CORE

Metadata, citation and similar papers at core.ac.uk

Provided by Kent Academic Repository

A study on iris textural correlation using steering kernels

Yang Hu, Konstantinos Sirlantzis, Gareth Howells
School of Engineering and Digital Arts, University of Kent
University of Kent, Canterbury, UK, CT2 7NT
{yh94, K.Sirlantzis, W.G.J.Howells}@kent.ac.uk

Abstract

Research on iris recognition have observed that iris texture has inherent radial correlation. However, currently, there lacks a deeper insight into iris textural correlation. Few research focus on a quantitative and comprehensive analysis on this correlation. In this paper, we perform a quantitative analysis on iris textural correlation. We employ steering kernels to model the textural correlation in images. We conduct experiments on three benchmark datasets covering iris captures with varying quality. We find that the local textural correlation varies due to local characteristics in iris images, while the general trend of textural correlation goes along the radial direction. Moreover, we demonstrate that the information on iris textural correlation can be utilized to improve iris recognition. We employ this information to produce iris codes. We show that the iris code with the information on textural correlation achieves an improved performance compared to traditional iris codes.

1. Introduction

Biometric systems aim to automatically authenticate human based on their behavioural and physiological characteristics, instead of conventional passwords or ID cards. A large number of biometric traits have been employed by biometric systems, for example, fingerprint, iris, face, palm print, finger vein, gait, voice, retina, etc. Among these biometric traits, iris has shown a high reliability due to the stability and distinctiveness of iris textures. State-of-the-art iris recognition algorithms have achieved very promising performance [3, 4, 21, 2, 11, 8]. Current large scale deployments of iris recognition systems in UAE and India are considered successful. Also, significant research on less constrained iris recognition have been conducted and the results are encouraging [9, 10, 15, 16, 18, 19, 13].

Intuitively, iris textures have inherent correlation. For example, a furrow or ciliary pattern tends to propagate in the radial direction, as observed in [3, 9]. Also, some experimental evidences exist for such correlation. In [5], Daugman compares the capacity between real iris codes and synthetic 'white noise' iris codes which have fully random and uncorrelated bits. The experimental results show that the capacity of real iris codes is lower than 'white noise' iris codes. Such capacity difference reflects the anatomical correlation within a natural iris, besides the correlation induced by Gabor filters [5]. In [11], Ma *et al*. conclude that the information along the angular direction of an iris image is highly discriminating. This conclusion means that the iris texture along the angular direction has higher randomness (*i.e*. with less correlation). In [9], Liu *et al*. propose a Markov network model with radial connections for code-level information fusion. The improved performance of this model demonstrates the radial correlation of iris textures.

However, the evidences of iris textural correlation are less direct in the above works. In these works, this correlation is recognized based on the performance of iris matching results, rather than directly modelling and analysing the textural correlation in iris images. Therefore, although the experimental results have successfully revealed the existence of iris textural correlation, the following questions remain open: (1) What is the characteristic of textural correlation in human iris? For example, does all the textural correlation follow radial direction? If not, is the radial correlation dominant? What is the direction that the iris texture is most likely to be uncorrelated (*i.e*. the most random and distinctive)? What is the distribution of the direction of iris textural correlation? (2) Is it possible to utilize the information on iris textural correlation to improve iris recognition?

To investigate the above questions, a model for iris textural correlation and a quantitative analysis on it are desirable.

In this paper, we employ steering kernels (SKs) [17] to model and analyse the iris textural correlation. The SK is able to reflect the correlation between a pixel and its neighbouring regions. Therefore, by calculating and analysing the SKs centered at each pixel of iris images, we are able to quantitatively analyse the iris textural correlation. Fig. 1 shows some examples of SKs. Fig. 1(a) shows an unwrapped iris image with four marked example pixels. Fig. 1(b) presents the image patches centered at each

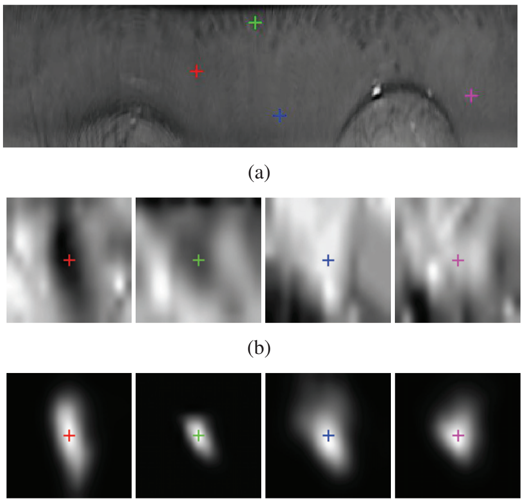


Figure 1: Examples of steering kernels modelling iris textural correlations: (a) an unwrapped iris image with example pixel marked by four colours; (b) the images patch centered at each example pixel; the contrast of each patch is enhanced for better visual inspection; (c) the estimated steering kernel centered at each example pixel.

marked pixel (the contrast of these image patches is enhanced for better visual inspection). Fig. 1(c) shows the SKs corresponding to each patch. It can be seen that the SK assigns higher intensity to the pixel with a higher correlation to the centre pixel, so it tends to spread along the direction with a higher textural correlation to the centre pixel.

Based on SKs, we perform two analysis. The first analysis focuses on the general textural correlation. It aims to understand the general trend of iris textural correlation. The second analysis studies the distribution of local textural correlation. The aim is to gain the statistical knowledge on local textural correlation in iris images.

Furthermore, we investigate the method to utilize the information of iris textural correlation to improve iris recognition. Specifically, we utilize the textural correlation information to produce iris codes, in order to alleviate the influence of noise on individual bits in iris codes. We calculate the value of a specific bit using not only the Gabor filtering result on this bit, but also its neighbouring bits with high textural correlation to this bit. Experimentally, we find that the iris code produced using this method achieves an improved performance compared to traditional iris codes.

Contribution. Our contribution is two-fold. First, we make a quantitative analysis on the textural correlation in human iris images. To the best of our knowledge, this is the first work quantitatively analysing the iris textural correlation. It provides a deeper insight into the characteristic of iris textures. Second, we show that the information on iris textural correlation can be utilized in the stage of iris codes generation, improving iris recognition performance.

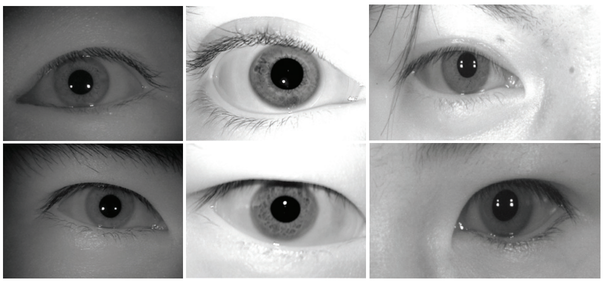

(a) CASIAT (b) ND0405 (c) CASIAD

Figure 2: Example images of the datasets used in the experiment.

The rest of the paper is organised as follows. In section 2, we introduce the datasets used for analysis and the pre-processing for the data. In section 3, we present the method and result of our analysis on iris textural correlation. In section 4, we describe the method to produce iris codes utilizing iris textural correlation, and we report the experimental result. Finally, in section 5, we conclude the paper.

2. Datasets and pre-processing

2.1. Datasets

We conduct experiments on three benchmark datasets: CASIA.v4 thousand (CASIAT) [1], ND-iris-0405 (ND0405) [14] and CASIA.v4 distance (CASIAD) [1]. As described below, these three datasets cover the iris captures with varying image quality. We show some example images of each dataset in Fig. 2.

CASIAT consists of 2000 eyes captured at a close distance. This dataset includes $20,000$ images with a high overall quality. The main noise factor is the specular reflection and glasses in some images. In our experiment, we use this dataset to represent the iris captures with a satisfactory quality. Correspondingly, we eliminate some noisy captures based on the iris segmentation results (see the pre-processing below). The final subset used in the experiment has $12,108$ images from 1790 eyes.

ND0405 is a dataset captured at a close distance including $64,982$ images from 712 eyes. Despite the close capturing, the images in this dataset suffer from noise due to real world conditions, including blurring, eyelids occlusion, specular reflection, rotation, off-angle, soft contact lenses, *etc*. We use ND0405 to represent the iris data with a relatively higher quality. The capture has good resolution and clear iris texture, but suffering from some noise and degradations. We use a subset of ND0405 consisting of the first 8 images of each eye, and we eliminate some images with too small iris region. It leads to a subset of 5613 images from 712 eyes used in the experiment.

CASIAD is a distantly captured dataset including both eyes from 142 subjects. The stand-off distance is around 3 meters. The images include most of facial features and patterns. The eye region of most images suffers from low resolution, blurring, blinking, eyelids occlusion, specular reflection, motion, *etc*. We use CASIAD to represent the iris data with a relatively lower quality. The captures suffer from the information loss and noise due to the distant capture. We conduct the experiment on a subset of CASIAD constructed by detecting left and right eye regions using Viola-Jones object detector [20], manually correcting false detections, and eliminating images with too small iris regions. It results a subset with 5037 images from 284 eyes for experiment.

From each dataset, we use a subset for parameter tuning, and we use the rest of the data for test. Specifically, for CASIAT, we use the images of the first 50 eyes for parameter tuning; for ND0405 and CASIAD, we use the images of the first 20 eyes for parameter tuning. The images of the rest eyes are used for test on each dataset.

2.2. Pre-processing

The pre-processing includes iris segmentation and normalization (unwrapping). For CASIAT, we seek limbic and pupillary boundaries using the Hough transform based segmentation in [12]. We eliminate the images with segmentation failures in these two boundaries. The reason is to keep the CASIAT subset representing the iris captures with a satisfactory quality, since the segmentation failure in these two boundaries is mainly due to the influence of noise. For the remaining images in CASIAT, we detect eyelids, reflections and shadow using the algorithm in [7]. We manually correct the segmentation failures in eyelids (see the method below).

For ND0405 and CASIAD, we perform the whole iris segmentation process using the algorithm in [7], and we manually correct the false segmentations. In other words, we preserve the noisy captures in these two datasets, enabling them to represent iris captures with varying quality.

The manual correction of segmentation failures is performed as follows. For eyelids, we manually label the correct boundary as the final boundary. In terms of limbic and pupillary boundaries, after manually labelling a correct boundary, we select the final boundary among a set of boundaries near to the manual one by maximizing the score of Daugman's integro-differential operator [3, 4]. Experimentally, we find that the above correction method for limbic and pupillary boundaries significantly improves the performance of iris recognition, compared to directly using the manually labelled boundary as the final boundary. We think the reason is that the position of limbic and pupillary boundaries sought by the integro-differential operator is more stable than the manually labelled boundary.

For all the datasets, we unwrap the segmented iris region using the rubber sheet model in [3, 4]. The size of unwrapped images is set to 100×360.

3. Analysis of iris textural correlation

In this section, we analyse the iris textural correlation using steering kernels (SKs) [17]. First, we describe the method to calculate SKs. Then, we present and discuss the result of analysis using SKs.

3.1. Steering kernels (SKs)

We use SKs to model the correlation between a pixel and its neighbouring regions. Let $\boldsymbol{x} \in \mathbb{R}^2$ be the coordinate of a centre pixel. The SK centered at $\boldsymbol{x}$ is defined as follows [17]:

$$\boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right) = \frac{\sqrt{\det\left(\boldsymbol{C}_i\right)}}{2\pi h^2} \exp\left(-\frac{\left(\boldsymbol{x}_i - \boldsymbol{x}\right)^T \boldsymbol{C}_i \left(\boldsymbol{x}_i - \boldsymbol{x}\right)}{2h^2}\right) \tag{1}$$

where $\boldsymbol{x}_i$ denotes the coordinate of a pixel in the neighbouring region of $\boldsymbol{x}$, $\boldsymbol{C}_i \in \mathbb{R}^{2\times 2}$ is a covariance matrix including the local shape information at $\boldsymbol{x}_i$, and h is a smoothing parameter. We directly use the value of $\boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right)$ as the correlation between two pixels at $\boldsymbol{x}_i$ and $\boldsymbol{x}$. A SK measures the correlation between two pixels from two aspects: distance and shape. In terms of distance, it can be seen that $\boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right)$ has a higher value if $\boldsymbol{x}_i$ is closer to $\boldsymbol{x}$. In other words, the closer a pixel to the centre pixel, the higher correlation they have. As for the shape, the value of $\boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right)$ is influenced by the covariance matrix $\boldsymbol{C}_i$ which includes the local shape information. The effect of $\boldsymbol{C}_i$ is similar to the covariance matrix of a Gaussian kernel.

The key to calculate a SK is the computation of $\boldsymbol{C}_i$. In [17], $\boldsymbol{C}_i$ is decomposed into three components, including scaling, rotation and elongation:

$$\boldsymbol{C}_i = \gamma_i \boldsymbol{U}_{\theta_i} \boldsymbol{\Lambda}_i \boldsymbol{U}_{\theta_i}^{\mathrm{T}} \tag{2}$$

where γ_i is the scaling parameter, U_{θ_i} is the rotation matrix and Λ_i is the elongation matrix. U_{θ_i} and Λ_i are further defined as follows:

$$\boldsymbol{U}_{\theta_i} = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix} \tag{3}$$

$$\boldsymbol{\Lambda}_i = \begin{bmatrix} \sigma_i & 0 \\ 0 & \sigma_i^{-1} \end{bmatrix} \tag{4}$$

In Eqn. 3 and Eqn. 4, θ_i and σ_i are rotation and elongation parameters, respectively.

Now, the problem becomes how to determine the three shape parameters γ_i, θ_i and σ_i. The three parameters can be estimated based on the dominant directions of local gradient fields. For a pixel at $\boldsymbol{x}_i$, we use $\boldsymbol{G}_i = \left[\boldsymbol{g}_x\left(\boldsymbol{x}_i\right), \boldsymbol{g}_y\left(\boldsymbol{x}_i\right)\right] \in$

$\mathbb{R}^{M\times 2}$ to denote the local gradient matrix of a patch centered at $\boldsymbol{x}_i$, where M is the number of pixels in this patch, $\boldsymbol{g}_x(\boldsymbol{x}_i)$ and $\boldsymbol{g}_y(\boldsymbol{x}_i)$ are column vectors of the gradient along x and y directions in this patch, respectively. The dominant directions of local gradient field can be sought by the truncated singular value decomposition of $\boldsymbol{G}_i$:

$$\boldsymbol{G}_i = \boldsymbol{U}_i \boldsymbol{S}_i \boldsymbol{V}_i^{\mathrm{T}} \tag{5}$$

where 2×2 orthogonal matrix $\boldsymbol{V}_i$ includes the vectors of dominant directions of local gradient field, and 2×2 diagonal matrix $\boldsymbol{S}_i$ includes the energy in dominant directions. Let $\boldsymbol{v} = [v_1, v_2]^{\mathrm{T}} \in \mathbb{R}^2$ be the column in $\boldsymbol{V}_i$ corresponding to the smaller singular value in $\boldsymbol{S}_i$. θ_i is obtained by:

$$\theta_i = \arctan(v_1/v_2) \tag{6}$$

Essentially, θ_i is calculated as the most common direction in the local gradient field. Note that the SK estimates local orientations based on the vertical and horizontal gradients of image patches. Since the vertical and horizontal gradients of image patches can be viewed as the output of two filters, such method to compute local orientations is similar to steerable filters which are able to estimate local orientations using the output of multiple basis filters [6].

The elongation parameter σ_i is determined using the energy in dominant directions. Let s_1 and s_2 be the larger and smaller eigen values in $\boldsymbol{S}_i$, respectively. σ_i is obtained by:

$$\sigma_i = \frac{s_1 + \lambda}{s_2 + \lambda} \tag{7}$$

where λ is a regularization parameter to preserve the shape of the SK when $s_1 \approx s_2 \approx 0$. Actually, $s_1/s_2 \gg 1$ means that the patch centered at $\boldsymbol{x}_i$ includes more edges. In this case, Eqn. 7 results that $\sigma_i \gg 1$, *i.e.* a more elongated SK. On the other hand, $s_1/s_2 \approx 1$ means that the patch mainly contains flat regions. Correspondingly, Eqn. 7 leads to a σ_i close to 1, keeping a nearly circular shape.

The scaling parameter γ_i is calculated by:

$$\gamma_i = \sqrt{\frac{s_1 s_2 + \lambda'}{M}} \tag{8}$$

where λ' is a regularization parameter to prevent γ_i from becoming 0. Since $\sqrt{s_1 s_2}$ is the geometric mean, Eqn. 8 leads to a larger γ_i in flat regions ($s_1 \approx s_2$) and a smaller γ_i in textured regions ($s_1 \gg s_2$).

The overall effect of the above calculation of γ_i, θ_i and σ_i is that the SK has a more circular shape and larger spreading in flatter regions, while it becomes narrow and spreads along the local edge direction in textured regions. This is reasonable as a model of the correlation between a pixel and its neighboring regions. For flatter regions, the pixel distant from the centre pixel is still possible to be correlated with the centre pixel, and the correlation between the centre pixel and pixels at different directions tends to be similar. It is consistent with a SK with a larger spreading and more circular shape. In contrast, for textured regions, the pixel distant from the centre pixel is less possible to be correlated with the centre pixel, since they are more likely to be separated by textures. Also, the correlation depends on the local texture direction. This is consistent with a narrow SK spreading along the local edge direction.

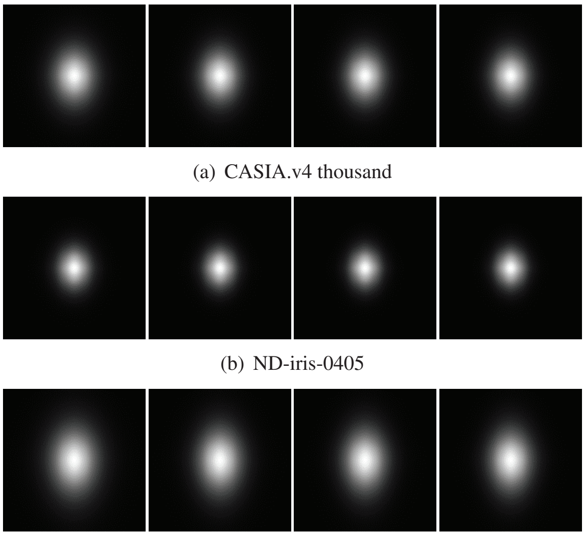
Fig. 1 shows some examples of SKs in. we can see that the shape of these SK is consistent with the above analysis in textured regions (the left two images in Fig. 1(b) and Fig. 1(c)) and flatter regions (the right two images in Fig. 1(b) and Fig. 1(c)). We refer to [17] for more examples and discussion on SK.

3.2. Analysing iris textural correlation using steering kernels (SKs)

This subsection presents the detailed analysis of iris textural correlation modelled by SKs. First, we describe the parameter setting to calculate the SKs for this analysis. Then, we analyse the general textural correlation. It seeks the knowledge on the general trend of iris textural correlation. Finally, we analyse the local textural correlation. It aims to understand the local characteristics of iris textures.

Parameter setting. The parameters to determine the SK are as follows: h in Eqn. 1, the size of SK, regularization parameters λ and λ' in Eqn. 7 and Eqn. 8, M which is the size of $\boldsymbol{G}_i$ (*i.e.* the size of the image patch to compute shape parameters γ_i, θ_i and σ_i). In this paper, we set h to 2, the size of SK to 15×15, λ to 1, λ' to $1e$-7. We select these parameters by visually examine the produced SK on the data for parameter tuning, with considering the parameter setting in [17]. In terms of M, this parameter influences the scale of analysis (*i.e.* the size of the patch used to calculate the local shape information). In our experiment, we vary M to investigate the textural correlation in different scales. Specifically, M is varied among 49, 121, 255 and 361, and they correspond to patch sizes of 7×7, 11×11, 15×15 and 19×19, respectively. In the rest of this paper, we refer to these patch sizes as scale, and we use scale instead of M.

General textural correlation (GTC). We use the mean of SKs to model the GTC. Specifically, given an unwrapped iris image, we calculate a set of SKs centered at each pixel. Therefore, each SK models the textural correlation between a specific pixel and its neighbouring region. We use the mean of these SKs (MSKs) to model the GTC in this iris image. Since each individual SK tends to spread along the direction with a higher correlation to the corresponding centre pixel (the pixels in this direction have higher intensity due to a higher correlation to the centre pixel), the MSK will spread along the direction with a generally higher correlation to all the pixels in an iris image. For a dataset with multiple images, we calculate a MSK for each image, and



(a) CASIA.v4 thousand

(b) ND-iris-0405

(c) CASIA.v4 distance

Figure 3: The mean of mean steering kernels (MMSK) calculated from each dataset with different scales; from left to right: the scale is 7×7, 11×11, 15×15 and 19×19.

we use the mean of all the MSKs (MMSK) to model the GTC in this dataset. To mitigate the influence of outliers (*i.e.* non-iris pixels like eyelids), we only consider the SKs centered at the pixel with less than 10% outliers in its neighbouring region whose size is equal to the scale parameter.

The results of our analysis are as follows. Firstly, we visually show the GTC in each dataset. In Fig. 3, we present the MMSK in each dataset with different scales. It can be seen that these MMSKs generally have an elliptical shape spreading along the vertical direction. It means that the GTC is stronger along the radial direction than the other directions in all the three datasets. In other words, the GTC tend to go along the radial direction.

Also, we numerically validate the direction of GTC in each dataset. We define the direction of a SK as the angle between the direction with the largest spreading and the positive horizontal direction. We compute the direction with the largest spreading by PCA (we normalize the range of this direction to $[0°, 180°]$). We use the direction of SK as the direction of iris textural correlation, since the direction with the largest spreading in a SK can be viewed as the direction with the strongest correlation to the centre pixel. Therefore, we calculate the direction of MMSK as the direction of GTC. To obtain an estimate of error, we partition each dataset into multiple folds, and we calculate MMSK and its direction for each fold. Tab. 1 reports the mean and standard deviation (std) of the directions of MMSKs. We find that the directions of all the MMSKs are very close to $90°$. This show a general radial correlation of iris textures.

Scale	CASIAT	ND0405	CASIAD
7×7	89.77 ± 0.06	89.73 ± 0.20	89.86 ± 0.04
11×11	89.87 ± 0.08	89.48 ± 0.04	89.92 ± 0.03
15×15	89.70 ± 0.04	89.67 ± 0.09	89.89 ± 0.06
19×19	89.75 ± 0.06	89.62 ± 0.20	89.88 ± 0.05

Table 1: The direction of the mean of mean steering kernels (MMSK) on each dataset with different scales (in degree).

Scale	CASIAT	ND0405	CASIAD
7×7	89.7 ± 3.5	89.8 ± 12.4	89.8 ± 3.8
11×11	89.7 ± 3.8	89.9 ± 12.7	89.8 ± 3.9
15×15	89.7 ± 4.3	89.9 ± 14.0	89.9 ± 4.0
19×19	89.7 ± 5.3	89.7 ± 15.4	90.0 ± 4.2

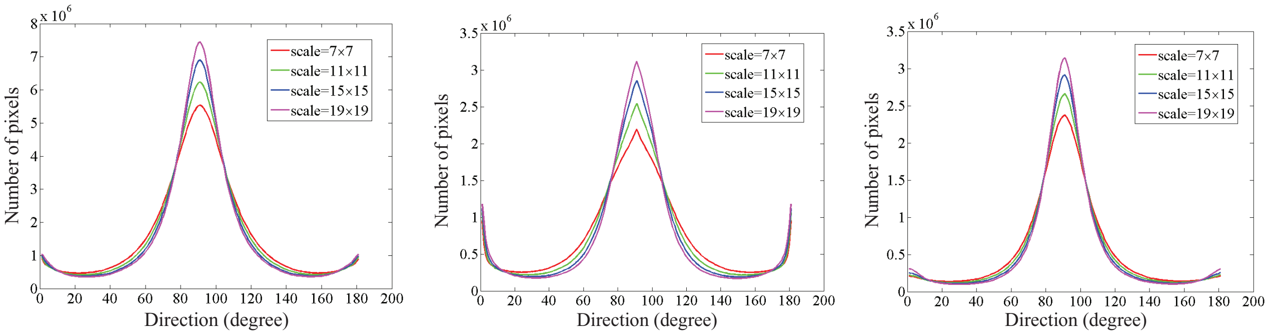
Table 2: The mean and standard deviation of the direction of mean steering kernel (MSK) on each dataset with different scales (in degree).

Moreover, the shape and direction of the MMSKs shown in Fig. 3 and Tab. 1 change little with scales. It means that the general radial correlation preserves in different scales.

Finally, we study the GTC in each iris image using the direction of mean SK (MSK). Given a dataset, we calculate a MSK for each image, and we calculate the direction of each MSK as the direction of the GTC in each image. In Tab. 2, we report the mean and std of the direction of MSK in each dataset. The result clearly shows that the GTC in each iris image approximately goes along the radial direction ($90°$) in all the datasets with different scales, despite of some variations. We also observe that, on each dataset, the mean direction changes little with scale, but the std increases when the scale gets larger. In our opinion, the reason is that a larger scale may induce more textural variation, so it increases the std of the direction of MSK. Interestingly, ND0405 dataset has a higher std than the other two datasets. It means that the iris texture in this dataset is more variant.

In all, Fig. 3 visually demonstrates that the iris texture has a general radial correlation in different scales, and it is quantitatively verified by the results in Tab. 1 and Tab. 2.

Local textural correlation (LTC). We also investigate the local characteristics of iris textural correlation. We perform this analysis based on the distribution of the direction of individual SKs. Specifically, given an iris image, we calculate a set of SKs centered at each pixel. For each individual SK, we calculate its direction as defined in the above analysis of GTC. These directions of local SKs reflect the textural correlation in different local regions of an iris image. Therefore, we use the histogram of these directions to represent the distribution of LTC in an iris image. For a



(a) CASIA.v4 thousand

(b) ND-iris-0405

(c) CASIA.v4 distance

Figure 4: The distribution of the direction of individual steering kernels on each dataset.

Scale	CASIAT	ND0405	CASIAD
7×7	90.0 ± 0.0, 21.8 ± 0.6, 157.0 ± 0.0	90 ± 0.0, 22.8 ± 0.4, 157.3 ± 0.5	90.0 ± 0.0, 23.7 ± 0.8, 156.3 ± 0.9
11×11	90.0 ± 0.0, 22.0 ± 0.0, 155.6 ± 0.8	90 ± 0.0, 26.7 ± 0.9, 154.0 ± 0.0	90.0 ± 0.0, 26.9 ± 0.3, 155.1 ± 0.3
15×15	90.0 ± 0.0, 24.0 ± 0.0, 154.1 ± 0.3	90 ± 0.0, 29.8 ± 0.6, 151.0 ± 0.0	90.0 ± 0.0, 27.1 ± 0.6, 153.8 ± 0.4
19×19	90.0 ± 0.0, 26.0 ± 0.0, 154.1 ± 0.3	90 ± 0.0, 32.0 ± 0.0, 148.8 ± 0.6	90.0 ± 0.0, 29.1 ± 0.3, 152.5 ± 0.7

Table 3: The direction corresponding to the peak and valleys in the distribution in Fig. 4 (in degrees, format: peak, left valley, right valley).

dataset, we calculate the overall distribution of LTC by accumulating the histogram obtained from each image in this dataset. We use a histogram consisting of 181 bins corresponding to the integer degrees between 0° and 180° (note that 0° and 180° are actually equivalent). Similarly to the analysis on the GTC, to alleviate the influence of outliers, we calculate the histogram only using the SKs centered on the pixel with less than 10% outliers in its neighbouring region whose size is equal to the scale parameter.

In Fig. 4, we plot the distribution on the three datasets with different scales. We also study the directions corresponding to the peak and left/right valleys (left/right valleys are the lowest point at the leff/right of the peak). To obtain an estimate of error, we partition each dataset into multiple folds; for each fold, we obtain a distribution of LTC and find peak and valleys; we report the mean and std of the directions of peaks and valleys obtained from all folds in Tab. 3. We have several observations from Fig. 4 and Tab. 3:

First, the majority direction of SKs concentrates around 90° at all the scales on all the datasets. The peak of all the distributions locates exactly at 90°. This shows that, locally, most of iris textures have radial or near-radial correlation.

Second, the distribution is more concentrated around 90° when the scale is larger. It means that the LTC is more radial if investigated on larger regions.

Third, interestingly, all the distributions in Fig. 4 are nearly symmetric. This is validated by the position of left and right valleys in Tab. 3. We can see that the two valleys are approximately symmetric to the radial direction (90°). It means that the structures with radially symmetric directions have a similar amount in human iris.

Fourth, the directions of left and right valleys are around 20°~30° to the horizontal direction (note that both 0° and 180° are horizontal directions). It means that human iris textures are most unlikely to spread along these two directions. Therefore, the direction around 20°~30° to the angular direction may be the most distinguishing direction for human iris textures, due to the highest randomness.

From the above analysis on GTC and LTC, we conclude that the LTC varies due to local characteristics, while the GTC tend to go along the radial direction.

Also, we observe that the larger scale leads to higher std (higher variation) in GTC (Tab. 2), but it results more concentrated distribution (lower variation) in LTC (Fig. 4). We think a possible reason is as follows. Locally, the larger scale is able to seek more radial correlation, so the corresponding distribution is more concentrated. However, the local radial correlation in the larger scale may be weaker than that in the smaller scale, since a larger region may exist higher textural variation. Such weaker local correlation increases the variation in GTC. In other words, although the larger scale is able to seek more SKs with the largest spreading along near-radial directions (*i.e.* more concentrated distribution of LTC), the high intensity pixels in these SKs are less concentrated around the radial direction compared to the SKs with a smaller scale (*i.e.* a weaker correlation).

Consequently, the MSK in the larger scale is more diffused. It leads to a larger variation in the direction of MSKs.

The individuality of LTC. We also investigate if LTC is person-specific. We measure the individuality by the discriminability index (DI) [3] calculated from the intra-class and inter-class distances between distributions of LTC of different iris captures. We used cosine distance as the distance between distributions of LTC. We find that the obtained DIs are generally around 0.6 for all the dataset at all the scales. It means that LTC has very slight individuality.

4. Improving iris recognition utilizing textural correlation

In this section, we investigate the method to improve iris recognition by utilizing the information on iris textural correlation. We show that an improvement can be achieved by utilizing this information to produce iris codes.

4.1. The proposed method

We compute a bit in an iris code not only using the Gabor filtering result at this bit, but also using its neighboring bits that have a higher textural correlation to this bit. The motivation is that the bits with higher textural correlation should be highly correlated in the iris code as well; by considering the highly correlated bits together to compute the iris code, we expect that we are able to suppress the flipping of individual bits due to noise, thus improving the performance.

Specifically, we use the traditional iris code as the initial iris code, and we compute the final iris code using the initial iris code and the textural correlation information. Given an unwrapped iris image, we extract the initial iris code using 1-D log-Gabor filter [12]. We operate on the iris code from real and imaginary parts separately. Without losing generality, we focus on the iris code from the real part in the following. Let $c_{\boldsymbol{x}}$ be the bit in the position $\boldsymbol{x}$ of an initial iris code from the real part of 1-D log-Gabor filtering. Let $\tilde{c}_{\boldsymbol{x}}$ be the corresponding bit of $c_{\boldsymbol{x}}$ in the final iris code. Recall that $\boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right)$ denotes the SK centered at $\boldsymbol{x}$. We compute $\tilde{c}_{\boldsymbol{x}}$ as follows:

$$\tilde{c}_{\boldsymbol{x}} = f_{\mathrm{bw}}\left(\frac{\sum\limits_{i} \boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right) c_{\boldsymbol{x}_i}}{\sum\limits_{i} \boldsymbol{K}\left(\boldsymbol{x}_i - \boldsymbol{x}\right)}, 0.5\right) \tag{9}$$

where $f_{\mathrm{bw}}\left(x, y\right)$ is a binarization function: $f_{\mathrm{bw}}\left(x, y\right) = 1$ if $x > y$, otherwise $f_{\mathrm{bw}}\left(x, y\right) = 0$.

Eqn. 9 performs a textural correlation-weighted majority voting with the SK as a model for textural correlation. It assigns higher weights to the bits with a higher textural correlation to the centre bit. The final value of the centre bit is determined using the majority in its neighboring bits after weighting. By doing this, we bound local bits together based on their textural correlation to mitigate the influence of noise on the individual bit.

4.2. Experimental analysis

Experiment setting. To study the effectiveness of the iris code produced using textural correlation information (ICTC) in section 4.1, we evaluate the performance of ICTC using the traditional iris code as baseline. We conduct experiments on the three datasets described in section 2. The partition of gallery and probe sets is as follows. For CASIAT, we use the first image of each eye as gallery and the rest images as probe. For ND0405, we use the first two images of each eye as gallery and the rest images as probe. For CASIAT, we use the first five images of each eye as gallery and the rest images as probe. The parameters of 1-D log-Gabor filtering (wavelength/sigmaOnf) are selected using the parameter tuning data described in section 2. Their values are as follows: 22/0.48 for CASIAT, 35/0.49 for ND0405, 19/0.48 for CASIAD.

As for the parameters to calculate the SK, we set the scale to 11×11. The setting of the other parameters are the same as that in section 3.2. Especially, the kernel size is 15×15. It means that, in Eqn. 9, $\boldsymbol{x}_i$ is restricted to a 15×15 square region centered at $\boldsymbol{x}$.

Results. We evaluate the performance using receiver operating characteristic (ROC). For each dataset, we randomly select some subjects from test data for 100 times, and we calculate the ROC curve for each random selection. We show the mean ROC curves of ICTC and baseline in Fig. 5. It can be seen that, on CASIAD dataset, ICTC has a better performance than baseline. On ND0405 and CASIAT datasets, ICTC performs better than the baseline when FAR is lower than 0.1%, and the performance of the two iris codes are similar when FAR is larger than 0.1%.

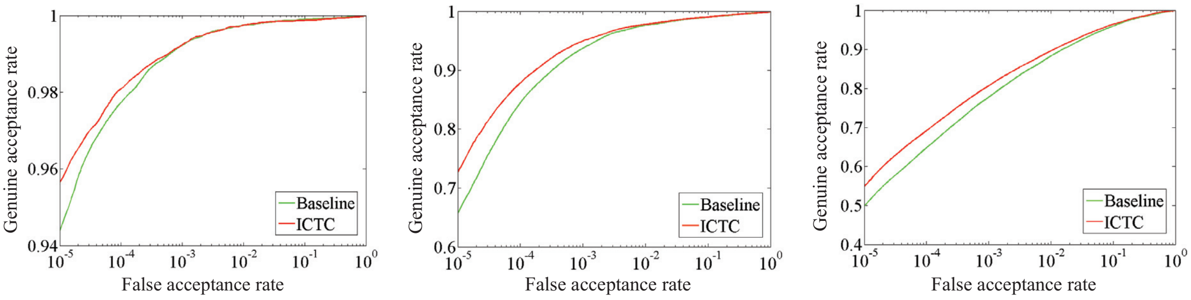
Also, we report the mean and standard deviation of genuine acceptance rate when the false acceptance rate is 0.01% (GAR@FAR=0.01%) in Tab. 4. It can be seen that ICTC has a better GAR when FAR= 0.01%. It demonstrates that the performance of iris recognition can be improved by utilizing the information of iris textural correlation.

5. Conclusion

In this paper, we study the textural correlation in human iris images. We find that the iris textural correlation has the following general and local characteristics:

General characteristic: the general correlation of iris textures approximately goes along the radial direction. The numerical direction of general textural correlation is very close to 90°, but it has some variations in different images.

Local characteristic: (1) the distribution of the direction of local textural correlation concentrates around the radial direction (90°), with the peak locating at 90°; (2) the texture with radially symmetric directions has similar amount; (3) the most distinguishing direction of iris images may be



(a) CASIA.v4 thousand

(b) ND-iris-0405

(c) CASIA.v4 distance

Figure 5: The ROC curves for comparison.

Method	CASIAT	ND0405	CASIAD
Baseline	97.7 ± 0.2	84.5 ± 2.8	64.8 ± 2.3
ICTC	98.1 ± 0.2	87.9 ± 2.0	69.2 ± 2.0

Table 4: GAR@FAR= 0.01% for comparison (percentage).

around 20°~30° to the angular direction, due to the highest randomness in textures along this direction.

Also, we demonstrate that it is possible to improve iris recognition using the information on textural correlation.

Future work may focus on utilizing the above local characteristic (3) for iris recognition, from the aspects like feature design and parameter selection.

References

[1] Biometrics ideal test. `http://biometrics.idealtest.org/dbDetailForUser.do?id=4`.

[2] K. Bowyer, K. Hollingsworth, and P. Flynn. Image understanding for iris biometrics: A survey. *Image Vision Comput.*, 110(2):281–307, 2008.

[3] J. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(11):1148–1161, 1993.

[4] J. Daugman. How iris recognition works. *IEEE Trans. Circuits Syst. Video Technol.*, 14(1):21–30, 2004.

[5] J. Daugman. Information theory and the iriscode. *IEEE Trans. Inf. Forensics Security*, 11(2):400–409, 2016.

[6] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(9):891–906, 1991.

[7] Y. Hu, K. Sirlantzis, and G. Howells. A robust algorithm for color iris segmentation based on 1-norm regression. *Int. Joint Conf. Biometrics*, 2014.

[8] A. Kumar and A. Passi. Comparison and combination of iris matchers for reliable persional authentification. *Pattern Recognit.*, 43:1016–1026, 2010.

[9] J. Liu, Z. Sun, and T. Tan. Code-level information fusion of low-resolution iris image sequences for persion identification at a distance. *IEEE Int. Conf. Biometrics: Theory, Appl. and Syst.*, 2013.

[10] N. Liu, M. Zhang, H. Li, Z. Sun, and T. Tan. DeepIris: learning pairwise filter bank for heterogeneous iris verification. *Pattern Recong. Lett.*, 2016.

[11] L. Ma, T. Tan, Y. Wang, and D. Zhang. Person idenfication based on iris texture analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(12):1519–1533, 2004.

[12] L. Masek and P. Kovesi. MATLAB source code for a biometric identification system based on iris patterns. *School Comput. Sci. Softw. Eng., Univ. Western Australia*, 2003.

[13] J. Matey, O. Naroditsky, K. Hanna, R. Kolczynski, D. LoIacono, S. Mangru, M. Tinker, T. Zappia, and W. Zhao. Iris on the move: acquisition of images for iris recognition in less constrained environments. *Proc. IEEE*, 94:1936–1947, 2006.

[14] P. Phillips, W. Scruggs, A. O'Toole, P. Flynn, K. Bowyer, C. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 large-scale experimental results. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32:831–846, 2009.

[15] H. Proenca and L. Alexandre. Toward convert iris biometric recognition: experimental results from the nice contests. *IEEE Trans. Inf. Forensics Security*, 7:798–808, 2012.

[16] H. Proenca, S. Filipe, R. Santos, J. Oliveira, and L. Alexandre. The UBIRIS.v2: a database of visible wavelength images captured on-the-move and at-a-distance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1529–1535, 2010.

[17] H. Takeda, S. Farsiu, and P. Milanfar. Kernel regression for image processing and reconstruction. *IEEE Trans. Image Process.*, 16(2):349–366, 2007.

[18] C. Tan and A. Kumar. Unified framework for automated iris segmentation using distantly acquired face images. *IEEE Trans. Image Process.*, 21(9):4068–4079, 2012.

[19] C. Tan and A. Kumar. Accurate iris recognition at a distance using stabilized iris encoding and zernike moments phase features. *IEEE Trans. Image Process.*, 23(9):3962–3974, 2014.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2001.

[21] R. Wildes. Iris recognition: an emerging biometric technology. *Proc. IEEE*, 85(9):1348–1363, 1997.