# Towards More Accurate Iris Recognition Using Deeply Learned Spatially Corresponding Features

Zijing Zhao, Ajay Kumar
Department of Computing, The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
jason.zhao@connect.polyu.hk, ajay.kumar@polyu.edu.hk

## Abstract

*This paper proposes an accurate and generalizable deep learning framework for iris recognition. The proposed framework is based on a fully convolutional network (FCN), which generates spatially corresponding iris feature descriptors. A specially designed Extended Triplet Loss (ETL) function is introduced to incorporate the bit-shifting and non-iris masking, which are found necessary for learning discriminative spatial iris features. We also developed a sub-network to provide appropriate information for identifying meaningful iris regions, which serves as essential input for the newly developed ETL. Thorough experiments on four publicly available databases suggest that the proposed framework consistently outperforms several classic and state-of-the-art iris recognition approaches. More importantly, our model exhibits superior generalization capability as, unlike popular methods in the literature, it does not essentially require database-specific parameter tuning, which is another key advantage over other approaches.*

## 1. Introduction

Iris recognition has emerged as one of the most accurate and reliable biometric approaches for the human recognition. Automated iris recognition systems therefore have been widely deployed for various applications from border control [22], citizen authentication [23], forensic [24] to commercial products [25]. The usefulness of iris recognition has motivated increasing research effort in the past decades for exploring more accurate and robust iris matching algorithms under different circumstances [1-6].

In recent years, deep learning has gained tremendous success especially in the area of computer vision, and accomplished state-of-the-art performance for a number of tasks such as general image classification [17], object detection [18] and face recognition [15] [19]. However, unlike face, in the field of iris recognition, in the best of to our knowledge, there is almost nil attention to incorporate remarkable capabilities of the deep learning and achieve superior performance than popular or state-of-the-art iris recognition methods.

In this paper we propose a new deep learning based iris recognition framework which not only achieves satisfactory matching accuracy but also exhibits outstanding generalization capability to different databases. With the design of effective fully convolutional network, our model is able to significantly reduce parameter space and learn comprehensive iris features which generalize well on different datasets. A newly developed *Extended Triplet Loss* (*ETL*) function provides meaningful and extensive supervision to the iris feature learning process with limited size of training data.

The main contributions of this paper can be summarized as follows: (i) We develop a new deep learning based iris recognition framework which can be highly generalized for operating on different databases that represent diverse deployment environments. A new *Extended Triplet Loss* function has been developed to successfully address the nature of iris pattern for learning comprehensive iris features (more details in Section 2.2 and 3). Significant advancement therefore has been made to bridge the gap between deep learning and iris recognition. (ii) Under fair comparison, our approach consistently outperforms several state-of-the-art methods on different datasets. Even under challenging scenario that without having any parameter tuning on the target dataset, our model can still achieve superior performance over state-of-the-art methods that have been extensively tuned.

### 1.1. Related Work

One of the most classic and effective approaches for automated iris recognition was proposed by Daugman [1] in 2002. In his work, Gabor filter is applied on the segmented and normalized iris image, and the responses are then binarized as *IrisCode*. The hamming distance between two *IrisCodes* is used as the dissimilarity score for verification. Based on [1], 1D log-Gabor filter was proposed in [2] to replace 2D Gabor filter for more efficient iris feature extraction. A different approach, developed in [3] in 2007, has exploited discrete cosine transforms (DCT) for analyzing frequency information of image blocks and generating binary iris features. Another frequency information based approach was proposed in [5] in 2008, in which 2D discrete Fourier transforms (DFT) was employed. In 2009, the multi-lobe differential filter (MLDF), which is a specific kind of ordinal filters, was proposed in [4] as an alternative to the Gabor/log-Gabor filters for generating iris templates.

Unlike the popularity of deep learning for various computer vision tasks, especially for face recognition, the literature so far has not yet fully exploited its potential for iris recognition. There has been very little attention on exploring iris recognition using deep learning. A deep representation for iris was proposed in [27] in 2015, but the purpose was for spoofing detection instead of iris recognition. A recent approach named DeepIrisNet in [28] has investigated deep learning based frameworks for general iris recognition. This work is essentially a direct application of typical convolutional neural networks (CNN) without much optimization for iris pattern. Our reproducible experimental comparison in section 5.3 further indicates that under fair comparison, this approach [28] cannot deliver superior performance even over other popular methods. Another recent work [37] has attempted to employ deep belief net (DBN) for iris recognition. Its core component, however, is the optimal Gabor filter selection, while the DBN is again a simple application on the *IrisCode* without iris-specific optimization. Above studies have made preliminary exploration but failed to establish substantial connections between iris recognition and deep learning.

### 1.2. Limitations and Challenges

Despite the popularity of iris recognition in biometrics, conventional iris feature descriptors does have several limitations. The summaries of earlier work in [7] and [8] reveal that existing methods can achieve satisfactory performance, but the performance needs to be further improved to meet the expectations for wider range of deployments. Besides, traditional iris features, such as *IrisCode*, are mostly based on empirical models which apply hand-crafted filters or feature generators. As a result, these models rely heavily on parameter selection when applied for different databases or imaging environments. Although there are some standards on iris image format [29], the selection of parameter for feature extraction remains empirical, or based on training methods such as boosting [30]. This situation can be observed from [4], where eight different combinations of parameters for ordinal filters delivered varying performance on three databases, or from [9] which employed two sets of parameters for log-Gabor filter on two databases by extensive tuning. Another limitation is that due to the simplicity of conventional iris descriptors, they are less promising to fully exploit the underlying distribution from various types of iris data available today. Learning data distribution from large amount of samples to further advance performance is one of the key trends nowadays.

Deep learning has the potential to address the above limitations, since the parameters in deep neural networks are learned from data instead of being empirically set, and deep architectures are known to have good generalization capability. However, new challenges emerge while incorporating typical deep learning architectures (*e.g.*, CNN) for the iris recognition, which can be primarily attributed to the nature of iris patterns. Different from face, iris pattern is observed to reveal little structural information or meaningful hierarchies. Iris texture is believed to be random [31]. Earlier promising works on iris recognition [1-5] mainly employed small-size filters or block-based operations to obtain iris features. Therefore, we can infer that the most discriminative information in the iris pattern comes from the local intensity distribution of an iris image rather than the global features, if any. CNN is known as effective for extracting features from low level to high level, and from local to global, due to the combination of convolutional layers and fully connected layers [20]. However, as discussed above, high level and global features may not be the optimal for iris representation.

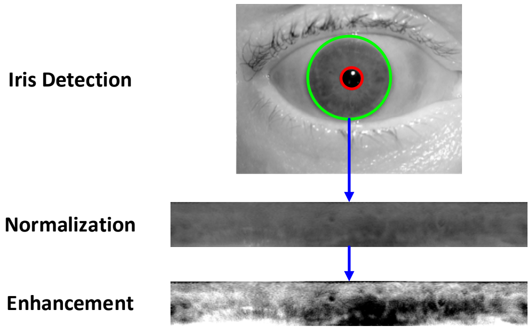


**Figure 1:** Illustration of key steps for iris image preprocessing.

This paper aims to develop more accurate and robust deep learning based iris feature representation framework, making solid contributions towards fully discovering the potential of deep learning for the iris recognition. Such objectives have not yet been pursued in the literature. Different from [28] and [37], this paper proposes a novel deep network and customized loss function, which are highly optimized for extracting discriminative iris features, which have been comparatively evaluated with several state-of-the-art methods on multiple iris image databases.

The rest of this paper is organized as follows: Section 2-4 detail the proposed approach in terms of network architecture, improved triplet loss function and feature encoding respectively; Section 5 presents the experimental configurations, results and analysis; finally, the key conclusions from this paper are presented in Section 6.

## 2. Network Architecture

We have developed a highly optimized and unified deep learning architecture, referred to as *UniNet*, for both iris region masking and feature extraction, which is based on fully convolutional networks (FCN) [15]. A new

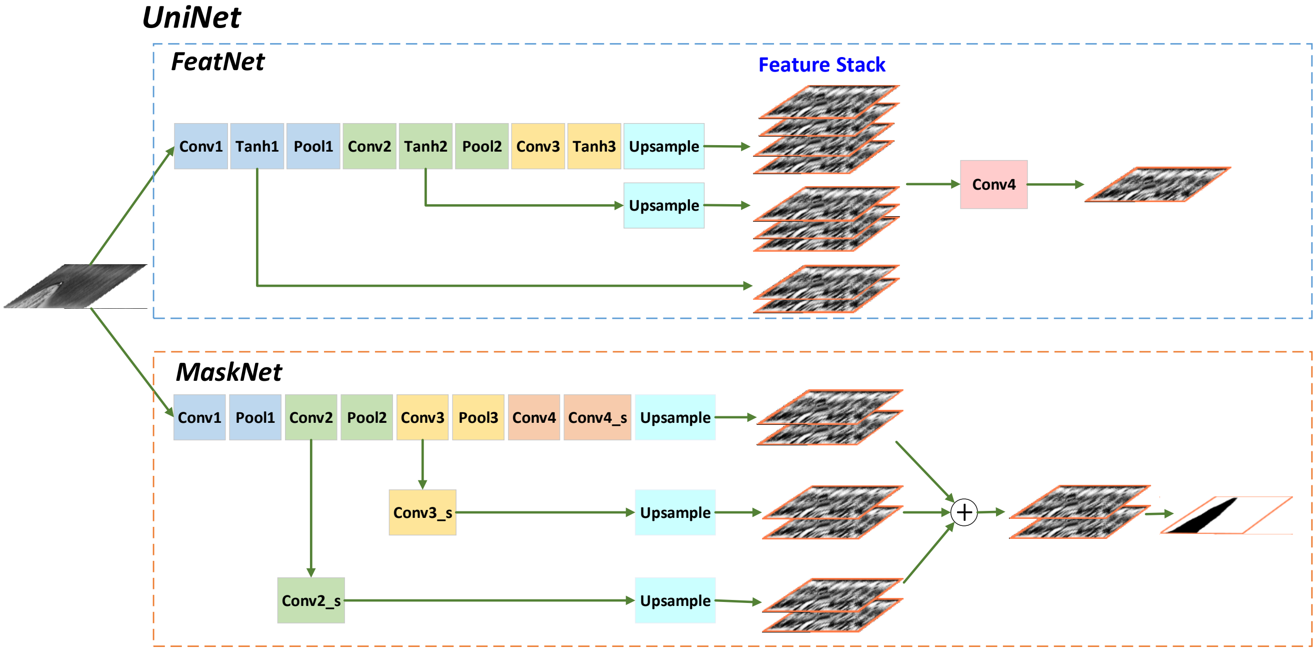


**Figure 2:** Detailed structures for *FeatNet* (top) and *MaskNet* (bottom) respectively. The *FeatNet* generates a single-channel feature map for each sample for matching. The *MaskNet* outputs a two-channel map, on which the values for each pixel along two channels represent the probabilities of belonging to iris and non-iris regions, respectively.

Table 1: Layer configurations for *MaskNet* and *FeatNet*.

| ***FeatNet*** | | | | |
|---|---|---|---|---|
| **Layer** | **Type** | **Kernel size** | **Stride** | **# Output channels** |
| Conv1 | Convolution | 3×7 | 1 | 16 |
| Conv2 | Convolution | 3×5 | 1 | 24 |
| Conv3 | Convolution | 3×3 | 1 | 32 |
| Conv4 | Convolution | 3×3 | 1 | 1 |
| Tanh1, 2, 3 | TanH activation | / | / | / |
| Pool1, 2, 3 | Average pooling | 2×2 | 2 | / |

| ***MaskNet*** | | | | |
|---|---|---|---|---|
| **Layer** | **Type** | **Kernel size** | **Stride** | **# Output channels** |
| Conv1 | Convolution | 3×3 | 1 | 16 |
| Conv2 | Convolution | 3×3 | 1 | 32 |
| Conv2_s | Convolution | 1×1 | 1 | 2 |
| Conv3 | Convolution | 3×3 | 1 | 64 |
| Conv3_s | Convolution | 1×1 | 1 | 2 |
| Conv4 | Convolution | 3×3 | 1 | 128 |
| Conv4_s | Convolution | 1×1 | 1 | 2 |
| Pool1, 2 | Max pooling | 2×2 | 2 | / |
| Pool3 | Max pooling | 4×4 | 4 | / |

customized loss function, named *Extended Triplet Loss (ETL)*, has been developed to accommodate the nature of iris texture in supervised learning. The motivations and technical details for the proposed approach are explained in the following sections.

## 2.1. Image Preprocessing

For all the experiments presented in this paper, we use a recent iris segmentation approach [10] for iris detection and normalization. The resolution after normalization is uniformly set to $64\times512$. We then apply a simple contrast enhancement process, which adjusts the image intensity so that 5% pixels are saturated at low and high intensities. The enhanced images are used as input to the deep network for training and testing. Figure 1 illustrates the key steps of image preprocessing.

## 2.2. Fully Convolutional Network

The proposed unified network (termed as *UniNet*) is composed of two sub-networks, *FeatNet* and *MaskNet*, whose detailed structures are presented in Figure 2 and Table 1. Both of the two sub-networks are based on fully convolutional networks (FCN) which is originally developed for semantic segmentation [15]. Different from common convolutional neural network (CNN), the FCN does not have fully connected layer. The major components of FCN are convolutional layers, pooling layers, activation layers, etc. Since all these layers operate on local regions around pixels from their bottom map, the output map can preserve *spatial correspondence* with the original input image. By incorporating up-sampling layers, FCN is able to perform pixel-to-pixel prediction. In the following we detail the two components of *UniNet*.

- ***FeatNet***

*FeatNet* is designed for extracting discriminative iris

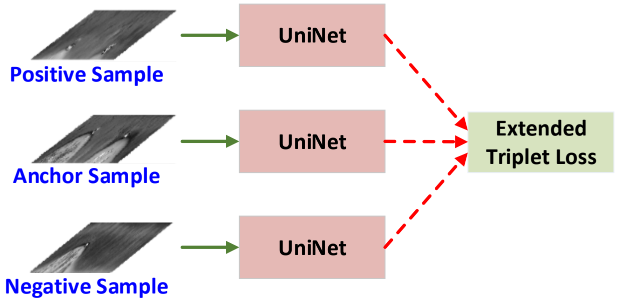


**Figure 3:** Triplet-based network organization for training.

features which can be used in matching. As shown in Figure 2, the input iris image is forwarded by several convolutional layers, activation layers and pooling layers. The network activations at different scales, i.e., TanH1-3, are then up - sampled if necessary to the size of original input. These features form a multi-channel feature stack which contains rich information from different scales, and are finally convolved again to generate an integrated single-channel feature map.

The reason for selecting FCN instead of CNN for iris feature extraction primarily lies in the previous analysis on iris patterns in Section 1.2, i.e., the most discriminative information of an iris probably comes from small and local patterns. FCN is able to maintain local pixel-to-pixel correspondence between input and output, and therefore is a better candidate for the iris feature extraction.

- ***MaskNet***

*MaskNet* is set to perform non-iris region masking for normalized iris images, which can be regarded as a specific problem for the semantic segmentation. It is basically a simplified version of the FCNs proposed in [15]. Similar to those in [15], *MaskNet* is supervised by a pixel-wise softmax loss, where each pixel is classified into one of two classes, *i.e.*, iris or non-iris. In our practice, *MaskNet* is trained with 500 randomly selected samples from the training set of ND-IRIS-0405 database, and the ground truth masks are manually generated by us. We would like to declare that the main focus of this paper is on learning effective iris feature representation. *MaskNet* is developed to provide adequate and immediate information for masking non-iris regions, which is necessary for the newly designed loss function (will be detailed in Section 3) and also for the matching process. The placement of *MaskNet* in the unified network also preserves the possibilities that iris masks may be jointly optimized/fine-tuned with the feature representations, which is one of our future research goals. At this stage, however, *MaskNet* is pre-trained and fixed during learning the iris features. A sample evaluation for its performance is provided in the *supplementary file*.

### 2.3. Triplet-based Network Architecture

A triplet network [16] was implemented for learning the convolutional kernels in *FeatNet*. The overall structure for the triplet network in the training stage is illustrated in Figure 3. As shown in the figure, three identical *Uninets*, whose weights are kept identical during training, are placed in parallel to forward and back-propagate the data and gradients for anchor, positive and negative samples respectively. The anchor-positive (AP) pair should come from the same person while the anchor-negative (AN) pair comes from different persons. The triplet loss function in such architecture attempts to reduce the anchor-positive distance and meanwhile increase the anchor-negative distance. However, in order to ensure more appropriate and effective supervision in the generation of iris features by the FCN, we improve the original triplet loss by incorporating a bit-shifting operation. The improved loss function is referred to as *Extended Triplet Loss* (*ETL*), whose motivation and mechanism are detailed in Section 3.

## 3. Extended Triplet Loss Function

### 3.1. Triplet Loss Function Incorporating with Masks and Bit-Shifting

The original loss function for a triplet network is defined as follows:

$$L=\frac{1}{N}\sum_{i=1}^{N}\left[\left\|\boldsymbol{f}^{A}_{i}-\boldsymbol{f}^{P}_{i}\right\|^{2}-\left\|\boldsymbol{f}^{A}_{i}-\boldsymbol{f}^{N}_{i}\right\|^{2}+\alpha\right]_{+} \qquad (1)$$

where $N$ is the number of triplet samples in a mini-batch, $\boldsymbol{f}^{A}_{i}$, $\boldsymbol{f}^{P}_{i}$ and $\boldsymbol{f}^{N}_{i}$ are the feature maps of anchor, positive and negative images in the *i*-th triplet respectively. The symbol $[\bullet]_{+}$ is the as same as used in [16] and is equivalent to $\max(\bullet,0)$. $\alpha$ is a preset parameter to control the desired margin between anchor-positive distance and anchor-negative distance. Optimizing above loss will lead to the anchor-positive distance being reduced and anchor-negative distance being enlarged until their margin is larger than a certain value.

In our case, however, using Euclidean distance as the dissimilarity metric is far from sufficient. As discussed earlier, we propose using spatial features which have the same resolution with the input, the matching process has to deal with non-iris region masking and horizontal shifting, which are frequently observed in iris samples as illustrated in Figure 4. Therefore in the following, we extend the original triplet loss function, which we refer to as the *Extended Triplet Loss* (*ETL*):

$$ETL=\frac{1}{N}\sum_{i=1}^{N}\left[D(\boldsymbol{f}^{A}_{i},\boldsymbol{f}^{P}_{i})-D(\boldsymbol{f}^{A}_{i},\boldsymbol{f}^{N}_{i})+\alpha\right]_{+} \qquad (2)$$

where $D(\boldsymbol{f}^{1},\boldsymbol{f}^{2})$ represents the *Minimum Shifted and Masked Distance (MMSD)* function, defined as follows:

$$D(\boldsymbol{f}^{1},\boldsymbol{f}^{2})=\min_{-B\le b\le B}\left\{FD(\boldsymbol{f}^{1}_{b},\boldsymbol{f}^{2})\right\} \qquad (3)$$

where $FD$ is the *Fractional Distance* which takes feature masks into consideration:

$$FD(\boldsymbol{f}^1, \boldsymbol{f}^2) = \frac{1}{|M|} \sum_{(x,y)\in M} (f^1{}_{x,y} - f^2{}_{x,y})^2$$
$$M = \{(x,y) \mid m^1{}_{x,y} \neq 0 \text{ and } m^2{}_{x,y} \neq 0\} \tag{4}$$

where $\boldsymbol{m}^1$ and $\boldsymbol{m}^2$ are the binary masks for two feature maps, in which zero means the current position is non-iris. In other words, *FD* only measures the distances at valid iris pixel positions, and normalizes the total distance by the number of valid pixels. In (3), the subscript $b$ means the feature map has been shifted horizontally by $b$ pixels, i.e., a shifted feature map has the following spatial correspondence with the original one:

$$\boldsymbol{f}_b[x_b, y] = \boldsymbol{f}[x, y]$$
$$x_b = (x - b + w) \bmod w \tag{5}$$

where $x, y$ are the spatial coordinates and $x_b$ is obtained by shifting the pixel to the left by a step of $b$. Note that when $x$ is less than $b$, the pixel position will be directed to the right end of the map, as the iris map is normalized by unwrapping the original iris circularly and the left end is therefore physically connected with the right end. When $b$ is negative, the bit-shifting operation would shift the map to the right by $-b$ pixels. The iris feature matching then is meaningful by computing the *MMSD* between feature maps. In order to maintain simplicity of the notations for the upcoming derivation, we denote the offsets that fulfills the *MMSD* of AP-pair and AN-pair as follows:

$$b_{AP} = \underset{-B\le b\le B}{\operatorname{argmin}} \left\{ FD(\boldsymbol{f}^A{}_b, \boldsymbol{f}^P) \right\}$$
$$b_{AN} = \underset{-B\le b\le B}{\operatorname{argmin}} \left\{ FD(\boldsymbol{f}^A{}_b, \boldsymbol{f}^N) \right\} \tag{6}$$

During the back-propagation (BP) of the training process, the gradients (or partial derivatives) of the new loss on the anchor, positive and negative feature maps need to be computed. For simplicity, let us firstly derive the partial derivative *w.r.t* the positive feature map $\boldsymbol{f}^P$. From (2) it can be derived that for one sample in the batch:

$$\frac{\partial ETL}{\partial \boldsymbol{f}^P} = \begin{cases} 0, & \text{if } ETL = 0 \\ \dfrac{1}{N} \dfrac{\partial ETL}{\partial D(\boldsymbol{f}^A, \boldsymbol{f}^P)} \dfrac{\partial D(\boldsymbol{f}^A, \boldsymbol{f}^P)}{\partial \boldsymbol{f}^P}, & \text{otherwise} \end{cases} \tag{7}$$

Again from (2) we can see that $ETL = 0$ is equivalent to $D(\boldsymbol{f}^A, \boldsymbol{f}^P) - D(\boldsymbol{f}^A, \boldsymbol{f}^N) + \alpha \le 0$. We only need to show the derivation when $ETL$ is not 0. Let us define the set of common valid iris pixel positions for AP pair as:

$$M_{AP} = \{(x,y) \mid \boldsymbol{m}^A[x,y] \neq 0 \text{ and } \boldsymbol{m}^P[x_{b_{AP}}, y] \neq 0\} \tag{8}$$

From (3), (4) we have the following pixel-wise derivatives:

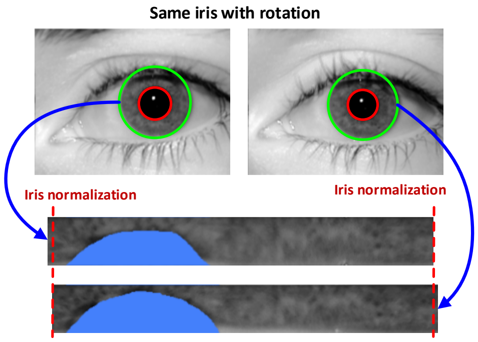


**Figure 4:** Illustration of occlusions (labeled in blue) and horizontal translation which usually exist between two normalized iris images even from a same iris.

$$\frac{\partial D(\boldsymbol{f}^A, \boldsymbol{f}^P)}{\partial \boldsymbol{f}^P[x,y]} = \frac{\partial FD(\boldsymbol{f}^A{}_{b_{AP}}, \boldsymbol{f}^P)}{\partial \boldsymbol{f}^P[x,y]}$$
$$= \begin{cases} 0, \text{ if } (x,y) \notin M_{AP} \text{ or } ETL = 0 \\ \dfrac{-2}{|M_{AP}|} (\boldsymbol{f}^A[x_{b_{AP}}, y] - \boldsymbol{f}^P[x,y]), \text{ otherwise} \end{cases} \tag{9}$$

And apparently $\dfrac{\partial ETL}{\partial D(\boldsymbol{f}^A, \boldsymbol{f}^P)} = 1$, thus from (7) and (9).

$$\frac{\partial ETL}{\partial \boldsymbol{f}^P[x,y]} = \begin{cases} 0, \text{ if } (x,y) \notin M_{AP} \text{ or } ETL = 0 \\ \dfrac{-2(\boldsymbol{f}^A[x_{b_{AP}}, y] - \boldsymbol{f}^P[x,y])}{N\,|M_{AP}|}, \text{ otherwise} \end{cases} \tag{10}$$

Similarly, for the partial derivatives on the negative feature map, we have:

$$\frac{\partial ETL}{\partial \boldsymbol{f}^N[x,y]} = \begin{cases} 0, \text{ if } (x,y) \notin M_{AN} \text{ or } ETL = 0 \\ \dfrac{2(\boldsymbol{f}^A[x_{b_{AN}}, y] - \boldsymbol{f}^N[x,y])}{N\,|M_{AN}|}, \text{ otherwise} \end{cases} \tag{11}$$

The final step is to calculate the derivatives *w.r.t* the anchor feature map. It can be seen from (3)-(5) that shifting the first map to the left by $b$ pixels is equivalent to shifting the second map to the right by $b$ pixels. Making use of this property, we have $FD(\boldsymbol{f}^A{}_{b_{AP}}, \boldsymbol{f}^P) = FD(\boldsymbol{f}^A, \boldsymbol{f}^P{}_{-b_{AP}})$ and $FD(\boldsymbol{f}^A{}_{b_{AN}}, \boldsymbol{f}^N) = FD(\boldsymbol{f}^A, \boldsymbol{f}^N{}_{-b_{AN}})$. It is therefore quite straightforward to obtain from (2)-(4):

$$\frac{\partial ETL}{\partial \boldsymbol{f}^A[x,y]} = -\frac{\partial ETL}{\partial \boldsymbol{f}^P[x_{-b_{AP}}, y]} + \frac{\partial ETL}{\partial \boldsymbol{f}^N[x_{-b_{AN}}, y]} \tag{12}$$

After calculating the derivative maps *w.r.t* $\boldsymbol{f}^A$, $\boldsymbol{f}^P$ and $\boldsymbol{f}^N$ respectively, the rest of the BP process is the same as for common convolutional neural networks. Above derivation shows that gradients will be computed only for pixels that are not masked. In this way, features are learned only within valid iris regions, while non-iris regions will be ignored since they are not of our interest. After the last convolutional layer, a single-channel feature map is generated which can be used to measure similarities between the iris samples.

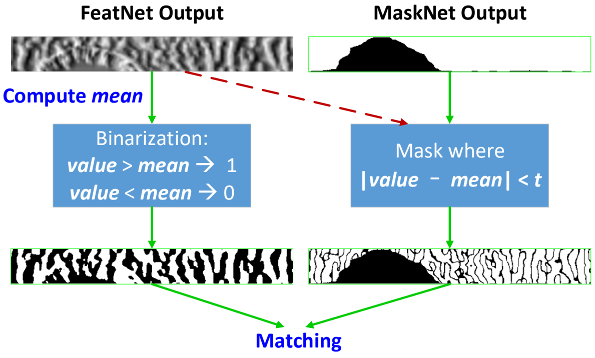


**Figure 5:** Illustration of feature binarization process.

# 4. Feature Encoding and Matching

We perform a simple encoding process for the feature map output from *UniNet*. The feature maps originally contain real values, and it is straightforward to measure the fractional Euclidean distance between the masked maps for matching, as the network is trained in this manner. However, binary features are more popular in most of the research works on iris recognition (e.g., [1]-[6], [9]), since it is widely accepted by the community that binary features are more resistant to illumination change, blurring and other underlying noise. Besides, binary features consume smaller storage and enable faster matching. Therefore, we also investigated the feasibility of binarizing our features with a reasonable scheme as described in the following:

For each of the output feature map, the mean value of the elements within the non-masked iris regions is firstly computed as $m$. This mean value is then used as the threshold to binarize the original feature map. In order to avoid marginal errors, elements with feature values $v$ close to $m$ (i.e., $|v-m|<t$) are regarded as less reliable and will be masked together with the original mask output by *MaskNet*. Such a further masking step is conceptually similar to "Fragile Bits" [12], which discovered that some bits in *IrisCode*, with filtered responses near the axes of the complex space, are less consistent or unreliable. The range threshold $t$ for masking unreliable bits is uniformly set to 0.6 for all the experiments. The feature encoding process can be demonstrated in Figure 5. For matching, we use the *fractional Hamming distance* [2] from the binarized feature maps and extended masks. It is observed that using the binary features does not degrade the performance compared with using the real-valued features, and even yield slight improvements in some cross-dataset scenarios, probably due to the factors discussed above.

# 5. Experiments and Results

Thorough experiments were conducted to evaluate the performance of the proposed approach from various aspects. The following sections detail the experimental settings along with the reproducible [38] results.

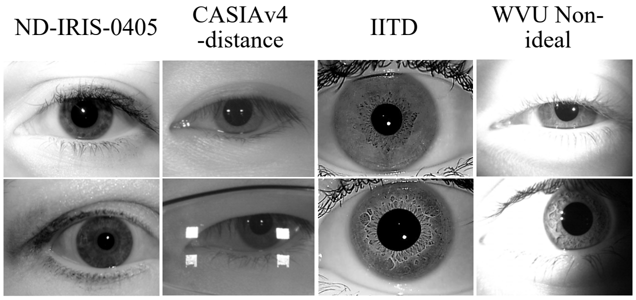


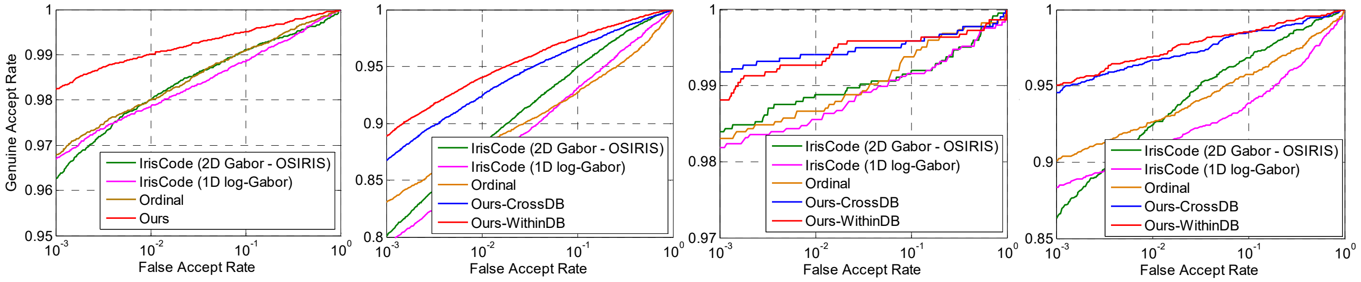
**Figure 6:** Sample raw images from four employed databases.

## 5.1. Databases and Protocols

We employed the following four publicly available databases our experiments:

- **ND-IRIS-0405 Iris Image Dataset (ICE 2006)**
  This database [32] contains 64,980 iris samples from 356 subjects and is one of the most popular iris databases in the literature. The training set for this database is composed of the first 25 left eye images from all the subjects, and the test set consists of first 10 right eye images from all the subjects. The test set, after removing some falsely segmented samples, contains 14,791 genuine pairs and 5,743,130 imposter pairs.
- **CASIA Iris Image Database V4 – distance**
  This database (subset) [33] includes 2,446 samples from 142 subjects. Each sample captures the upper part of face and therefore contain both left and right irises. The images were acquired from 3 meters away. An OpenCV-implemented eye detector [36] was applied to crop the eye regions from the original images. The training set consists of all the right eye images from all the subjects, and the test set comprises all the left eye images. The test set generates 20,702 genuine pairs and 2,969,533 imposter pairs.
- **IITD Iris Database**
  The IITD database [34] contains 2,240 image samples from 224 subjects. All of the right eye iris images were used as training set while the first five left eye images were used as test set. The test set contains 2,240 genuine pairs and 624,400 imposter pairs.
- **WVU Non-ideal Iris Database – Release 1**
  The WVU Non-ideal database [35] (Rel1 subset) comprises 3,043 iris samples from 231 subjects which were acquired under different extends of off-angle, illumination change, occlusions, etc. The training set consists of all of the right eye images, and the test set was formed by the first five left eye images from all the subjects. The test set has 2,251 genuine pairs and 643,565 imposter pairs.

From the above introduction we can observe that the imaging conditions for these databases are quite different. Sample images from the four employed datasets are provided in Figure 6, where noticeable variation in image



(a) ND-IRIS-0405 (b) CASIA.v4-distance (c) IITD (d) WVU Non-ideal

**Figure 7:** ROCs for comparison with other state-of-the-art methods on for employed databases. *Best viewed in color.*

**Table 2:** Summary of false reject rates (FRR) at **0.1%** false accept rate (FAR) and equal error rates (EER) for the comparison.

| | **ND-IRIS-0405** | | **CASIA.v4-distance** | | **IITD** | | **WVU Non-ideal** | |
|---|---|---|---|---|---|---|---|---|
| | **FRR** | **EER** | **FRR** | **EER** | **FRR** | **EER** | **FRR** | **EER** |
| IrisCode (OSIRIS) | 3.73% | 1.70% | 19.93% | 6.39% | 1.61% | 1.11% | 13.70% | 4.43% |
| IrisCode (log-Gabor) | 3.31% | 1.88% | 20.72% | 7.71% | 1.81% | 1.38% | 11.63% | 6.82% |
| Ordinal | 3.22% | 1.74% | 16.93% | 7.89% | 1.70% | 1.25% | 9.89% | 5.19% |
| Ours-CrossDB | / | / | 13.27% | 4.54% | **0.82%** | **0.64%** | 5.46% | 2.83% |
| Ours-WithinDB | **1.78%** | **0.99%** | **11.15%** | **3.85%** | 1.19% | 0.73% | **5.00%** | **2.28%** |

quality can be observed. It is therefore judicious to assume that these databases can represent diverse deployment environments.

## 5.2. Test Configurations

We incorporated following two configurations during the test phase for extensive evaluation of the proposed model.

- **CrossDB**
  In the *CrossDB* configuration, we use the ND-IRIS-0405 as the training set. During testing, the trained model was directly applied on CASIA.v4-distance and IITD *without any further tuning*. The purpose of the *CrossDB* setting is to examine the generalization capability of the proposed framework under challenging scenario that few training samples are available.
- **WithinDB**
  In this configuration we use the network trained on ND-IRIS-0405 as the initial model, then fine-tune it using the independent training set from the target database. The fine-tuned network is then evaluated on the respective test set. Being capable of learning from data is the key advantage of deep learning, therefore it is judicious to examine the best possible performance from the proposed model by fine-tuning it with some samples from the target database. The fine-tuned models from the *WithinDB* configuration are expected to perform better than the one with *CrossDB*, due to higher consistency of image quality between the training set and test set.

It should be noted that in both of the above configurations, training set and test set are totally separated, *i.e.*, none of the iris images are overlapping between the training set and test set. All the experimental results were generated under all-to-all matching protocol, *i.e.*, the scores of every image pair in the test set have been counted.

## 5.3. Comparison with Earlier Works

We present comparative experimental results using several highly competitive benchmarks. Gabor filter based *IrisCode* [1] has been the most widely deployed iris feature descriptor, largely due to the fact that few alternative iris features in the literature are universally accepted as better than *IrisCodes*. Instead, the majority of recent works on iris biometrics are more on improving segmentation and/or normalization models [10] [11], applying multi-score fusion [9] or feature bits selection [12]. In other words, in the context of iris feature representations, *IrisCode* is still the most popular and highly competitive approach, and therefore is definitely a fair benchmark for the performance evaluation. *IrisCode* has a number of advanced versions. From the publicly available ones, we selected OSIRIS [13], which is an open source tool for iris recognition. It implements a band of multiple tunable 2D Gabor filters that can encode iris patterns at different scales, therefore is a highly credible competitor. Another classic implementation of *IrisCode* is based on 1D log-Gabor filter(s) [2], which is claimed to encode iris patterns more efficiently, and is also widely chosen as benchmark in a variety of research works (*e.g.*, [6], [10]). Therefore, this approach is also investigated. Apart from the Gabor series filters, ordinal filters proposed in [4] can serve as a different type of iris feature extractors to complement the comparisons. The aforementioned benchmarks have been extensively tuned on target databases during testing to ensure as good performance as possible. For more details on tuning these methods, please refer to the attached *supplementary file*.

The comparison results are shown in Figure 7 and Table 2. Consistent improvements from our method over others can be observed on all of the four databases, under both *WithinDB* and *CrossDB* configurations. Such results suggest that the proposed iris feature representation not only achieves superior accuracy but also exhibits outstanding generalization capability. Even without additional parameter tuning, the well-trained model from our framework is promising to be directly used in deployment environments with varying image qualities. The relaxation of parameter tuning is apparently a highly desirable property for many real-life applications. An interesting finding is that on IITD database, the *CrossDB* model performs better even than the fine-tuned one. This is possibly because most of the images in IITD are with high qualities and less challenging, and its training set is not large enough, which causes slight over-fitting problem.

We also provide comparison with VeriEye SDK 9.0 [14], which is a commercial product for iris recognition and has gained considerable popularity due to its effectiveness. However, since it is not open source, there may be some underlying factors affecting the final results. Despite such concern, the comparison may still be interesting, and we include such results in our *supplementary file*.

### 5.4. Comparison with Other Deep Learning Configurations

In order to ascertain the effectiveness of the proposed network architecture for spatial feature extraction and the extended triplet loss, we also compared our method against typical deep learning architectures that are widely employed in various recognition tasks. The tested configurations are introduced in the following:

(i) *CNN+softmax/triplet loss*

CNN+softmax is the most widely employed deep learning configurations in the community, such as in [17] and [20]. Besides, CNN+triplet loss is gaining increasing popularity after it was proposed in [16], and therefore may also be interesting and worth evaluating. For the CNN model, we have chosen the popular VGG-16 which has achieved superior performance in face recognition.

(ii) *FCN+triplet loss*

Comparative evaluation has also been performed on using the proposed FCN (*FeatNet* only) and the original triplet loss function without incorporating bit-shifting and masking. Such comparison may assert the necessity of extending the original triplet loss.

(iii) *DeepIrisNet* [28]

We also compared our method against the recent deep learning based iris recognition framework, DeepIrisNet, which reports promising results. This architecture actually belongs to the CNN+softmax category, but we separately inspected it as it is directly proposed for iris recognition. Since the original model in their paper is not publicly available, we carefully implemented and trained the CNN according to all the details in [28].

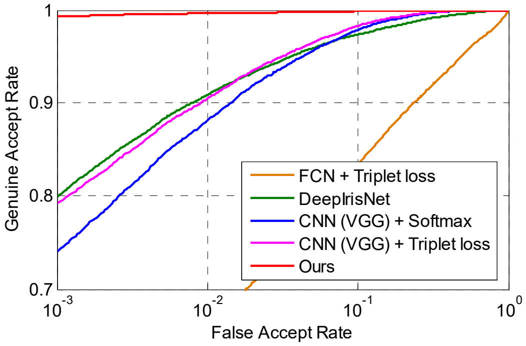


**Figure 8:** ROC curves for typical deep learning architectures available in the literature and our method on ND-IRIS-0405.

The comparison with aforementioned configurations was performed on ND-IRIS-0405 dataset, which has the largest number of training images among employed ones. The test set is kept consistent during the comparison. Hyper-parameters of the training processes for above architectures have been carefully investigated to achieve best possible performance. The results on the same test set are presented in Figure 8.

It can be observed from Figure 8 that our newly developed architecture significantly outperforms other deep learning configurations. CNN based configurations have failed to deliver satisfactory results especially at lower FAR. Such results support our previous analysis that global and high level features extracted by CNN may not be suitable for iris recognition. The poor performance from FCN+triplet loss strongly suggests that it is necessary to account for bit-shifting and non-iris region masking when learning spatially corresponding features through FCN.

We also evaluated the computational complexity of the proposed model. The results indicate that the execution time of our method can meet general real-time requirements, which is presented in our *supplementary file*.

## 6. Conclusions

This paper has developed a novel deep learning based iris feature representation which can offer superior matching accuracy and generalization capability for the iris recognition. The specially designed *Extended Triplet Loss* function can provide effective supervision for learning comprehensive and spatially corresponding iris features through the fully convolutional network. Further extension of this work should focus on learning more robust iris mask information through the deep networks, which is expected to further exploit the spatially corresponding features for more accurate iris recognition.

## Acknowledgment

This work is supported by the General Research Fund from the Hong Kong Research Grant Council grant no. 15206814 (PolyU 152068/14E).

# References

[1] J. Daugman, "How Iris Recognition Works", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21-30, Jan. 2004.

[2] L. Masek, "Recognition of Human Iris Patterns for Biometric Identification." *The University of Western Australia* 2, 2003.

[3] D. Monro, S. Rakshit and D. Zhang, "DCT-Based Iris Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 586-595, 2007.

[4] Z. Sun and T. Tan, "Ordinal Measures for Iris Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2211-2226, 2009.

[5] K. Miyazawa, K. Ito, T. Aoki, K. Kobayashi and H. Nakajima, "An Effective Approach for Iris Recognition Using Phase-Based Image Matching", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1741-1756, 2008.

[6] J. Pillai, V. Patel, R. Chellappa and N. Ratha, "Secure and Robust Iris Recognition Using Random Projections and Sparse Representations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1877-1893, 2011.

[7] M. Burge and K. Bowyer, *Handbook of Iris Recognition*, 2$^{nd}$ *Ed*. London: Springer, 2016.

[8] P. Grother, G.W. Quinn, M. L. Ngan and J. R. Matey, “IREX IV: Part 1, Evaluation of Iris Identification Algorithms”, *NIST Interagency/Internal Report (NISTIR)-7949*, Jul. 2013.

[9] A. Kumar and A. Passi, "Comparison and Combination of Iris Matchers for Reliable Personal Authentication", *Pattern Recognition*, vol. 43, no. 3, pp. 1016-1026, 2010.

[10] Z. Zhao and A. Kumar, "An Accurate Iris Segmentation Framework under Relaxed Imaging Constraints Using Total Variation Model." in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3828-3836.

[11] H. Proença, "Iris Recognition: On the Segmentation of Degraded Images Acquired in the Visible Wavelength", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1502-1516, 2010.

[12] K. Hollingsworth, K. Bowyer and P. Flynn, "The Best Bits in an Iris Code", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 964-973, 2009.

[13] N. Othman, B. Dorizzi and S. Garcia-Salicetti, "OSIRIS: An Open Source Iris Recognition Software", *Pattern Recognition Letters*, vol. 82, pp. 124-131, 2016.

[14] VeriEye SDK 9.0: http://www.neurotechnology.com/verieye.html

[15] J. Long, E. Shelhamer and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015, pp. 3431-3440.

[16] F. Schroff, K. Dmitry and P. James, "Facenet: A Unified Embedding for Face Recognition and Clustering", in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015.

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions", in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015, pp. 1-9.

[18] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014, pp. 580-587.

[19] Y. Sun, X. Wang and X. Tang, "Deeply Learned Face Representations are Sparse, Selective, and Robust." in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015, pp. 2892-2900.

[20] Y. Sun, X. Wang and X. Tang, "Deep Learning Face Representation from Predicting 10,000 Classes", in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014, pp. 1891-1898.

[21] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition." in *British Machine Vision Conference (BMVC)*, vol. 1, no. 3, p. 6, 2015.

[22] J. Daugman, "Iris Recognition Border-Crossing System in the UAE." *International Airport Review* 8, no. 2, 2004.

[23] J. Daugman, “600 Million Citizens of India are Now Enrolled with Biometric Id,” SPIE newsroom 7, 2014.

[24] NIST Presentation, “Forensic Data for Face & Iris”, http://biometrics.nist.gov/cs_links/standard/ansi-overview_2010/presentations/Forensic_data_for_Face_Iris.pdf

[25] H. King, "Galaxy Note 7 is First Samsung Device with Iris Scanner", *CNNMoney*, 2016. [Online]. Available: http://money.cnn.com/2016/08/02/technology/samsung-note-7/index.html. [Accessed: 09- Oct- 2016].

[26] L. Ma, T. Tan, Y. Wang and D. Zhang, "Efficient Iris Recognition by Characterizing Key Local Variations", *IEEE Transactions on Image Processing*, vol. 13, no. 6, pp. 739-750, 2004.

[27] D. Menotti, G. Chiachia, A. Pinto, W. Robson Schwartz, H. Pedrini, A. Xavier Falcao and A. Rocha, "Deep Representations for Iris, Face, and Fingerprint Spoofing Detection", *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 864-879, 2015.

[28] A. Gangwar and A. Joshi, "DeepIrisNet: Deep Iris Representation with Applications in Iris Recognition and Cross-Sensor Iris Recognition", in *Image Processing (ICIP), 2016 IEEE International Conference on*, 2016, pp. 2301-2305.

[29] ISO/IEC 19794-6:2011 (2011). Information technology -- Biometric Data Interchange Formats -- Part 6: Iris Image Data. Standard, International Organization for Standardization, Geneva, CH.

[30] Z. He, Z. Sun, T. Tan, X. Qiu, C. Zhong and W. Dong, "Boosting Ordinal Features for Accurate and Fast Iris Recognition", in *Computer Vision and Pattern Recognition (CVPR), 2008 IEEE Conference on*, 2008, pp. 1-8.

[31] J. Daugman, "The Importance of Being Random: Statistical Principles of Iris Recognition", *Pattern Recognition*, vol. 36, no. 2, pp. 279-291, 2003.

[32] K. Bowyer, and P. Flynn, "The ND-IRIS-0405 Iris Image Dataset", Notre Dame CVRL Technical Report, 2009.

[33] CASIA.v4 Iris Database: http://biometrics.idealtest.org/dbDetailForUser.do?id=4

[34] IITD Iris Database: http://www.comp.polyu.edu.hk/~csajaykr/IITD/Database_Iris.htm

[35] S. Crihalmeanu, A. Ross, S. Schuckers, L. Hornak, “A Protocol for Multibiometric Data Acquisition, Storage and Dissemination”, Technical Report, WVU, Lane Department of Computer Science and Electrical Engineering, 2007.

[36] OpenCV based face and eye detector: http://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.html

[37] F. He, Y. Han, H. Wang, J. Ji, Y. Liu and Z. Ma, "Deep Learning Architecture for Iris Recognition Based on Optimal Gabor Filters and Deep Belief Network", *Journal of Electronic Imaging*, vol. 26, no. 2, p. 023005, 2017.

[38] Web link to download the source code and executable files for the approach detailed in this paper: http://www.comp.polyu.edu.hk/~csajaykr/deepiris.htm

# Towards More Accurate Iris Recognition Using Deeply Learned Spatially Corresponding Features

## Supplementary File

This supplementary file aims to provide more details on the experimental configurations which could not be included in the paper due to space limitation.

## 1. Details on Training/Test Set Composition

The details on the division of the training set and the test set on the four employed databases are provided in Table S1. Both the training set and the test set are formed with the first $X$ ( $X = 25$, 10 or 5, shown in Table S1) or all of the left/right eye images from each of the subjects. If a subject has less than $X$ images in the respective database, then all images from this subject will be included.

Table S1: Summary of the division for training set and test set on the employed databases.

| Database | Training Set | | | | Test Set | | | |
|---|---|---|---|---|---|---|---|---|
| | #subjects | #images/subject | side | #images | #subjects | #images/subject | side | #images |
| ND-IRIS-0405 | all 356 | first 25 | left | 9,301 | all 356 | first 10 | right | 3,394 |
| CASIA.v4-distance | all 411 | first 25 | left | 6,840 | all 411 | first 10 | right | 3,939 |
| IITD | all 224 | all | right | 1,052 | all 224 | first 5 | left | 1,120 |
| WVU Non-ideal | all 231 | all | right | 1,511 | all 231 | first 5 | left | 1,137 |

During the training phase, the triplet-based architecture introduced in Section 2 requires the input data to be triplet sets (anchor-positive-negative entries) instead of single images. Therefore the training images in each of the databases need to be presented as triplet entries which are generated from the combinations of images. However, enumerating all the possible triplet combinations in the training set will lead to high storage and computational complexity, we therefore selectively generate part of the possible triplet entries for training, as described in the following: For each training set, we firstly enumerate all the possible anchor-positive (genuine) pairs, since the numbers of available genuine pairs are relatively small; for each anchor-positive pair, we randomly select five negative samples that are from different subjects than the anchor subject, and form the anchor-positive-negative triplet. In other words, each genuine pair in the training set will generate five triplet entries for training.

## 2. Details on Tuning Comparative Methods

We have extensively tuned the benchmarking methods (*i.e.*, OSIRIS [13], 1D log-Gabor [2] and ordinal [4]) to ensure that their best possible performances are used for the fair comparisons. We iteratively adopted possible combinations of the parameters for these approaches on each of the training sets within the empirically selected ranges, similar to as in many references (*e.g.*, [2], [4] and [9]). The best performing parameters on the training sets were then employed on the respective test sets for the performance evaluation.

- Parameters for *IrisCode* (OSIRIS 2D Gabor filters):

  A Gabor filter band containing six filters is provided in the original OSIRIS implementation [13]. In addition to the default one, we generated five Gabor filter bands for tuning this tool to obtain the best performance. Based on [1], a 2D Gabor filter for generating *IrisCode* can be formulated as:

  $$g(x,y)=e^{-(\frac{x^2}{\alpha^2}+\frac{y^2}{\beta^2})}e^{-i\omega x} \tag{1}$$

  Each set of parameters $(\alpha,\beta,\omega)$ can be used to produce two filters which are the real and imaginary parts of the complex filter kernel. We apply three sets of parameters to form a band of six filters. The five additional Gabor filter bands are then generated using the following parameters:

  (i) $(\alpha,\beta,\omega)\in\{(3, 1.5, 0.4\pi), (5, 1.5, 0.2\pi), (7, 1.5, 0.1\pi)\}$
  (ii) $(\alpha,\beta,\omega)\in\{(3, 1.5, 0.4\pi), (5, 1.5, 0.3\pi), (7, 1.5, 0.2\pi)\}$
  (iii) $(\alpha,\beta,\omega)\in\{(5, 2, 0.3\pi), (7, 2, 0.2\pi), (9, 2, 0.1\pi)\}$
  (iv) $(\alpha,\beta,\omega)\in\{(3, 2, 0.3\pi), (6, 2, 0.2\pi), (9, 2, 0.1\pi)\}$
  (v) $(\alpha,\beta,\omega)\in\{(5, 1.5, 0.3\pi), (7, 1.5, 0.2\pi), (9, 1.5, 0.1\pi)\}$

- Parameters for *IrisCode* (1D log-Gabor filter):

  Based on the model presented in [2], two parameters were tuned as follows:
  $\sigma/f$ (bandwidth over frequency): ranges from 0.3 to 0.6, with a step of 0.05.
  $\lambda$ (wavelength): ranges from 15 to 40, with a step of 1.
  **182** combinations in total.

- Parameters for ordinal filter based method:

  Based on the model presented in [4], four parameters were tuned as follows:
  *n* (number of lobes): ranges between {2, 3}.
  *s* (size of each lobe): ranges among {5, 7, 9}.
  *d* (distance between lobes): ranges among {5, 9, 13, 17}.
  $\sigma$ (standard deviation of each lobe): ranges among {1.5, 1.7, 1.9}.
  **72** combinations in total.

The best parameters automatically selected using the above detailed steps are provided in Table S2. It can be observed that such optimal parameters vary for one dataset to another, which underlines the *need* for selecting parameters for conventional methods according to the imaging environments and the quality of images for different databases. In contrast, our *CrossDB* model is able to deliver stable and satisfactory performance on the four public databases without any tuning, as shown in Figure 7 and Table 2 in the paper.

Table S2: Best performing parameters for *IrisCode* [2] [13] and Ordinal filters [4] on four employed databases.

| Method | Parameter | ND-IRIS-0405 | CASIA.v4-distance | IITD | WVU Non-ideal |
|---|---|---|---|---|---|
| *IrisCode*<br>(2D Gabor - OSIRIS) | config. | default | (iii) | (ii) | (i) |
| *IrisCode*<br>(1D log-Gabor) | $\sigma/f$<br>$\lambda$ | 18<br>0.45 | 24<br>0.35 | 18<br>0.4 | 15<br>0.55 |
| Ordinal filter | $n$<br>$s$<br>$d$<br>$\sigma$ | 3<br>5<br>9<br>1.9 | 3<br>9<br>13<br>1.7 | 2<br>7<br>5<br>1.9 | 3<br>9<br>5<br>1.9 |

It is worth mentioning that we did *not* use the original or built-in iris segmentation/normalization procedures from OSIRIS and Masek's 1D log-Gabor implementation. Iris segmentation has been shown to have significant impact on the recognition accuracy. Therefore to ensure the fairness in the evaluation of proposed iris feature representation, we uniformly adopt [10] for iris detection and normalization (as [10] has shown superior results on multiple public databases and also provides implementation codes), and use the output of *MaskNet* as the iris masks for our method and other investigated methods in this paper.

## 3. Analysis on Complexity

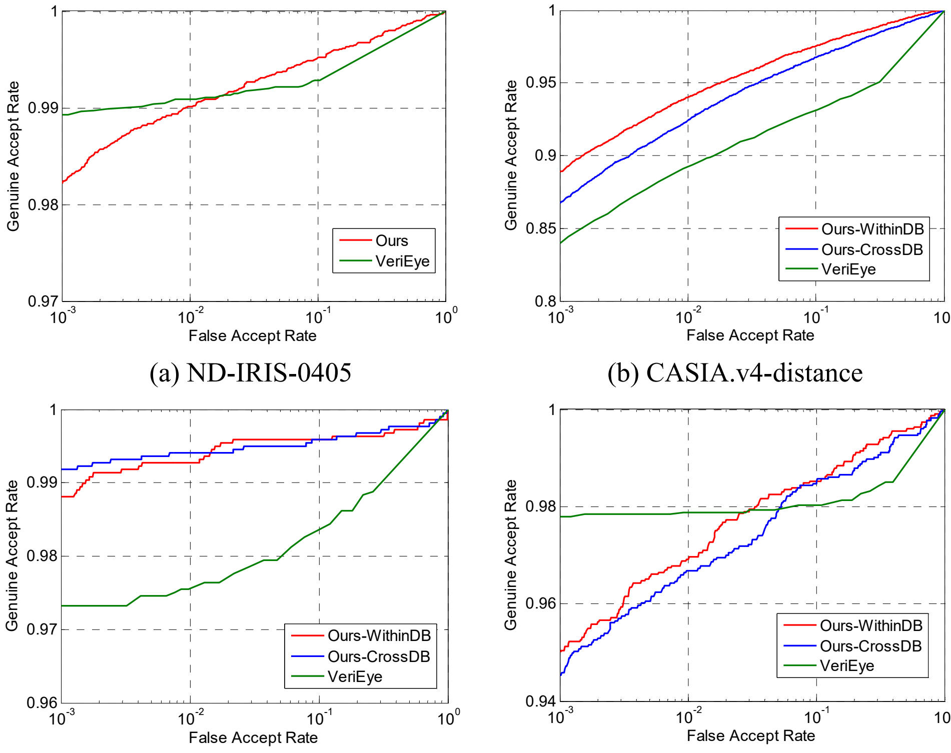
The computational complexity of our model has been evaluated in order to address the potential concerns on the feasibility for the deployment. Since our FCN does not employ fully connected layers, the number of parameters is significantly reduced and therefore it is much spatially simpler than conventional CNN based architectures. Table S3 summaries the computational time for feature extration and the storage required by our model, as compared with the CNN based approach in [28]. It can be noted that the space and time complexities for our approach are quite small.

Table S3: Summary of number of parameters, model storage size and feature extraction time per image, run with Matlab wrapper and C++ implementation, on Intel i7-4770 CPU, 16G RAM and Nvidia GTX670 GPU.

| Approach | #Parameters | Model Size (Byte) | Feature Extraction Time | |
|---|---|---|---|---|
| | | | GPU | CPU |
| Ours | ~ 110.7 K | 1.5 M | 7.6 ms | 236 ms |
| DeepIrisNet [28] | ~55,420 K | 289.0 M | 12.7 ms | 335 ms |

## 4. Comparison with VeriEye SDK 9.0

In this paper we have provided reproducible performance comparison with the *IrisCode* [2] [13] and ordinal filter based method [4]. Although these methods are widely cited and have shown to offer competitive performance in the literature, it can be interesting to provide comparison with some commercial solutions for iris recognition, as they are considered to be more suitable and optimized for real-life deployment. We therefore performed comparative evaluation using a popular commercial product, VeriEye iris recognition SDK from Neurotechnology [14], which released the latest version 9.0 in October 2016 and is available with us. The VeriEye SDK accepts original eye images (without normalization) as input and has its built-in iris segmentation components. Since this software is not open-source for its core functions, we are not able to interference its iris segmentation process. Therefore, the comparison results presented in this section may not be fully representing the effectiveness of iris feature representation, which is the key focus of this paper. Instead, it can be a sample reference for overall performance evaluation. The results for the comparison are shown in Figure S1.



(a) ND-IRIS-0405

(b) CASIA.v4-distance

(c) IITD

(d) WVU Non-ideal

Figure S1: Receiver operating characteristic (ROC) curves from our approaches and the commercial product VeriEye SDK on four databases.

As shown in the figure, on ND-IRIS-0405 and WVU Non-ideal databases, VeriEye has better genuine accept rates (GAR) at lower false accept rates (FAR), while our approach consistently outperforms VeriEye on CASIA.v4-distance and IITD datasets. As discussed earlier, the difference in the segmentation process may have certain impact on the presented recognition results. Besides, VeryEye has a built-in quality assessment function that it does not match images with low quality, which may improve its overall performance to a certain extent, while our approach does not evaluate image quality at the current stage. Considering above factors, it is judicious to believe that our prototype model can already offer highly competitive performance compared with the well optimized commercial system.

## 5. Sample Evaluation for *MaskNet*

As mentioned in the paper, our key focus is on learning more effective iris feature representation. *MaskNet* is an essential component of *UniNet* for providing immediate and appropriate non-iris masking information to the proposed *Extended Triplet Loss* (*ETL*) function. In order to assert the adequateness of the masking information during the feature learning process, we have performed a sample evaluation of *MaskNet*. For the evaluation benchmark, we use more recent and advanced iris segmentation approach [10] as this method has already provided comparison with other promising methods in the literature. Similar to as used in [10], the average segmentation error is defined as follows:

$$\bar{\mathrm{e}} = \frac{1}{N \times w \times h} \sum_{i=1}^{N} \sum_{x=1}^{w} \sum_{y=1}^{h} \boldsymbol{M}_i[x, y] \oplus \boldsymbol{G}_i[x, y] \tag{2}$$

where $N$ is the number of test images, $w$ and $h$ are the weight and height of each image respectively, $\boldsymbol{M}_i$ is the $\boldsymbol{i}$-th computed mask and $\boldsymbol{G}_i$ is the corresponding ground truth mask that should be manually generated. $\oplus$ is the exclusive-OR (XOR) operation for two binary values. This error rate measures the level of disagreement between the computed mask and the ground truth. The difference with [10] is that we measure the segmentation error after iris normalization.

The *MaskNet* employed in our experiments was trained with 500 randomly selected left eye images from ND-IRIS-0405 database, with manually labeled iris masks as the ground truth. The test sets for its evaluation are also generated from the same database, excluding the training samples. We used the following two sets for the testing: (a) 100 randomly selected samples and their ground truth masks manually created by us; (b) 792 samples and their ground truth masks which are available from a public iris segmentation ground truth database, IRISSEG-EP [H. Hofbauer, F. Alonso-Fernandez, P. Wild, J. Bigun and A. Uhl, “A ground truth for iris segmentation”. *ICPR*, 2014]. The average segmentation errors of *MaskNet* and [10] are shown in Table S4.

Table S4: Comparison of average segmentation errors from *MaskNet* and [10].

| **Approach** | **Average Segmentation Error** | |
|---|---|---|
| | **Set (a)** | **Set (b)** |
| *MaskNet* | 5.89% | 9.00% |
| ICCV [10] | 6.73% | 11.83% |

The results shown in Table S4 suggest that for both test sets, the developed *MaskNet* can achieve superior segmentation accuracy compared with state-of-the-art iris segmentation approach. It is therefore reasonable to conclude that *MaskNet* is able to provide appropriate information for identifying valid iris region during the feature learning process via *ETL*.