Optimization method for non-cooperative iris recognition task using Daugman integro-differential operator

I Petrov and N Minakova

1Altai State University, 61 Lenin ave., Barnaul, 656049, Russia

E-mail: PetrovIV90@gmail.com

Abstract. We considered the issues arising when Daugman’s integrodifferential operator is used to localize iris edges. Daugman’s integro-differential operator is a widely common method to localize the iris, however, problems of the operator optimization poorly presented in the literature. The article considers existing methods of optimization of the operator applied at the iris recognition step. Based on the provided research we proposed using Nelder-Mead and Differential Evolution methods to optimize the integro-differential operator. The problems of the iris localization and their solving methods were considered. The article focuses attention on non-cooperative iris recognition. The very general boundary conditions based on iris’s anatomy which are not dependent on captured image properties were defined. The results of the comparative analysis of the accuracy and performance of selected optimization methods of Daugman’s integrodifferential operator were presented at the experimental results chapter. It was found that the Differential evolution optimization method gives fine performance and correctness. It was concluded that the Differential evolution is expedient as Daugman’s integro-differential operator optimization method.

1. Introduction

Biometric systems based on iris recognition are considered one of the most reliable [1-7]. The iris has a rich texture that considered being highly discriminative between any two individuals [1-9]. It is almost immune to change over time. However, automated identification of iris is not a trivial task.

One of the most difficult stages of iris recognition is localization. In 1993 John Daugman presented the iris recognition algorithm included localization step that became one of the most common methods of iris localization [9]. Daugman’s localization method is based on approximating the iris boundaries by two non-concentric circles. It reduces to maximizing a special integro-differential operator that works as a detector of circular boundaries. The stronger the contrast at the boundary of the circular region, the stronger the response of this operator. The Daugman operator is calculated by the following formula (1).

$$\max_{(r,x_0,y_0)} \left| G_\sigma(r) \ast \frac{\partial}{\partial r} \oint_{r,x_0,y_0} \frac{I(x,y)}{2\pi r} \, ds \right|,$$

(1)

where $G_\sigma(r)$ is the smoothing Gaussian function with a scale $\sigma$, and $I(x, y)$ is the intensity of image at coordinates $(x, y)$, $ds$ is the circular arc with radius $r$. Searching performs through the three parameters space includes center coordinates $(x_0, y_0)$ and radius $r$. 


2. Goal and objectives of the work
The search for the maximum of operator (1) is a complex task. The guaranteed way to find the maximum is to apply the operator over the entire image area and over every possible radius. However, this will make the recognition speed unacceptable for most real-world tasks. In some works, for example [10], the authors solve the problem by significantly limiting the scope of the operator. In most publicly available test dataset images are usually taken under the same conditions, which allow setting specific boundary conditions for each specific dataset. This method affects the universality of the algorithm, imposes additional requirements on equipment and shooting conditions, requires more cooperation from users. To solve such problems, it is necessary to use special optimization methods. The goal of this work is to search for reliable methods of optimizing the operator (1) in conditions close to non-cooperative [11] recognition. The optimization problem is complicated by the nature of the operator since it contains an image function that is not smooth and contains many local maxima. Search and development of new optimization methods is a relevant task. In order to achieve the goal of the work, the following tasks were set:

- explore existing localization methods based on operator optimization (1);
- study the existing methods of multidimensional optimization, select promising methods for comparative analysis;
- prepare a set of test iris images obtained in various conditions;
- identify the most common boundary and conditions;
- prepare the selected optimization methods to solve the operator optimization problem (1);
- analyze the accuracy and performance of the selected methods.

3. Optimization methods
The problem of optimizing the integro-differential operator wasn't detailed described in the original Daugman’s work [9, 12]. As an option for optimization, it was proposed to use the gradient descent method using a multigrid search. The algorithm starts with a rough approximation. As climbing along the gradient, the sensitivity of the operator is increasing in proportion to the steepness of the rise. The paper does not provide data on which value to choose as an initial assumption. In paper [13], it was proposed to use the so-called integro-differential ring (ItgDiff-ring). The search goes in a spiral from the starting point. If a point with a large value of the operator comes across the path, then the point is taken as optimal. The spiral search begins around this point. The search continues until more optimal values appear during the spiral traversal.

The authors first searched for the potential area of the iris by preliminary clustering the image on the area (eyebrows, skin, sclera). We can consider the localization of the iris borders by the integro-differential operator as a multidimensional search for the global extremum. In this paper, several optimization methods were chosen to test their applicability to this problem. Based on the characteristics of the objective function, the following methods were selected for comparative analysis:

- integro-differential ring (ItgDiff-ring);
- Nelder-Mead method;
- differential evolution.

The listed methods are direct search methods (zero-order methods). Zero-order methods use only information about the function itself without taking the derivative. Since the objective function has no analytical representation and is not smooth, the use of methods of optimization of zero-order, in this case, can be considered justified.

4. Nelder-Mead method
The Nelder – Mead method [14] is a method for optimizing the function of many variables. The method does not use function gradients. It can be used for non-smooth, non-differentiable functions. The method is based on operating with simplexes. Simplexes are a multidimensional generalization of triangles in two-dimensional space. The basic operation algorithm includes the following steps.
To maximize the objective function $f$ from $n$ parameters, a simplex containing $n + 1$ vertices $(x_0, x_1, \ldots, x_n)$ is initiated. Then the loop starts. At each iteration of the loop, the vertices of the simplex are sorted by the value of the objective function $f$: $f_0 < f_1 < \cdots < f_n$.

We distinguish three vertices: the vertex with the maximum (best value) of the function $x_b$, the vertex with the worst value of the function $x_w$ and the vertex with the value of the function following the worst value $x_i$. From the vertices, the center of mass $x_m$ is calculated, excluding the vertex $x_w$ (2):

$$x_m = \frac{1}{n} \sum_{i \neq w} x_i,$$

where $i \in (0, \ldots, n)$.

The current simplex is transformed based on a set of rules. First, the vertex $x_w$ moves by its reflection relative to the hyperplane consisting of vertices with the best values of the objective function $x_i$, $i \neq w$ (3).

$$x_r = x_m + \alpha(x_m - x_w),$$

where $\alpha$ is the coefficient, usually set to 1.

If the function value in $x_r$ is the best, then the stretching operation (4) is applied.

$$x_e = x_m + \gamma(x_r - x_m),$$

where $\gamma$ is a stretching ratio. It is usually accepted as 1.

If the function value in $x_r$ is the worst, then the shrink operation (5) is applied.

$$x_c = x_m - \beta(x_m - x_w),$$

where $\beta$ is the shrink ratio, it usually has a value of 0.5.

If the value of $f(x_r)$ is not the worst, but worse than the value of $f(x_i)$, then compression with reflection (6) is performed.

$$x_c = x_m + \beta(x_m - x_w).$$

If all the steps taken have not led to the desired result, then the entire simplex is compressed, redefining all the vertices except the vertex $x_b$ (7).

$$x_i = x_b + \rho(x_i - x_b),$$

where $\rho$ is the scaling factor, usually taken as 0.5.

In the last step of the iteration, a convergence check is performed. The size of a simplex can be chosen as a criterion for convergence. If the convergence criterion is not satisfied, then a new iteration is started.

5. Differential evolution
This is an optimization method [15] that works iteratively. The method is aimed at improving the solution according to specified quality criteria. The method, like the previous one, belongs to the class of heuristic algorithms. It can optimize complex nonanalytic functions for which correct differentiation is impossible, and which have many local minima. The objective function can be arbitrary not only continuous. Even though the method was developed to optimize continuous functions with continuous arguments, it can be applied to integer-valued arguments. The basic scheme of the method is shown in figure 1.

![Figure 1. Scheme of the differential evolution method.](image-url)
The boundary conditions for each argument of the objective function are determined at the initialization stage. The initial values of arguments are picked in the defined range. For each of the \( N \) vectors of parameters, mutation, recombination, and selection are performed. Mutation extends the search space.

For this parameter vector \( x_{i,G} \), three vectors \( x_{r_1,G}, x_{r_2,G}, x_{r_3,G} \) are randomly chosen such that the indices \( i, r_1, r_2, r_3 \) are different. Add the weighted difference of the two vectors to the third (8)

\[
v_{j,G+1} = x_{i,G} + K \cdot (x_{r_1,G} - x_{i,G}) + F \cdot (x_{r_2,G} - x_{r_3,G}),
\]

where \( i, r_1, r_2, r_3 \in \{1, 2, ..., N\} \) are randomly selected numbers. They should not coincide with each other, the \( F \) scaling factor is given by a constant from the range, and \( K \) is the combination factor. \( v_{j,G+1} \) is called a donor-vector.

At the recombination stage, successful solutions from previous generations are combined (9). The trial vector \( u_{j,i,G+1} \) is obtained from the elements of the target vector \( x_{i,G} \), and the elements of the donor vector \( v_{j,i,G+1} \). The elements of the donor vector enter the trial vector with probability \( CR \).

\[
u_{j,i,G+1} = \begin{cases} v_{j,i,G+1}, & \text{rand}_{j,i} \leq CR \cup j = I_{\text{rand}} \\ x_{j,i,G}, & \text{rand}_{j,i} > CR \cap j \neq I_{\text{rand}} \end{cases}
\]

where \( i = 1, 2, ..., N; j = 1, 2, ..., D, \text{rand}_{j,i} \in [0,1] \) is a random number, \( I_{\text{rand}} \in \{1, 2, ..., D\} \) is a randomly chosen index.

At the selection stage, the target vector \( x_{i,G} \) is compared with the trial vector \( u_{i,G+1} \). The one at which the value of the function is greater (if maximization is in progress) is adopted in the next generation (10).

\[
x_{i,G+1} = \begin{cases} u_{i,G+1}, & f(u_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G}, & f(u_{i,G+1}) > f(x_{i,G}) \end{cases}
\]

where \( f \) is the objective function, \( i = 1, 2, ..., N \).

The mutation, recombination and selection procedures are repeated until an interruption condition is reached.

6. Preliminary processing of the iris images

For reliable iris recognition, the global maximum of the integro-differential operator must always coincide with the true pupil border. In some cases, it is not so. An essential factor that can lead to the case when global maxima won’t be matched with pupil boundary is corneal reflections. Corneal reflections on the dark area of the pupil give a strong gradient that surpasses the gradient on the pupil edge. In the paper [16] authors examine in detail the reflections issues. To solve the problem, it proposes the method of detection and removing reflections based on interpolation of reflection area by the intensity of nearby pixels. The detection of the reflection area is performed by using an image thresholding procedure. The corneal reflection has a maximal intensity on the image. Using the top 5% brightest intensity in the iris image as thresholding value allows evaluating a binary map of reflections [16].

Another factor than can mangle the result of integro-differential operator applying is eyelashes. Eyelashes that overlay the iris area can make a huge intensity difference which can be wrongly taken as an iris boundary. In order to solve this problem, in the paper [17] authors propose using a horizontal rank filter to eliminate eyelashes.

7. Boundary conditions defining

Boundary condition defining is an important step. Strict restriction of boundary conditions allows decreasing computational load and probability of wrong localization. However, for non-cooperative recognition, there must be given the most general boundary conditions. To define the general boundary conditions, we used the following approach.
The natural restriction for $x_0, y_0$ in the operator (1) is the width and height of the image. The range for the center of the iris $x_0, y_0$ can be limited so that the entire region of the iris is in the picture. The minimum and maximum radius of the iris can be selected based on the minimum possible resolution for its texture. These values can be determined based on the size of the structural elements on the texture of the iris. The size of the pupil can be determined on the basis of anatomical considerations, the radius of the iris and the pupil are related. It is known [18], that the radius of the pupil can change from 0.1 to 0.8 times of the iris radius. Thus, we can define the relation between iris and pupil radiuses. Since the response of the operator to the border of the pupil is usually stronger, the search should begin with the pupil. Moreover, for the universality of the method, it was decided to apply the freest boundary conditions.

The maximum and minimum radii are selected taking into account the anatomical considerations described above. The center of the iris will also be significantly limited. The center of the iris can not exit the pupil area, so it is advisable to perform the search within these limits.

8. Initialization and local maxima problems

The convergence of optimization methods largely depends on the choice of initial values at the initialization stage. The method of setting the initial values is determined from the specifics of the task. Since non-cooperative recognition is analyzed, the iris may be in an arbitrary region of the input image. The Nelder-Mead and ItgDiff-ring methods do not have mechanisms for exiting local extremes; therefore, the initial values should be set as close as possible to the global maximum of the operator (1).

In this paper, the following method was chosen. A grid with evenly spaced nodes is superimposed on the parameter space. The distance between the nodes is determined based on the minimum possible radius of the iris (the definition of the minimum radius was discussed above). The blur rate $\sigma$ in the operator (1) is set to the maximum value for maximum blur. This reduces the probability of skipping borders if they fall into the space between grid nodes. The result of the approximate determination of the maximum on the grid is then used to initialize the methods. Notice that in the original work describing the ItgDiff-ring method, preliminary clustering of the image was carried out to limit the search area of the iris. However, this method is not suitable for the considering problem since the parameters of the clustering method are specific to each iris image dataset.

The differential evolution method has mechanisms for exiting local extremes due to the mutation stage. Nevertheless, to reduce the average time for determining the maximum, the initial population is set using the Latin Hypercube sampling, which allows increasing the coverage of the parameter space.

9. Experimental results

In this work, we have been determining pupil boundaries from public image datasets using the optimizing methods described above.

For testing purposes, we selected the next public iris image datasets: CASIA Iris Interval [19], CASIA Iris Lamp [19], UTIRIS V.1 [20], MMU Iris [21], Retica iris [22]. Images of selected datasets had different sizes, lighting conditions, and contrast. We performed pupil boundary localization by using different optimization methods on an integro-differential operator (1). Algorithms were applied with the same parameters for all datasets that allowed getting conditions close to non-cooperative recognition as much as possible. Localization errors were estimated. We grouped them by deviation type (deviation of center coordinates, deviation of radius) and by deviation degree (1-2 pixels, 3-5 pixels, and greater than 5 pixels).

In papers [11, 23] it was shown that small deviations have minimal impact on recognition accuracy. Centre coordinates determination errors have a bigger impact than errors in radius determination. Small error (1-2 pixels) can be leveled in the normalization stage. Results presented in table 1.
Table 1. Result of pupil localization accuracy analyze.

| Type                  | Degree, pixels | Differential evolution | Nelder-Mead | ItgDiff-ring |
|-----------------------|----------------|------------------------|-------------|--------------|
|                       |                | pp*        | no pp | pp          | no pp | pp    | no pp |
| Dataset: Retica iris, 50 images |                |            |       |             |       |       |       |
| From center           | <2             | 15         | 20    | 17          | 20    | 15    | 23    |
|                       | 3-5            | 0          | 2     | 1           | 1     | 2     | 0     |
|                       | >5             | 35         | 28    | 32          | 29    | 33    | 27    |
| Of radius             | <2             | 16         | 24    | 18          | 23    | 15    | 23    |
|                       | 3-5            | 7          | 7     | 5           | 5     | 8     | 5     |
|                       | >5             | 27         | 19    | 27          | 22    | 27    | 22    |
| Dataset: MMU Iris, 456 images |                |            |       |             |       |       |       |
| From center           | <2             | 211        | 213   | 227         | 287   | 228   | 284   |
|                       | 3-5            | 12         | 47    | 13          | 67    | 14    | 69    |
|                       | >5             | 227        | 190   | 210         | 96    | 208   | 97    |
| Of radius             | <2             | 309        | 217   | 305         | 184   | 311   | 169   |
|                       | 3-5            | 65         | 53    | 62          | 26    | 56    | 29    |
|                       | >5             | 76         | 180   | 83          | 240   | 83    | 252   |
| Dataset: CASIA Iris Lamp, 600 images |                |            |       |             |       |       |       |
| From center           | <2             | 475        | 420   | 432         | 422   | 434   | 420   |
|                       | 3-5            | 38         | 41    | 35          | 37    | 35    | 41    |
|                       | >5             | 87         | 139   | 133         | 141   | 131   | 139   |
| Of radius             | <2             | 474        | 431   | 421         | 429   | 423   | 427   |
|                       | 3-5            | 25         | 31    | 29          | 29    | 29    | 29    |
|                       | >5             | 101        | 138   | 150         | 142   | 148   | 144   |
| Dataset: UTIRIS V.1, 525 images |                |            |       |             |       |       |       |
| From center           | <2             | 359        | 510   | 112         | 305   | 126   | 307   |
|                       | 3-5            | 0          | 0     | 0           | 2     | 0     | 0     |
|                       | >5             | 166        | 15    | 413         | 218   | 399   | 218   |
| Of radius             | <2             | 359        | 510   | 113         | 303   | 125   | 304   |
|                       | 3-5            | 6          | 1     | 5           | 2     | 4     | 2     |
|                       | >5             | 160        | 14    | 407         | 220   | 396   | 219   |
| Dataset: CASIA Iris Interval, 2600 images |                |            |       |             |       |       |       |
| From center           | <2             | 2584       | 2582  | 2584        | 2583  | 2583  | 2581  |
|                       | 3-5            | 1          | 7     | 5           | 6     | 7     | 8     |
|                       | >5             | 15         | 11    | 11          | 11    | 10    | 11    |
| Of radius             | <2             | 2578       | 2567  | 2575        | 2567  | 2574  | 2566  |
|                       | 3-5            | 3          | 13    | 4           | 14    | 5     | 14    |
|                       | >5             | 19         | 20    | 21          | 19    | 21    | 20    |

* preliminary processing
Figure 2. Examples of images from the selected datasets. We carried out manual marking of iris contours.

We estimated the computational efficiency of each optimization method. The count of the objective function invocations was chosen as a criterion of computational efficiency. For Nelder-Mead and ItgDiff-ring methods, the number of called functions was taken into account only at the stage of execution of the optimization methods themselves. Initialization of the initial values for these methods was carried out by preliminary search for the maximum on a coarse grid of size 35x35x35.

Results presented in table 2.

Table 2. Results of the computational efficiency.

| Method         | Dataset            | Function invocation count |
|----------------|--------------------|---------------------------|
|                |                    | Min | Max | Average |
| Differential evolution | Retica iris | 3649 | 25474 | 14561 |
|                 | MMU Iris           | 1714 | 17644 | 9679  |
|                 | CASIA Iris Lamp    | 3109 | 25429 | 14269 |
|                 | UTRIS V.1          | 3874 | 30154 | 17014 |
|                 | CASIA Iris Interval| 2074 | 10264 | 6169  |
| Nelder-Mead*   | Retica iris        | 79  | 120  | 99    |
|                 | MMU Iris           | 71  | 136  | 103   |
|                 | CASIA Iris Lamp    | 89  | 130  | 109   |
|                 | UTRIS V.1          | 89  | 150  | 119   |
|                 | CASIA Iris Interval| 85  | 258  | 171   |
| ItgDiff-ring*  | Retica iris        | 4568 | 44331 | 24449 |
|                 | MMU Iris           | 3561 | 21309 | 12435 |
|                 | CASIA Iris Lamp    | 4640 | 46090 | 25365 |
|                 | UTRIS V.1          | 4721 | 48151 | 26436 |
|                 | CASIA Iris Interval| 4550 | 42770 | 23660 |

* The results do not take into account the number of function invocations at the stage of selecting the initial values.
Figure 3. Example of boundary localization for the image from the Retica iris dataset without preliminary processing (at the left side) and with preliminary processing (at the right side).

Figure 4. Examples of localization failure due to the strong gradient at the line of the outer iris boundary.
10. Results discussion
It was found that results in a vast degree depend on the testing dataset. All analyzing methods showed low accuracy on the Retica iris dataset. It can be explained by the fact that images of the dataset have bad quality.

Preliminary processing that including Gauss filter, corneal reflections removal [16] and eyelash removal [17] allowed improving localization result (figure 3).

It was revealed that the main reason for poor localization of the pupil in the Retica iris and MMU Iris bases is a stronger gradient at the outer border of the iris, in some cases even causing erroneous localization of the outer border instead of the pupil (figure 4). For the Iris MMU base, the use of preprocessing caused a significant deterioration in localization accuracy.

All the methods presented in the work showed a good result (99% accuracy) for the CASIA Iris Interval base. Images capturing for this dataset was carried out in laboratory conditions, with the same lighting and full cooperation of the subjects. For the CASIA Iris Lamp, UTIRIS V.1 databases, the best result was shown by the Differential Evolution method (accuracy 79% and 99%, respectively).

At the same time, the use of pre-processing significantly worsened the localization accuracy for the CASIA Iris Lamp base, but at the same time significantly improved for the UTIRIS V.1 base.

The best performance was shown by the differential evolution method. This can relate to the fact that in order to initialize the initial values in the Nelder-Mead and ItgDiff-ring methods, it was necessary to implement a rough preliminary determination of the maximum from the grid 35x35x35. This required an additional ~42K invocations of the objective function.

In general, the computational complexity of the Differential Evolution method also depends on the resolution of the test images. For images from the MMU Iris database with a resolution of 320x280 pixels, an average of ~9K calls were required, while for images from the UTIRIS V.1 database with a resolution of 1000x776 pixels, an average of ~17K invocations were required.

ItgDiff-ring showed the worst computational efficiency. The method performs optimization only according to the coordinates of the center of the pupil; for optimization along the radius, it is necessary to iterate through all possible values of the radius. In addition to an additional 42K invocations, it is necessary to do a radial search at each iteration of the method. Thus, the computational complexity of this method, in contrast to the Nelder-Mead method, depends on the resolution of the image.

11. Conclusion
A comparative analysis of optimization methods was carried out under conditions of non-cooperative recognition (images of a different quality from several datasets). For all datasets, a single algorithm was used without tuning parameters. When defining the boundary conditions, only the anatomical features of the iris were taken into account. Image resolution was different for each base. The sizes, position, angle of the iris, contrast, illumination, and other conditions were different and varied widely.

The use of pre-processing has shown ambiguous results. For the corneal reflection removal method, it was not possible to determine a parameter that is universal for all test images. For images from MMU Iris and Retica datasets, when a threshold set at 5% of the maximum brightness level, most of the skin in the lower part mistaken for a huge glare. Interpolation of such a large area according to the method [16] leads to false contours.

The results of computational experiments showed that the Nelder-Mead and Integro-differential ring methods perform the function of the completion after a rough estimate of the global maximum on the grid.

Without a rough definition of a global maximum, methods are highly likely to converge at a false local maximum. Although operations such as rough calculation of the maximum on the grid lend themselves well to parallelization, the Differential Evolution method compares favorably with their background. During the experiment, the Differential Evolution method showed the best result in terms of localization accuracy and computational efficiency. However, it has certain stochasticity.
Since the determination of the external boundary can perform without advanced optimization methods, this article examined only the internal borders of the iris. After determining the internal boundary, the determination of the external boundary can be carried out in very narrowly defined boundary conditions.

The center of the outer border can not go beyond the border of the pupil. Based on the radius of the pupil, we can more accurately set the range of radius of the outer border, based on the anatomical features. We can introduce additional linear constraints that account for the fact that the outer and inner borders must not touch each other.

References
[1] Minakova N and Petrov I 2018 Journal of Physics: Conference Series 944 012080
[2] Minacova N and Petrov I 2015 IOP Conference Series: Materials Science and Engineering 93 012056
[3] Minakova N N and Petrov I V. 2018 14th International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering, APEIE 2018 - Proceedings (Institute of Electrical and Electronics Engineers Inc.) pp 183–7
[4] Burge M J and Bowyer K W 2013 Handbook of Iris Recognition ed D M J B Virginia and P K Bowyer (Springer-Verlag London)
[5] Ashbourn J and Guide 2011 Guide to Biometrics for Large-Scale Systems Technological, Operational, and User-Related Factors (Springer-Verlag London)
[6] Petrovska-Delacrétaz D and Chollet G 2009 Guide to Biometric Reference Systems and Performance Evaluation (Springer-Verlag London)
[7] Jain A K, Flynn P and Ross A A 2008 Handbook of Biometrics (Springer-Verlag London)
[8] Daugman J 2003 Pattern Recognition 36 279–91
[9] Daugman J 1993 IEEE Trans. Pattern Anal. Mach. Intell. 15 1148–61
[10] Shamsi M, Saad P B, Ibrahim S B and Kenari A R 2009 SoCPaR 2009 - Soft Comput. Pattern Recognit. 393–398
[11] Proença H 2011 Eur. Signal Process. Conf. 2259–2263
[12] Daugman J 2009 Essent. Guid. to Image Process. 14 715–39
[13] Tan T, He Z and Sun Z 2010 Image Vis. Comput. 28 223–30
[14] Nelder J A and Mead R 1965 Comput. J. 7 308–313
[15] Storn R and Price K 1997 J. Glob. Optim. 11 341–59
[16] Aydi W, Masmoudi N and Kamoun L 2011 Int. J. Electron. Commun. Eng. 5 1068–1072
[17] He Z, Tan T, Sun Z and Qiu X 2008 Proc. of Int. Conf. Image Process. ICIP 265–8
[18] Thainimit S, Alexandre L A and De Almeida V M N 2013 Proc. of 13th Int. Symp. Commun. Inf. Technol. Commun. Inf. Technol. New Life Style Beyond Cloud, Isc. 501–6
[19] Cbsr.ia.ac.cn. (2019). Center for Biometrics and Security Research. [online] Available at: http://www.cbsr.ia.ac.cn/english/IrisDatabase.asp [Accessed 16 Nov. 2019].
[20] Hosseini M, Araabi B and Soltanian-Zadeh H 2010 Instrum. Meas. IEEE Trans. 59 792–804
[21] Multimedia University: MMU1 and MMU2 Iris Image Databases, 2008. http://pesona.mmu.edu.my/~ccteo. Accessed 17 Dec, 2015
[22] Neurotechnology.com. (2019). Download biometric algorithm demo software, SDK trials, product brochures. [online] Available at: http://www.neurotechnology.com/download.html [Accessed 12 Nov. 2019]
[23] Proença H and Alexandre L A 2010 Image Vis. Comput. 28 202–206