

Weighted Adaptive Hough and Ellipsopolar Transforms for Real-time Iris Segmentation

Andreas Uhl and Peter Wild*

Multimedia Signal Processing and Security Lab

Department of Computer Sciences, University of Salzburg, Austria

{uhl,pwild}@cosy.sbg.ac.at

Abstract

Efficient and robust segmentation of less intrusively or non-cooperatively captured iris images is still a challenging task in iris biometrics. This paper proposes a novel two-stage algorithm for the localization and mapping of iris texture in images of the human eye into Daugman's doubly dimensionless polar coordinates. Motivated by the growing demand for real-time capable solutions, coarse center detection and fine boundary localization usually combined in traditional approaches are decoupled. Therefore, search space at each stage is reduced without having to stick to simpler models. Another motivation of this work is independence of sensors. A comparison of reference software on different datasets highlights the problem of database-specific optimizations in existing solutions. This paper instead proposes the application of Gaussian weighting functions to incorporate model-specific prior knowledge. An adaptive Hough transform is applied at multiple resolutions to estimate the approximate position of the iris center. Subsequent polar transform detects the first elliptic limbic or pupillary boundary, and an ellipsopolar transform finds the second boundary based on the outcome of the first. This way, both iris images with clear limbic (typical for visible-wavelength) and with clear pupillary boundaries (typical for near infrared) can be processed in a uniform manner.

1. Introduction

Iris recognition identifies humans by their iris patterns. Irises are protected as internal flat organs and claimed to exhibit epigenetic randomness and stability over decades. They can be captured at-a-distance or on-the-move, and facilitate one-to-many identification with fast rotation-invariant comparators. Therefore, binary features are extracted from Daugman's doubly dimensionless representation [5]. But iris recognition is susceptible to poor

image quality. Efficient and robust segmentation of iris images is one of the most challenging problems in the field [12], especially for biometric systems without active participation of users. This led to a variety of public iris segmentation challenges in the last decade, e.g. ICE¹ and NICE². Still, computational demands as a very important factor in real-world applications and the tradeoff between accuracy and speed, have largely been neglected in evaluations so far. Furthermore, it has become common practice to enable participants to optimize their segmentation algorithm based on available training sets, which may lead to non-repeatable results when changing underlying datasets. This situation is even more critical, since the majority of segmentation algorithms is not publicly available for independent evaluations.

This work highlights real-time and database-independent iris segmentation. It proposes a generic iris segmentation technique under hard constraints: (a) segmentation in the order of deciseconds, and (b) no strong assumptions on image type and conditions. While most segmentation algorithms employ some sort of exhaustive searching or single error-prone strategies to detect pupillary and limbic boundaries [2, 11], this paper presents a two-stage iris segmentation framework. Compared to traditional techniques it has three major advantages: First, modules may easily be extended to incorporate more sophisticated techniques for individual tasks, yielding a trade-off between computation time and segmentation accuracy. Second, in presented configuration using a novel weighted version of an adaptive Hough transform [3] for approximate center detection and Polar and Ellipsopolar transforms for boundary detection, this incremental technique is faster and more scalable with respect to resolution. Third, failures can be detected and corrected in early stages leading to more robustness.

The paper is organized as follows: Section 2 reviews related work, Sec. 3 describes the proposed algorithm, experiments are outlined in Sec. 4 and summarized in Sec. 5.

*supported by FIT-IT Trust in IT-Systems, project no. 819382.

¹Iris Challenge Evaluation, <http://iris.nist.gov/ice/>

²Noisy Iris Challenge Evaluation Part I, <http://nice1.di.ubi.pt/>

Authors	Main stages	Algorithm steps	Database	Accuracy	Time
Labati <i>et al.</i> [8]	Center, Boundary (polar)	(1) agent-based method for center localization (2) multiple views boundary refinement	Casia.v3, Ubiris.2	90-97%	0.68 s
He <i>et al.</i> [7]	Center, Boundary (polar)	(1) reflection removal (2) Adaboost-cascade for approximate center detection (3) Pulling-and pushing model	Casia.v3, ICE	0.53-1.29% EER	12-22 ms
Chen <i>et al.</i> [4]	Sclera, Boundary (Hough)	(1) sclera location (2) fast Hough transform for limbic boundary, (3) eyelid detection (4) correction of limbic boundary by another Hough transform (5) eyelash detection and reliability	Ubiris.2	97-98%	0.83 s
Oroz <i>et al.</i> [9]	Center, Boundary (polar)	(1) color component selection (2) reflection extraction (3) limbic polar morphologic boundary extraction (4) center reestimation (5) pupillary boundary estimation	Ubiris.2	97%	3 s
Proença [11]	Sclera, Iris, Shape	(1) sclera detection (2) iris detection by machine learning (3) parameterization of iris shape	Ubiris.v2, ICE, FRGV, FERET	1.87-4.61% EER	0.70-0.78 s

Table 1. Experimental results of recently proposed multi-step iris segmentation algorithms.

2. Related Work

The main task of an iris segmentation algorithm is to map the iris texture of an $m \times n$ sized eye image I into Daugman's homogeneous rubbersheet model [5]. More precisely, a rendering map R is looked for, assigning - regardless of pupillary dilation and iris size - each pair (θ, r) of angle θ and pupil-to-limbic radial distance r the corresponding originating location $R(\theta, r)$ within I :

$$R : [0, 2\pi) \times [0, 1] \rightarrow [0, m] \times [0, n] \quad (1)$$

In Daugman's representation, R is established as a linear combination from circular pupillary and limbic polar boundary curves $P, L : [0, 2\pi) \rightarrow [0, m] \times [0, n]$, typically modeled as circles with not necessarily coinciding centers:

$$R(\theta, r) := (1 - r) \cdot P(\theta) + r \cdot L(\theta) \quad (2)$$

Note, that in practice, P and L may be modeled as arbitrary polar sampled curves. In case the iris texture is occluded by eyelids, the model assumes that P and L mark the true possibly occluded pupillary and limbic iris boundaries, respectively. Within the resulting iris texture, areas occluded by eyelids, eyelashes or reflections should be masked out using binary noise masks N of the same size:

$$N : [0, 2\pi) \times [0, 1] \rightarrow \{0, 1\} \quad (3)$$

Finally, R, P, L , and N are typically discretized in implementations.

State-of-the-art iris segmentation systems (see [2, 11] for an overview) may be classified into two different types. The first kind of algorithms are *model-based* techniques, fitting a parameterized shape by some exhaustive searching method to derive P and L . Compared to the traditional segmentation algorithms by Daugman [5] using an integrodifferential operator, or Wildes [16] using binary edge maps and Hough transform to approximate P and L as circles, the following

improvements have been proposed in the literature so far: (a) better models, such as ellipses or view-angle transformations to account for off-gaze [15], (b) better occlusion detection [17], (c) after-fit refinements permitting more complex shapes including active contours, active shape models [1] and Fourier series expansions [6], modeling and smoothing the series of k polar boundary gradient values $(r_\theta)_{\theta=0}^{k-1}$ with l Fourier coefficients $(f_t)_{t=0}^{l-1}$ to a new boundary $(r'_\theta)_{\theta=0}^{k-1}$:

$$f_t := \sum_{\theta=0}^{k-1} r_\theta e^{-2\pi i t \theta / k}, \quad r'_\theta := \frac{1}{N} \sum_{t=0}^{l-1} f_t e^{2\pi i t \theta / k} \quad (4)$$

A disadvantage of most of these single-strategy techniques is high computational demand. The second kind of algorithms are *multi-stage* methods, where not a single but multiple different models are applied to solve subproblems of the segmentation task, such as separate stages for sclera detection, center-detection, pupillary and limbic boundary detection. Each stage employs a simpler model, thus reducing the amount of parameters needed. The result serves for some transformation input to simplify the next stage. A common employed transform (e.g. in [8, 9, 7]) once an iris center (x, y) has been found is the polar transform T and its inverse T^{-1} , in order to simplify boundary detection:

$$T : [0, 2\pi) \times \mathbb{R}_0^+ \rightarrow \mathbb{R}^2, \quad T \begin{pmatrix} \theta \\ r \end{pmatrix} := \begin{pmatrix} x + r \cos \theta \\ y + r \sin \theta \end{pmatrix} \quad (5)$$

The approaches of Labati *et al.* [8] and Oroz *et al.* [9] in Table 1, listing examples of recently proposed multi-step algorithms, employ this transform to reduce parameters of their Hough-based model and speed-up the computation. He *et al.* [7] use polar transform as an intermediate step to get gradient values for their pulling-and-pushing model. Chen *et al.* [4] use again speed-ups of original Hough transform reducing parameter space with a pre-location of the pupillary center. Some newer approaches operating on visible-range iris data conduct separate Sclera detection stages, e.g. [11].

3. Technical Details

In order to obtain more robustness to various noisy artifacts of less intrusively or non-cooperatively captured iris images, the proposed algorithm illustrated in Fig. 1 is a *two-stage* segmentation technique.

The first task consists of finding a center point C within the input image I completely inside the limbic and pupillary boundary. The key operation to derive C is a novel weighted adaptive Hough transform, which determines the center of multiple approximately concentric rings at iteratively refined resolution. It accumulates lines in direction of the gradient at boundary edge candidate points, giving a higher weight to locations in the center of the accumulator. While C is not unique, ideally it will be close to the centers of circles approximating L and P . The idea is to exploit orientation and magnitude of both pupillary and limbic edges to find C , instead of sequential boundary extraction. As noted in [11], until now, almost all state-of-the-art segmentation systems employ a fixed order, in which pupillary and limbic boundaries are fitted, usually influenced by the type of training data. The reason for this behavior can be explained by the different nature of near infrared (NIR) images, with high contrast pupillary boundaries, and visible-range (VR) images, with clearly visible limbic edges. The advantage of the presented approach is a uniform processing of different image types emphasizing database-independence.

The second task consists of extracting P and L from a polar representation using C as origin. Within this representation all boundary edges have approximately the same orientation, which reduces the cost of edge detection. Like in the stage before, no strict order for the detection of limbic and pupillary boundary is given. Instead, an initial boundary B is detected from the polar transformed input by (a) determining the maximum-energy horizontal line (modeling a circle in cartesian coordinates), (b) maximizing the vertical polar gradient for each column (discrete angle), (c) smoothing the resulting curve, (d) remapping candidates to cartesian coordinates, and (e) fitting the edge points with an oriented ellipse. Since the found boundary B is either the limbic or pupillary boundary, the algorithm continues to find the second boundary based on the two hypotheses $H_0 : B = P$ and $H_1 : B = L$ using an Ellipsopolar transform. This transform maps ellipses concentric with elliptic boundary B to horizontal lines, which is helpful since P and L should be approximately concentric. Again, steps (a)-(e) are executed to derive an inner candidate P' and outer candidate L' . Based on the outcome of the gradient energy e of these contours one hypothesis is rejected. In case $e(P') > e(L')$ hypothesis H_0 is rejected and $P = P'$, otherwise $L = L'$. Finally, the rubbersheet transform is applied.

An intended goal of the proposed framework is the uniform processing of pupillary and limbic boundaries for VW

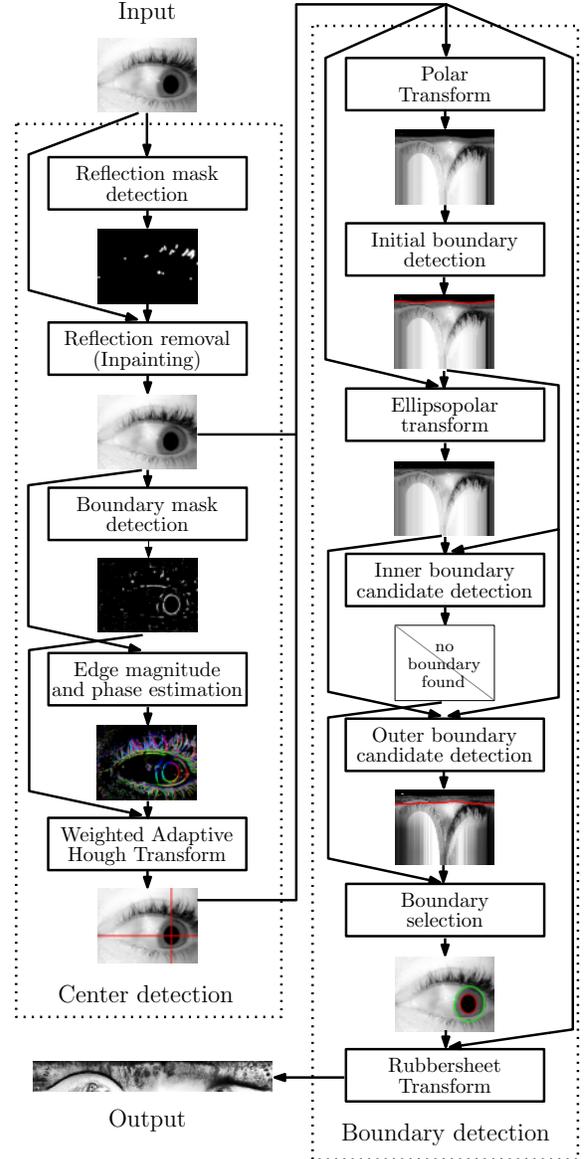


Figure 1. Architecture of proposed two-stage iris segmentation.

(visible wavelength) and NIR (near-infrared) images, the latter exhibiting a sharper pupillary boundary, especially for dark irides. Also noise masks, using e.g. methods in [7], are not yet computed. While they may certainly improve recognition rates when employing Daugman’s hamming distance (HD) comparator, (a) they are not critical for the rubbersheet mapping, (b) fast and well-working methods already exist, and (c) there are even new comparators available as alternatives to the storage of noise masks [13].

Segmentation software uses the OpenCV³ image library and is written in C++.

³Intel, Willow Garage: Open Source Computer Vision Library, <http://opencv.willowgarage.com>

3.1. Reflection Mask Detection and Removal

Reflections inside the pupil may significantly affect iterative center search. Therefore, a reflection mask $M : [0, m] \times [0, n] \rightarrow \{0, 1\}$ is computed in three steps: (a) Adaptive thresholding selects all pixels (x, y) with intensities exceeding the local mean $A(x, y)$ in the 23×23 neighborhood plus a constant $c = 60$:

$$M_A \begin{pmatrix} x \\ y \end{pmatrix} := \begin{cases} 1 & \text{if } I(x, y) > A(x, y) + c \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

(b) Region size filtering sets all connected one-components in M_A , which are less than 10 and greater than 1000 pixels, to zero; and (c) Morphological dilation is applied using a circular 11×11 structuring element.

In order to remove reflections, the original image I is inpainted using M , i.e. all selected regions are reconstructed from their boundary using the *Navier-Stokes* method natively provided by the OpenCV library, resulting in inpainted image I' .

3.2. Edge Detection and Boundary Mask

Edge phase $P_E : [0, m] \times [0, n] \rightarrow [0, 2\pi)$ and magnitude $M_E : [0, m] \times [0, n] \rightarrow \mathbb{R}_0^+$ are estimated from the inpainted image I' using horizontal and vertical 7×7 Sobel kernels. A boundary edge mask $E : [0, m] \times [0, n] \rightarrow \{0, 1\}$ detects initial candidate points for center estimation in the next step: from the top 20 percent of edge points with respect to edge magnitude all candidates within cells of a 30×30 grid with a dominant mean orientation (i.e. the magnitude of the mean orientation exceeds $m = 0.5$) are selected. The idea of this filtering is to remove candidate points in eyelashes with an almost equal amount of high edges with opposite directions.

3.3. Weighted Adaptive Hough Transform

In order to speed up traditional Hough transform, we have adopted an iterative approach to finding the center point C proposed by Cauchie *et al.* [3]. Instead of estimating center and radius of the most dominant circle, this method tries to find the center of the most distinctive concentric circles in the image using not only gradient magnitude but also gradient orientation. Compared to the basic version of this algorithm in [3], a number of modifications have been applied: (a) Bresenham's fast line algorithm is used to fill accumulator cells, (b) voting accounts for magnitude information, (c) a new weighting step employs a Gaussian function to account for prior knowledge, and (d) the region of interest refinement step has been simplified to select a highest energy cell only.

Starting with $i = 0$ our modified version of this algorithm consists of three iteratively executed steps at each stage i , see also Fig. 2: (a) Initialization: For a rectangular region of interest R_i of size $m_i \times n_i$ (initially $R_0 = I'$), an

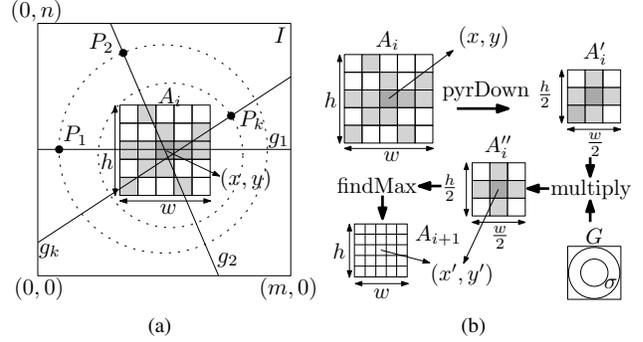


Figure 2. Weighted Adaptive HT: Voting (a) and Weighting (b).

accumulator A_i consisting of a fixed $w \times h$ grid of cells covering R_i is initialized with zero. From the set of candidate edge points E_{i-1} (initially $E_{-1} = \{(x, y) : E(x, y) = 1\}$), all points $P_j \in E_{i-1}$, whose gradient lines g_j do not intersect with R_i , are rejected in E_i . (b) Voting: All cells in A_i crossed by a gradient line g_j of a candidate point in E_i are incremented using the absolute gradient value. (c) Weighting: the accumulator A_i is downsampled, multiplied with a centered Gaussian kernel G , the cell of maximum value is found and R_{i+1} is centered in the cell's center (x', y') . As long as R_{i+1} (with $m_{i+1} = \frac{m_i}{2}, n_{i+1} = \frac{n_i}{2}$) is larger than a predefined threshold, the algorithm repeats with step (a), following a coarse to fine strategy, otherwise $C = (x', y')$.

By adding a weighting step, modeled by Gaussian G , it is possible to incorporate the heuristic, that at each stage it is likely to find the maximum cell near the center of the accumulator. Especially for early stages this is a valuable prior knowledge to suppress eyelid gradients. The implementation uses a 101×101 accumulator grid, 0.5 pixel as the accumulator precision threshold and the applied Gaussian sigma has been set to one third of the downsampled accumulator size.

3.4. Polar Transform Initial Boundary Detection

In the polar transform stage, found center C is used to polar unwrap the iris image I' using a discretized version of Def. 5, restricted to a maximum mapped radius, resulting in a $k \times g$ polar image I_p . The angular dimension typically depends on the desired output resolution, and is set to $k = 512$ per default, the radial dimension is set to $g = 512 * \frac{n}{m}$, the maximum mapped radius corresponds to the maximum distance from C to each image corner.

Initial Boundary detection operates on the polar transformed I_p to derive a boundary contour. Therefore, I_p is convolved with an oriented 21×21 Gabor kernel:

$$g(x, y) := \frac{\gamma}{2\pi\sigma^2} \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right)$$

where $x' = x \cos(\theta) + y \sin(\theta)$, $y' = y \cos(\theta) - x \sin(\theta)$ (7)

All candidate edges have the same orientation, therefore a good choice is $\lambda = 8\pi, \psi = \frac{\pi}{2}, \sigma = 6, \gamma = 0.5, \theta = -\frac{\pi}{2}$. The resulting gradient image is G_p .

Initial boundary B is determined as follows: (a) Starting at an offset $o = 12$ (to avoid image border effects), for each radial value $r \in [o, g - 1] \cap \mathbb{N}$ (corresponding to a line in G_p) the sum of gradient values at each discrete polar angle $\theta \in [0, k - 1] \cap \mathbb{N}$ is computed and the polar contour series $(b_\theta)_{\theta=0}^{k-1}$ is initialized by the radius r_{max} with maximum gradient sum, i.e:

$$\forall \theta : b_\theta = r_{max}. \quad (8)$$

(b) Using gradient fitting within a local radial window W (we use $W = [-15, 15] \cap \mathbb{N}$), the contour gradient is maximized yielding a refined contour $(b'_\theta)_{\theta=0}^{k-1}$:

$$\forall \theta : b'_\theta = b_\theta + \max_{i \in W} \arg (G_p(\theta, b_\theta + i)) \quad (9)$$

(c) The resulting contour is smoothed using 1D Fourier series expansions (see eq. 4) keeping the DC and 1 coefficient, followed by another gradient fitting (b) with reduced window size 5 and again a Fourier fitting (c) keeping the DC plus 3 coefficients, resulting in $(b''_\theta)_{\theta=0}^{k-1}$. Finally step (d) maps $(b''_\theta)_{\theta=0}^{k-1}$ back to cartesian coordinates (using the inverse polar transform) and uses the best-fitting ellipse in a least-squares sense, computed using the *Fitzgibbon* algorithm natively provided by OpenCV, as boundary curve B .

3.5. Ellipsopolar Transform Inner and Outer Boundary Detection

Typically, after the first iteration of polar boundary detection, found curve B can be used to derive a more accurate center than C . An even better idea is to take the shape of B into consideration: This work proposes the application of an ellipsopolar transform, which is essentially a polar transform after translation, rotation and stretching, mapping concentric ellipses to axis-parallel lines, see Fig. 3. Let B be the polar sampled representation of a general oriented ellipse with center (x, y) , half axes a, b and angle of inclination α , then the general ellipsopolar transform is defined as:

$$T_E : [0, 2\pi) \times \mathbb{R}_0^+ \rightarrow \mathbb{R}^2$$

$$T_E \begin{pmatrix} \theta \\ r \end{pmatrix} := \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} * \begin{pmatrix} ra \cos(\theta - \alpha) \\ rb \sin(\theta - \alpha) \end{pmatrix} \quad (10)$$

Note, that again a discretized version with maximum mapping radius of this map is applied, analogous to the polar transform introduced before. By mapping inpainted input image I' into ellipsopolar coordinates, we can again perform boundary fitting (i.e. seeking an inner boundary by

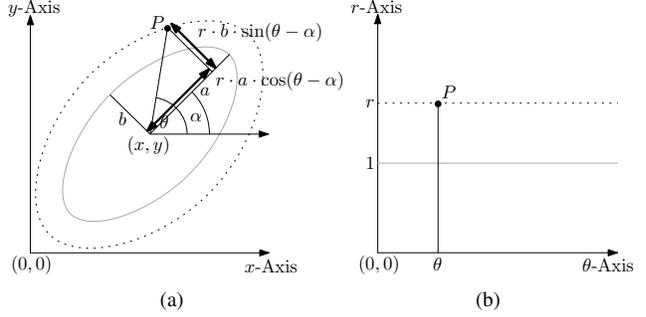


Figure 3. Ellipsopolar transform: Cartesian domain (a) and Ellipsopolar domain (b).

restricting our search to radial values $r < 1$ and an outer boundary with $r > 1$) using the same steps (a)-(d) introduced in the initial boundary detection phase, except, that now the initialization phase favors an elliptic boundary concentric to B , which is a very good first approximation. In order to further incorporate model-specific prior knowledge in step (a) a Gaussian weighting function is applied to the sum. The actual weight depends on the radial distance r of the current radial distance to the reference contour B (we use Gaussian parameters $\mu_i = 0.66, \sigma_i = 0.44$ for the inner search and $\mu_o = 2.5, \sigma_o = 1$ for the outer boundary search). The result of inner and outer boundary detection are elliptic boundary candidates P' and L' .

3.6. Boundary Selection and Rubbersheet Transform

Having computed P' and L' based on the hypotheses $H_0 : B = P$ and $H_1 : B = L$, respectively, the task of boundary/hypothesis selection involves the computation of the following gradient energy function e for an ellipsopolar sampled boundary curve X with respect to ellipse B (defining the ellipsopolar mapping):

$$e(X) := \sum_{\theta=0}^{k-1} M_E(X(\theta)) * W(\mu, \sigma, r(X, \theta, B)) \quad (11)$$

I.e., the gradient values in M_E at discrete sample points of X are summed and weighted using a Gaussian weighting function W using the ellipsopolar radial value $r(X, \theta, B)$ of X with respect to transform ellipse B at angle θ and fixed Gaussian parameters μ, σ . Depending on whether the inner or outer boundaries are evaluated, μ_i, σ_i or μ_o, σ_o from the inner and outer boundary detection stages introduced before are used. If $e(P') > e(L')$ we set $P = P'$, otherwise $L = L'$.

Finally, the resulting boundary curves P and L are subjected to Daugman's rubbersheet model (see Def. 2), and the resulting iris texture is enhanced using contrast-limited adaptive histogram equalization [14].

4. Experiments

Iris segmentation software is driven by three different quality factors: *Accuracy* is the desired property of robust detection of pupillary and limbic boundaries with few segmentation errors. Especially robustness to moderate noise factors, such as defocus or motion blur, varying illumination, off-gaze, high variance in iris size and non-linear distortions by the application of imperfect iris models, is desirable. This quality factor is evaluated by assessing the impact of segmentation on verification recognition accuracy, i.e. ROC curves, given in Figs. 4-6. *Speed*, i.e. adherence to near real-time constraints (at least less than 1 second processing time per image), is a quality factor often neglected in comparison studies, but this factor is most critical in applications. Average segmentation time is evaluated for each of the employed databases, see Fig. 7. Finally, *usability* is the quality factor most difficult to quantify, but possibly the most important of all three factors. Usability refers, to which extent a segmentation algorithm can be used with effectiveness, efficiency, and satisfaction. This includes reproducibility of segmentation performance under different sensors and environmental conditions (i.e. the algorithm avoids database-specific *tuning*), the ability to exchange and improve single submodules of the algorithm (i.e. ideally availability as open source software), and easy parameterless configuration and intuitive use of the segmentation tool. Usability is evaluated by setting results on different databases into context and analyzing segmentation errors and results in Figs. 9-10.

4.1. Experimental Setup

Experiments are carried out using 3 different datasets from open biometric databases: (a) *Casia-I* consists of the left-eye subset (1332 images) of *CASIA-V4*⁴ set *Interval*, good quality NIR illuminated indoor images with 320×280 pixel resolution, (b) *Casia-L* is composed of the first 10 left-eye samples of the first 100 users (1000 images) in *CASIA-V4* set *Lamp*, a more challenging 640×480 pixel resolution indoor NIR images dataset, (c) *ND* is a subset of 42 classes with 10 samples per class (420 images) in *ND-IRIS-0405*⁵, non-ideal 640×480 pixel resolution indoor NIR images.

For reference performance, the following algorithms have been used: (a) *OSIRIS*⁶ is an open source reference system for iris recognition and uses a two-stage approach for segmentation: First, the pupil region in the image is roughly searched using a binarization and exploiting the circularity of the pupil. Limbic and pupillary contours are

⁴The Center of Biometrics and Security Research, CASIA Iris Image Database, <http://biometrics.idealtest.com>

⁵Computer Vision Research Lab, Univ. of Notre Dame Iris Dataset 0405, <http://www.nd.edu>

⁶Krichen *et al.*: A biometric reference system for iris. OSIRIS version 2.01, http://svnext.it-sudparis.eu/svnview2-eph/ref_syst/Iris_Osiris/

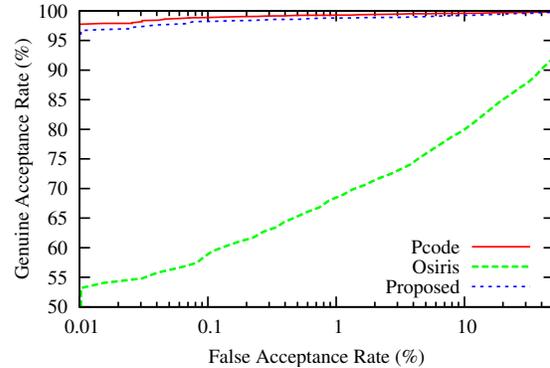


Figure 4. ROC on Casia-I using different segmentation.

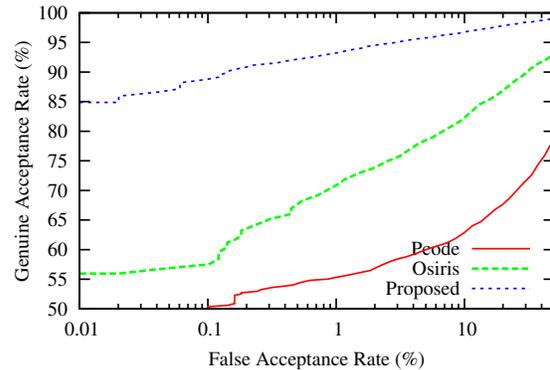


Figure 5. ROC on Casia-L using different segmentation.

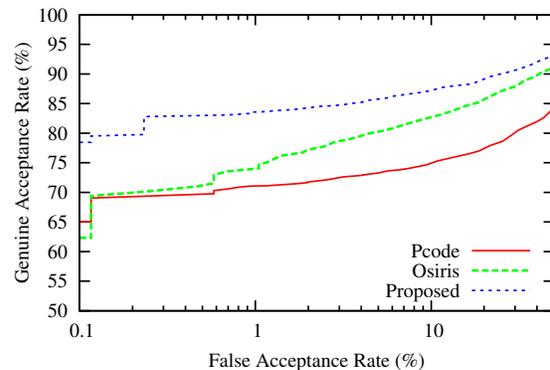


Figure 6. ROC on ND using different segmentation.

determined using Hough transform and active contours for refinement. (b) *Pcode*⁷ is a custom implementation of an Hough transform approach using (database-specific) contrast adjustment to enhance pupillary and limbic boundaries, Canny edge detection to detect boundary curves and enhancement techniques to remove unlikely edges.

For all algorithms, evaluations used the 512×64 pixel iris texture in doubly-dimensionless coordinates only, neither noise masks nor other enhancement methods on the

⁷Pschernig: Cancelable biometrics for iris detection with parameterized wavelets and wavelet packets, Master's thesis, Univ. Salzburg, 2009

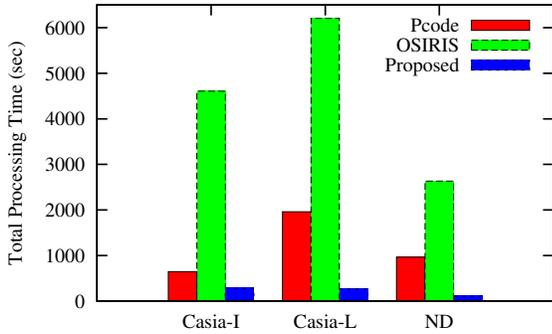


Figure 7. Total segmentation time for different datasets.

Algorithm	Equal Error Rate (EER)		
	Casia-I	Casia-L	ND
Pcode	0.74%	28.77%	22.01%
OSIRIS	16.40%	14.89%	15.45%
Proposed	1.20%	4.36%	12.90%

Table 2. Summarized segmentation accuracy.

Algorithm	Segmentation Time (ST)		
	Casia-I	Casia-L	ND
Pcode	0.49 s	1.96 s	2.29 s
OSIRIS	3.46 s	6.21 s	6.27 s
Proposed	0.21 s	0.26 s	0.25 s

Table 3. Summarized segmentation time per image.

original textures were considered. All obtained textures were enhanced using Contrast-limited adaptive histogram equalization [14]. To assess the impact on recognition accuracy, the feature extraction technique by Ma *et al.* [10] is applied, using a custom implementation optimized for Casia-I. This algorithm performs a dyadic wavelet transform and selects local minima and maxima above an adequate threshold in two subbands, yielding a 10240 bit code. Hamming distance is applied as dissimilarity measure and alignment is achieved applying up to 7-bit circular shifts in each direction. Note, that higher error rates compared to evaluations where noise masks are considered or the feature extraction stage is optimized for the employed dataset are expected and do not necessarily reflect segmentation errors (especially for the more challenging datasets).

4.2. Experimental Results

For the *Casia-I* dataset, experiments resulted in the best Equal Error Rate (EER) of 0.74% at on average 0.49 seconds segmentation time per image (ST) for the Pcode implementation. This result is not too surprising, as this method is explicitly tuned to deliver good results for this database, boundaries can well be represented with circles and usually strong models provide good results in case of weak data

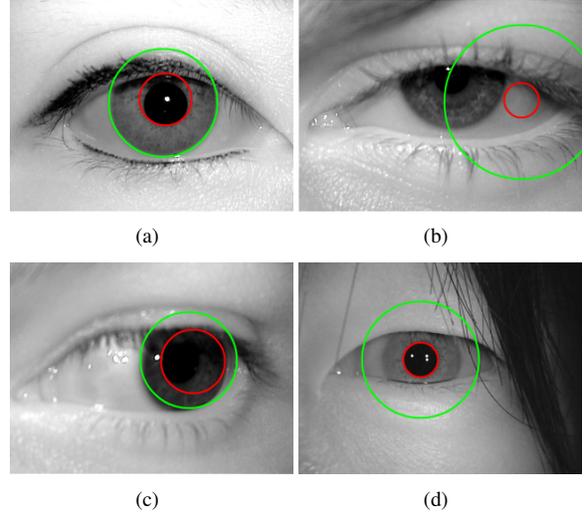


Figure 8. Typical Pcode segmentation failure types: eyelid matches circular model (a), squeezed eyes (b), inaccurate model for off-gaze (c) and oversegmentation due to hair (d).

(such as soft limbic boundaries). However, a strong model becomes useless if the data can not well be represented, see Fig 8. The proposed method follows closely with only slightly higher 1.20% EER at 0.21 ST. Finally, OSIRIS with 16.4% EER at 3.46 ST delivered worst accurate results and also the highest ST. When looking at the type of segmentation errors made by the algorithms, it is interesting to see, that OSIRIS frequently makes over-segmentation errors (due to less pronounced limbic boundaries) while our proposed method reveals a few over-segmentation errors due to sharp collarettes.

In *Casia-L* the proposed method provides the best results (4.36% EER at 0.26 ST) clearly outperforming OSIRIS (14.89% EER at 6.21 ST) and Pcode (28.77% EER at 1.96 ST). It is interesting to see, that a quite well-working segmentation system (Pcode) may fail completely, if assumptions do not hold. Pcode's segmentation errors include many complete failures and some under-segmentation errors due to eyelids. OSIRIS exhibits many over- and under-segmentation errors. For this dataset segmentation of the proposed algorithm works quite well, except in case of very strong eyelids, which affect ellipse fitting.

Even though the OSIRIS algorithm is tuned to deliver good results for ICE-2005 our implementation delivers the best results for *ND* (the parent database is a superset of ICE-2005) with 12.90% EER at 0.25 ST, followed by OSIRIS (15.45% EER at 6.27 ST) and finally Pcode (22.01% EER at 2.29 ST). The type of segmentation errors made by all three algorithms are comparable to Casia-L, with slightly more stable results provided by OSIRIS. Still a severe problem in OSIRIS is the fact that the computation of snakes is not reflected in the rubbersheet mapping, yielding mapping distortions, see Fig. 9.

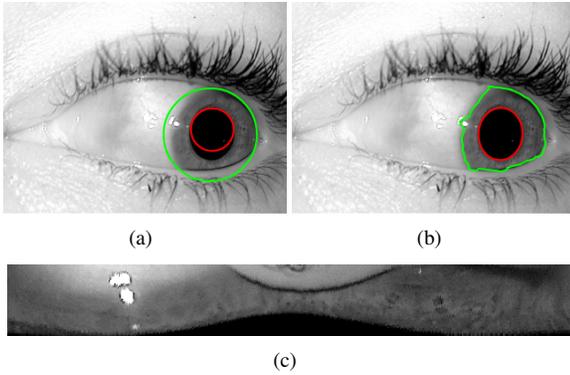


Figure 9. Osiris ND #0428d197 segmentation failure : rubbersheet mapping (a) causes distortions (c), despite of accurate snake (b).

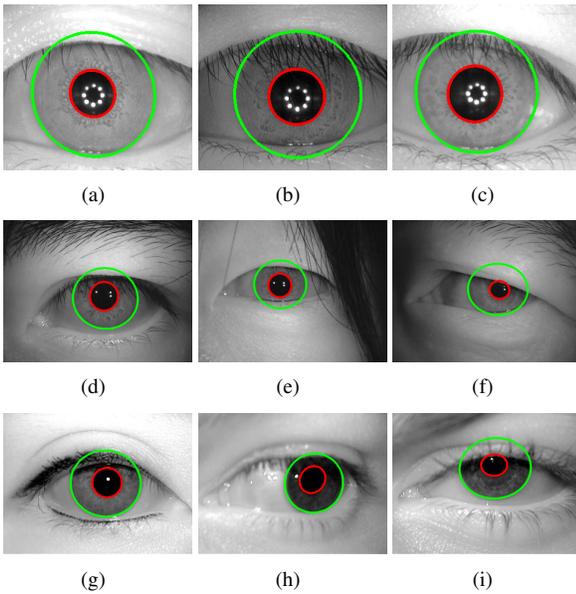


Figure 10. Good segmentation results of proposed algorithm for challenging examples: CASIA-I (a-c), Casia-L (d-f), ND (g-i).

In summary, from the tested algorithms the proposed technique clearly provided fastest and most stable results across databases, see Fig. 10. Observed recognition rates and processing time are given in Tables 2 and 3.

5. Conclusion

Typically, iris segmentation algorithms are optimized with respect to a given database, neglecting the need for high usability of segmentation algorithms. In this paper a two-stage iris segmentation framework is proposed, which aims to account for all three quality factors of segmentation: accuracy, speed and usability. The proposed algorithm is shown to require less processing time and to exhibit more robustness than tested open iris segmentation software. Experiments confirm, that existing segmentation software in the field is highly affected by the type of iris data, i.e. the used iris database. The proposed method tries to avoid

this over-fitting with respect to input data and does not use database-specific segmentation parameters or assumptions.

References

- [1] A. Abhyankar and S. Schuckers. Active shape models for effective iris segmentation. In *Proc. of SPIE*, 2006.
- [2] K. W. Bowyer, K. Hollingsworth, and P. J. Flynn. Image understanding for iris biometrics: A survey. *Comp. Vision and Image Underst.*, 110(2):281–307, 2008.
- [3] J. Cauchie, V. Fiolet, and D. Villers. Optimization of an hough transform algorithm for the search of a center. *Pattern Recognition*, 41(2):567–574, 2008.
- [4] Y. Chen, M. Adjouadi, C. Han, J. Wang, A. Barreto, N. Rische, and J. Andrian. A highly accurate and computationally efficient approach for unconstrained iris segmentation. *Image Vision Computing*, 28:261–269, 2010.
- [5] J. Daugman. How iris recognition works. *IEEE Trans. Circ. Syst. Video Techn.*, 14(1):21–30, 2004.
- [6] J. Daugman. New methods in iris recognition. *IEEE Trans. Syst., Man, and Cybern. Part B*, 37(5):1167–1175, 2007.
- [7] Z. He, T. Tan, Z. Sun, and X. Qiu. Toward accurate and fast iris segmentation for iris biometrics. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(9):1670–1684, sept. 2009.
- [8] R. D. Labati, V. Piuri, and F. Scotti. Agent-based image iris segmentation and multiplevews boundary refining. In *Proc. 3rd Int'l Conf. on Biometrics: Theory, App. and Syst.*, pages 204–210, Piscataway, NJ, USA, 2009. IEEE Press.
- [9] M. A. Luengo-Oroz, E. Faure, and J. Angulo. Robust iris segmentation on uncalibrated noisy images using mathematical morphology. *Image Vision Computing*, 28:278–284, 2010.
- [10] L. Ma, T. Tan, Y. Wang, and D. Zhang. Efficient iris recognition by characterizing key local variations. *IEEE Trans. Image Proc.*, 13(6):739–750, 2004.
- [11] H. Proença. Iris recognition: On the segmentation of degraded images acquired in the visible wavelength. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1502–1516, 2010.
- [12] H. Proença and L. A. Alexandre. Iris recognition: Analysis of the error rates regarding the accuracy of the segmentation stage. *Image and Vision Computing*, 28(1):202–206, 2010.
- [13] C. Rathgeb, A. Uhl, and P. Wild. Incremental iris recognition: A single-algorithm serial fusion strategy to optimize time complexity. In *Proc. 4th Intl. Conf. on Biometrics: Theory, App. and Syst.*, pages 1–6, 2010.
- [14] A. M. Reza. Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *J. VLSI Signal Process. Syst.*, 38(1):35–44, 2004.
- [15] S. Schuckers, N. Schmid, A. Abhyankar, V. Dorairaj, C. Boyce, and L. Hornak. On techniques for angle compensation in nonideal iris recognition. *IEEE Trans. on Systems, Man, and Cybernetics Part B*, 37(5):1176–1190, 2007.
- [16] R. P. Wildes. Iris recognition: an emerging biometric technology. In *Proc. of the IEEE*, volume 85, pages 1348–1363, 1997.
- [17] J. Zuo, N. Kalka, and N. Schmid. A robust iris segmentation procedure for unconstrained subject presentation. In *Proc. Biometric Consortium Conf.*, pages 1–6, 2006.