even better. Still precision may be further increased. And in our results, accuracy mostly hinges on proper segmentation. So some robust measures may be adopted to enhance the segmentation outcome. In our approach we attempted to keep to certain fundamental segmentation methodologies. This was done bearing in mind their simple implementation. As smartphones of todays' are equipped with extremely excellent camera, the full recognition system shows tremendous potential to be deployed on these devices for recognition, security and identity purpose. Already Samsung® [19] has created a built in iris scanner which works for the individual who is utilising it. Our next job would be to construct a cloud based server where iris data can be readily supplied via the smartphone. The classifier will run on the server and the provided data will be matched and confirmed. Thus, by simply utilising the cell phones, it will be feasible to construct a whole security system.

REFERENCE

[1] J. Daugman, "How iris recognition works.," in IEEE Transactions on circuits and systems for video technology, 2004.

[2] M. Trokielewicz, "Iris Recognition with a Database of Iris Recognition with a Database of Iris Images Obtained in Visible Light Using Smartphone Camera," in The IEEE International Conference on Identity, Security and Behavior Analysis (ISBA 2016), Sendai, Japan, 2016/02.

[3] M. D. Marsico, A. Petrosino and S. Ricciardi, "Iris recognition through machine learning techniques: A survey," Pattern Recognition Letters, vol. 82, pp. 106-115, 2016. 36

[4] K. B. Raja, R. Raghavendra and C. Busch, "features, Smartphone based robust iris recognition in visible spectrum using clustered k-means," in Biometric Measurements and Systems for Security and Medical Applications (BIOMS) Proceedings, 2014 IEEE Workshop on, IEEE, 2014, pp. 15-21.

[5] H. Proença and L. A. Alexandre, "{UBIRIS}: A noisy iris image database," in 13th International Conference on Image Analysis and Processing - ICIAP 2005, Cagliari, Italy, Springer, 2005, pp. 970-977.

[6] H. Proenca, S. Filipe, R. Santos, J. Oliveira and L. A. Alexandre, "The {UBIRIS.v2}: A Database of Visible Wavelength Images Captured OnThe-Move and At-A-Distance," IEEE Trans. PAMI, vol. 32, pp. 1529- 1535, 2010.

[7] M. D. Marsico, M. Nappi, D. Riccio and H. Wechslerd, "Mobile Iris Challenge Evaluation (MICHE)-I, biometric iris dataset and protocols," Pattern Recognition Letters, vol. 57, pp. 17-23, 2015.

[8] H. Proenca and L. A. Alexandre, "The NICE. I: noisy iris challenge evaluation-part I," in Biometrics: Theory, Applications, and Systems, IEEE, 2007, pp. 1-4.

[9] G. Santos, M. V. Bernardo, H. Proenca and P. T. Fiadeiro, "Iris Recognition: Preliminary Assessment about the Discriminating Capacity of Visible Wavelength Data," in 2010 IEEE International Symposium on Multimedia, IEEE, 2010, pp. 324-329.

[10] K. B. Raja, R. Raghavendra, V. K. Vemuria and C. Busch, "Smartphone based visible iris recognition using deep sparse filtering," Pattern Recognition Letters, no. 57, pp. 33-42, 2015. [11] Neurotechnology, "VeriEye SDK, version 4.3".

[12] Smart Sensors Ltd., "MIRLIN SDK, version 2.23," 2013.

[13] G. Sutra, B. Dorizzi, S. Garcia-Salicetti and N. Othman, "A biometric reference system for iris. OSIRIS," 2014.

[14] IriTech Inc., "IriCore Software Developers Manual," 2013.

[15] http://www.sinobiometrics.com, "Casia Iris Image Database".

[16] H. Rai and A. Yadav, "Iris recognition using combined support vector machine and Hamming distance approach," Expert systems with applications, vol. 41, pp. 588-593, 2014.

[17] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence," 1993.

[18] L. Masek and P. Kovesi, "MATLAB Source Code for a Biometric Identification System Based on Iris Patterns," in The School of Computer Science and Software Engineering, The University of Western Australia, 2003.

[19] "http://www.samsung.com/global/galaxy/galaxy-s8/security/".

[20] "Courtesy: https://www.pinterest.com/".