This WACV workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Alpha-wolves and Alpha-mammals: Exploring Dictionary Attacks on Iris Recognition Systems

Sudipta Banerjee[1], Anubhav Jain[1], Zehua Jiang[1], Nasir Memon[1], Julian Togelius[1], Arun Ross[2]
[1]New York University
{sb9084, aj3281, zj2086, memon, julian.togelius}@nyu.edu
[2]Michigan State University
rossarun@msu.edu

Abstract

A dictionary attack in a biometric system entails the use of a small number of strategically generated images or templates to successfully match with a large number of identities, thereby compromising security. We focus on dictionary attacks at the template level, specifically the IrisCodes used in iris recognition systems. We present an hitherto unknown vulnerability wherein we mix IrisCodes using simple bitwise operators to generate alpha-mixtures —alpha-wolves (combining a set of "wolf" samples) and alpha-mammals (combining a set of users selected via search optimization) that increase false matches. We evaluate this vulnerability using the IITD, CASIA-IrisV4-Thousand and Synthetic datasets, and observe that an alpha-wolf (from two wolves) can match upto 71 identities @FMR=0.001%, while an alpha-mammal (from two identities) can match upto 133 other identities @FMR=0.01% on the IITD dataset.

1. Introduction

In a dictionary attack, a small number of biometric samples or templates are strategically generated such that they successfully match with a large number of identities. Dictionary attacks on biometric recognition systems were first described in the context of fingerprints [26], where the authors demonstrated the vulnerability of *small-sized* sensors that enroll *multiple* low-resolution fingerprint samples. The authors synthesized fingerprint "templates" using a brute force approach that could match a large proportion of identities in an unseen population. They further devised the latent variable evolution method to generate *Deep MasterPrints* [5]. Following the success of masterprints, the feasibility of dictionary attacks in other biometric modalities were explored. *MasterFaces* [28] examined the vulnerability of face recognition systems to dictionary attacks with reasonable success. However, the authors in [29] pointed out the limited generalizability of face-based dictionary attacks across matchers, and contended that these attacks become less effective with increase in dimensionality of the facial representation. Dictionary attacks on speaker verification systems were introduced in [18].

Motivation: Iris recognition is deployed in many applications due to its high accuracy and fast matching [4,8]. The *biometric menagerie* [31] highlights that the *wolf-like* [11] behavior of an individual (*i.e.*, a single person fortuitously matches multiple people in a zero-effort imposter attack) is dominantly an image-specific issue stemming from non-ideal image acquisition in iris recognition systems, and not particularly a subject-specific issue. Iris image acquisition guidelines [22] dictate specific requirements (iris radius $\geq$ 80 pixels, rotation $< 15^{\circ}$ and daytime illumination). However, with commercial sensors mounted on hand-held devices, factors such as stand-off distances, indoor *vs.* outdoor, and occlusions (*e.g.*, eyeglasses) result in non-ideal conditions that can make iris templates extracted from such images vulnerable to false matches. The wolf-attack probability [30] can be also increased by using either a set of real or synthetic biometric samples that can adversarially match several users. We hypothesize that by carefully selecting users and further mixing their IrisCodes to form *alpha-mixtures* can significantly increase the success of dictionary attack. In this work, we adopt two strategies to generate alpha-mixtures at template level, (i) combine wolves inherent in the population to generate alpha-wolves, and (ii) combine users selected via a search-based scheme (with or without wolves) to generate alpha-mammals.

Contributions: (a) We conduct a *novel* vulnerability analysis of iris recognition systems that uses mixed IrisCodes, referred to as alpha-mixtures, to launch dictionary attacks at the template level. The mixture comprises alpha-wolves and alpha-mammals that are highly effective in matching arbitrary users not used in mixing. (b) We adopt different strategies for i) IrisCode selection (wolf selection

and search optimization), ii) mixing of IrisCodes (using simple bitwise logical operators and CNN-based mixing) and iii) evaluation (log-Gabor and spatial Gabor encoding) on multiple datasets. (c) We examine the utility of synthetic IrisCodes as dictionary attacks against real IrisCodes. We demonstrate the effectiveness of our method under limited knowledge assumptions with cross-attacks.

2. Related Work

2.1. Preliminaries

Iris recognition involves these steps [15]. 1) Iris image *acquisition* uses specialized sensors operating in the near-infrared spectrum (700-900nm). 2) Iris *segmentation* extracts the colored annular region between the limbus and pupillary boundaries. 3) Iris *encoding*, $\mathcal{E}(\cdot)$, obtains a compact template known as the IrisCode[1] using texture representation schemes such as Gabor filters. The IrisCode is typically a binary feature vector consisting of 2,048 bits that encodes the phase representation of the iris texture. Other types of encoding schemes have also been developed [16, 17, 19, 20, 24]. 4) Iris *matching*, $\mathcal{S}(\cdot,\cdot)$, uses fractional Hamming distance to measure the proportion of disagreement of the bits between two IrisCodes to produce a decision of match or non-match depending on the threshold, τ, at a selected False Match Rate (FMR).

Daugman's IrisCode [7] achieves extremely low FMR (1 in 26 million at HD=0.32) due to its high entropy [9] while maintaining fast matching using bitwise-Hamming distance. There has been a slew of other iris recognition methods using traditional filtering schemes, such as log-Gabor filters (LG) [19], spatial Gabor filters (QSW) [17], local-intensity variations [24], DCT-based analysis [20], cumulative sum based analysis [16]. Deep learning based iris recognition systems perform end-to-end matching with comparable matching accuracy and are not limited to binary features or Hamming distance. Refer to [21] for a survey of deep learning based iris recognition methods. Note that DL-based iris recognition typically do not use binary IrisCodes which is the focus of this work. In this work, we implement dictionary attacks using open-source implementation of *lg* and *qsw*-based features [25].

2.2. Morphing *vs.* Mixing

Erdogan [12] proposed generating a dual-identity iris image by creating a composite of two irides using multiple strategies. Rathgeb and Busch [23] proposed *morphing* two IrisCodes that matches the individuals whose IrisCodes contributed to the mixture. They performed random bit substitution, random row substitution and stability-based bit substitution to demonstrate that morphed IrisCodes can result in a fractional Hamming distance < 0.32. Similarly, [27] demonstrated that iris images from the left eye class can be morphed with images from the right eye class resulting in $> 90\%$ successful attacks. Note that our method performs *mixing* of IrisCodes using a function, $\mathcal{M}(\{IC_k\})$, where $k \geq 2$. See the details of mixing in Sec. 3. Unlike morphing, the mixed IrisCode can spoof multiple *other* identities, and not just the inputs to the mixing function.

[1] Typically, the term "IrisCode" has been associated with the Daugman-method [7] for extracting iris templates; however, in this work, we use it to indicate any binary code extracted from the iris.

3. Proposed Method

Combining IrisCodes requires three inputs: (i) a set of seed IrisCodes, (ii) the number of seed IrisCodes to be combined, and (iii) a mixing function to combine the seed IrisCodes. We describe the inputs for the two methods developed in this work below.

3.1. Method I: Generating Alpha-wolves

We ideally want the alpha-mixture to behave as wolves that causes a high number of biometric collisions. A simple yet effective way of ensuring the wolf-like behavior of the mixture is to begin with a set of disjoint wolves as seed IrisCodes. The strategic selection of wolves as seed IrisCodes brings us to the concept of Doddington's zoo [11]. The biometric menagerie classifies those individuals as *wolves* who successfully match other people (zero-effort imposter attack) resulting in high false matches. We utilize this phenomenon to rationalize our selection of wolves that will serve as the optimal set of seed IrisCodes. The *best* number of wolf (seed) IrisCodes to be combined is a hyper-parameter determined during evaluation. We select a fixed set of seed IrisCodes that match imposters at a false match rate (*e.g.*, FMR=0.01%). Next, we combine wolves following $\binom{n}{k}$, where n denotes the number of wolves for a dataset (training set) $\mathcal{D}_{tr}$, $|\mathcal{D}_{tr}| = d$, and $k = \{2, \cdots, n\}$. We select bitwise operations, *viz.*, $AND(\&), OR(|)$ and $XOR(\oplus)$ operators for mixing IrisCodes. Therefore, we use the mixing function as follows: $\mathcal{M}_1(IC_1 \& IC_2); \mathcal{M}_1(IC_1 | IC_2); \mathcal{M}_1(IC_1 \oplus IC_2)$ for $k = 2$. We select bitwise operators due to the binary nature of the IrisCode. Refer to Lines 1-14 in Algo. 1. The algorithmic time-complexity of wolf selection is $O(d)$ as it involves a single pass over the training set. The algorithmic time-complexity of wolf mixing is $O(kl)$, where $k * l << d$. We combine k seed wolves using l mixing functions resulting in $O(kl)$ time complexity for generating alpha-wolves.

3.2. Method II: Generating Alpha-mammals

We hypothesize that while mixing wolves leads to "alpha-wolves", more optimal combination of samples might exist in the dataset that does not necessarily involve wolves. We can consider a wolf as a point in the template space that is close to several other identities, simul-

taneously. It behaves as a centroid with maximal overlap with other identities. However, combining wolves may inadvertently push the mixed idenitity, *i.e.*, the "alpha-wolf", away from its optimal position, thereby reducing proximity with other individuals. This brings us to the idea of combining other members of the Doddington's zoo, such as mixing a wolf with a sheep (low False Match Rate and low False Non-Match Rate), that may lead to a more optimal dictionary attack. We *search* for this optimal set of IrisCodes to be combined using a second methodology coined "alpha-mammals". This hill-climbing approach makes no assumption about the need for multiple alpha-wolves nor on the number of IrisCodes that should be combined. We define the user coverage for an IrisCode IC from the training set $\mathcal{D}_{tr}$ at a threshold τ as follows.

$$Cov(IC) = \sum_{i \in \mathcal{D}_{tr}} [\mathcal{S}(i, IC) \leq \tau] \tag{1}$$

The *state space* is defined as the set of IrisCodes that are mixed together using some function $\mathcal{M}$. The *action space* is defined as either the deletion or addition of any IrisCode to the current state space. Finally, the reward is defined as the user coverage described in Eqn. 1. The algorithm starts with an empty set. In the first iteration, it always picks the wolf sample that leads to maximal coverage. Next, the algorithm optimizes over the set of users in $\mathcal{D}_{tr}$. Refer to Lines 15-46 in Algo. 1. The algorithmic time-complexity is $O(dtl)$, where d is the size of the training set, t is the number of iterations used and l is the number of mixing functions.

Algorithm 1: Generating Alpha-Mixtures

Data: $\mathcal{D}_{tr}$, $\mathcal{E}(\cdot)$, $\mathcal{S}(\cdot,\cdot)$, τ and $\mathcal{M}(\{\cdot\})$
Result: Set of Alpha-wolves: α_w, Alpha-mammals: α_m

1 **Generating Alpha-wolves:**
2 *Step I: Wolf selection*
3 $\{IC_1, \cdots, IC_d\} \leftarrow \mathcal{E}(\mathcal{D}_{tr}), |\mathcal{D}_{tr}| = d$; /* IrisCodes */
4 **if** $\mathcal{S}(IC_i, IC_j) \leq \tau \; \forall j$ *and* $i \neq j$; /* check for false match */
5 **then**
6 $\mathcal{W} \leftarrow IC_i$; /* seed wolves */
7 **end**
8 *Step II: Wolf mixing* ; /* $|\mathcal{W}| = n$ */
9 **for** $k = 2$ **to** n **do**
10 $\mathcal{C}_k \leftarrow \binom{n}{k}$; $\mathcal{P}_k \leftarrow \mathcal{W}\{\mathcal{C}_k\}$; /* select combinations of seed wolves */
11 **for** $l = 1$ **to** 3 **do**
12 $\alpha_{w_{kl}} \leftarrow \mathcal{M}_l(\{\mathcal{P}_k\})$; /* mix seed wolves with 3 bitwise ops. */
13 **end**
14 **end**
15 **Generating Alpha-mammals:**
16 $q \leftarrow \phi$; /* initialize an empty IC set */
17 $c \leftarrow 0$; /* set current coverage to 0 */
18 $c_n \leftarrow 0$; /* set best neighbor coverage to 0 */
19 **for** $l = 1$ **to** 3 **do**
20 **while** $c_n \geq c$ **do**
21 $c \leftarrow c_n \; q \leftarrow q_n$
22 *Step I: Check if appending sample helps.* ;
23 **for** $k = 1$ **to** $|\mathcal{D}_{tr}|$ **do**
24 $q' \leftarrow q.append(IC_k)$
25 $IC \leftarrow \mathcal{M}_l(q')$; /* mix set of ICs */
26 $c_k = \sum_{i \in \mathcal{D}_{tr}} [\mathcal{S}(i, IC) \leq \tau]$; /* cov on new IC */
27 **if** $c_k \geq c_n$; /* cov improved, update cov and IC set */
28 **then**
29 $c_n \leftarrow c_k$
30 $q_n \leftarrow q.append(IC_k)$
31 **end**
32 **end**
33 *Step II: Check if removing sample helps.* ;
34 **for** $k = 1$ **to** $|q|$ **do**
35 $q' \leftarrow q.remove(IC_k)$
36 $IC \leftarrow \mathcal{M}_l(q')$; /* mix set of ICs */
37 $c_k = \sum_{i \in \mathcal{D}} [\mathcal{S}(i, IC) \leq \tau]$; /* cov on new IC */
38 **if** $c_k \geq c_n$; /* cov improved, update cov and IC set */
39 **then**
40 $c_n \leftarrow c_k$
41 $q_n \leftarrow q.remove(IC_k)$
42 **end**
43 **end**
44 **end**
45 $\alpha_{m_l} \leftarrow \mathcal{M}_l(q)$
46 **end**
47 **Return:** α_w,α_m ; /* alpha-mixtures */

4. Experimental Design

We use two real iris datasets, namely, **IITD** [2], and **CASIA-IrisV4-Thousand** [1] and a synthetic iris dataset, namely, **CASIA-IrisV4-Synthetic** [1]. We consider left and right eye images as separate classes for both real datasets. The synthetic dataset has only a single eye for each identity. The IITD dataset comprises 224 subjects with 1,188 left eye images and 1,052 right eye images that is used for both wolf selection and evaluation. We use USIT v3.0.0 toolkit[2] to perform iris segmentation (using contrast-adjusted Hough transform (*caht*)), IrisCode feature extraction (using log-Gabor (*lg*) and quadrature spline wavelet (*qsw*)), and matching (using fractional Hamming distance (*hd*)). Note that USIT v3.0 converts the binary IrisCode to an 8-bit unsigned integer in [0, 255] and then flattens it to produce a 1-D vector of $1,280$ bytes ($2^3 \times 2^7 \times 10$ equivalent to $20 \times 512 = 10,240$ bits). Although the original authors indicated possible correlations between the first and last 10 rows [23], we use the entire template. We perform matching using the 1-D flattened representation while mixing is performed on the binary valued 2-D template.

CASIA-IrisV4-Thousand comprises 1,000 subjects with 10,000 left eye images and 10,000 right eye images. We use the COTS Neurotechnology VeriEye 12.4 SDK for segmentation, Libor Masek code for texture and mask generation, and USIT v3.0 for encoding and matching. We discard in-

[2]We use the Windows executable provided by the original authors at https://www.wavelab.at/sources/USIT/ in Method I and custom built Linux executable in Method II.

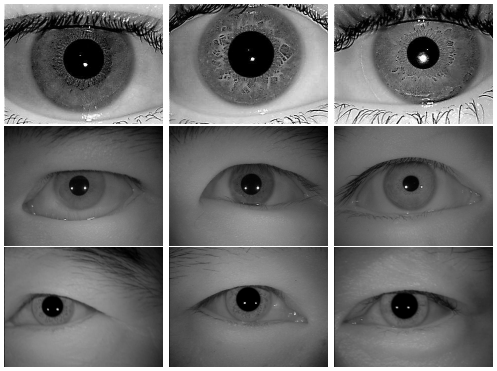


Figure 1. Examples of wolves used in generating alpha-wolves belonging to IITD (top row), CASIA-IrisV4-Thousand (middle row) and CASIA-IrisV4-Synthetic (bottom row) datasets. Note that, visually, wolves are high quality iris images.

admissible IrisCodes upon visual inspection (*e.g.*, all 0's). Next, we select wolves in 1:10 and 3:10 ratio from the entire set of 1,000 identities to form the training set, resulting in first 93 subjects using lg @FMR=0.05%, and the first 265 subjects using qsw features @FMR=0.01% (some identities were manually discarded due to unreliable encoding). We adopt this strategy to examine if wolves selected from a subset of the population (training set) can be effective against unseen subjects from the same dataset (test set). This helps us evaluate the effectiveness of the method with access to limited number of wolf samples. In this case, the user coverage is evaluated on the remainder of the target population (test set) excluding the subset from which the wolves were selected. See examples of wolves in Fig. 1. For computing the alpha-mammals, we search across the first 93 users and test on the remaining set of 907 users on the CASIA-IrisV4-Thousand dataset for both the qsw and lg features.

5. Results and Analysis

5.1. Results for Method I: Alpha-wolves

We select that mixed IrisCode that produces the highest false matches across the test set as the *best* alpha-wolf, *i.e.*, $\alpha_{w,best} = \arg\max_{\alpha_{w_k}} \mathcal{S}(\alpha_{w_k}, IC_q); \forall q \in \mathcal{D}_{te}$. Here, α_{w_k} denotes the set of alpha-wolves corresponding to mixing k out of n seed wolves, $\mathcal{S}(\cdot,\cdot)$ is the iris matcher, and $\mathcal{D}_{te}$ denotes the test set. We report the proportion of identities where at least one sample matched with the *best* alpha-wolf generated using different bitwise operators (AND, OR, XOR) and different feature extraction schemes (*lg* and *qsw*) on both eyes (left and right). We refer to this proportion as the user coverage that quantifies the success of the dictionary attack. We observe that as the number of wolves increase, the OR-mixture results in denser codes, while the AND-mixture results in sparser codes and the XOR-mixture contains equal proportion of 1's and 0's in the alpha-wolves. So, we restrict to mixing two, three and four seed wolves.

Table 1. Alpha-wolf attacks on IITD lg-Left dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.9/0.9/**10.7**	0.9/0.9/**20.5**	1.8/1.8/**79.0**	2.7/3.1/**89.7**
3	0.0/0.0/0.0	0.0/0.0/0.9	2.2/2.2/0.9	1.8/3.1/0.4
4	0.0/0.0/5.8	0.0/0.0/6.2	1.8/1.8/50.4	0.9/3.1/64.3

Table 2. Alpha-wolf attacks on IITD lg-Right dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.9/0.9/**17.9**	0.9/0.9/**21.4**	2.2/1.8/**82.1**	2.7/4.0/84.4
3	0.0/0.0/0.0	0.0/0.0/0.0	2.2/2.7/0.9	3.1/3.6/0.9
4	0.0/0.0/15.6	0.0/0.0/17.4	2.2/1.8/79.9	2.7/3.1/**88.3**

Table 3. Alpha-wolf attacks on IITD qsw-Left dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.9/0.9/**20.9**	0.9/0.9/**25.0**	2.2/2.2/**91.5**	4.5/3.6/**96.4**
3	0.0/0.0/0.0	0.0/0.0/0.4	3.1/2.7/1.3	4.5/3.6/1.8
4	0.0/0.0/4.0	0.0/0.0/5.4	2.7/3.1/83.0	4.0/2.2/94.2

Table 4. Alpha-wolf attacks on IITD qsw-Right dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.9/0.9/**31.7**	0.9/0.9/**42.4**	2.2/2.7/**88.8**	4.5/3.6/**90.6**
3	0.0/0.0/0.0	0.0/0.4/0.0	2.7/3.1/0.9	4.0/4.0/0.9
4	0.0/0.4/13.4	0.0/0.4/16.9	2.2/2.7/88.8	3.6/4.0/90.1

We present results on the IITD dataset in Tables 1,2,3 and 4. We observe that the overall user coverage is the highest for the XOR operator for mixing IrisCodes. We report the results at four $FMR(\%)$ values= $\{0.001, 0.01, 0.1, 1\}$ on IITD. As observed in the results, *qsw* feature-based IrisCode is more successful in generating dictionary attacks compared to log-Gabor (*lg*) feature-based IrisCode. Our attack achieves up to 31.7% user coverage @FMR=0.001% by XOR-ing two identities on *qsw*-right, 42.4% user coverage @FMR=0.01% by XOR-ing two identities on *qsw*-right, 91.5% user coverage @FMR=0.1% by XOR-ing two identities on *qsw*-left dataset, and up to 96.4% user coverage @FMR=1% by XOR-ing two identities on *qsw*-left.

We present results on CASIA-IrisV4-Thousand (for brevity we will refer it as CASV4-Th) dataset at $FMR(\%) = \{0.01, 0.05, 0.1, 1\}$ for lg-feature in Tables 5 and 6, and at $FMR(\%) = \{0.001, 0.01, 0.1, 1\}$

Table 5. Alpha-wolf attacks on CASV4-Th lg-Left dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.01%	0.05%	0.1%	1%
2	0.0/2.2/0.0	0.0/**57.1**/1.3	0.3/**64.4**/3.9	33.6/**98.3**/68.0
3	0.0/**2.4**/0.9	0.0/56.5/2.2	0.0/63.9/2.3	1.0/97.7/6.6
4	0.0/2.4/0.0	0.0/55.8/0.7	0.0/62.5/0.7	0.0/97.2/3.1

Table 6. Alpha-wolf attacks on CASV4-Th lg-Right dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.01%	0.05%	0.1%	1%
2	0.3/1.0/**10.0**	0.4/2.7/**67.9**	0.6/4.2/**70.3**	3.2/25.3/**93.7**
3	0.2/1.0/0.2	0.3/3.3/1.4	0.3/4.2/1.8	1.0/15.8/2.9
4	0.2/0.5/2.2	0.2/3.0/45.7	0.2/4.1/51.0	0.5/12.4/78.6

Table 7. Alpha-wolf attacks on qsw-CASV4-Th Left dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.0/0.3/**0.5**	0.0/0.8/**2.4**	0.8/2.7/**26.4**	54.0/32.8/**99.5**
3	0.0/0.3/0.0	0.0/0.7/0.0	0.5/2.5/1.4	32.8/25.4/8.4
4	0.0/0.3/0.0	0.0/0.7/1.7	0.0/2.3/5.6	10.6/18.5/77.7

Table 8. Alpha-wolf attacks on qsw-CASV4-Th Right dataset.

# seed wolves	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
	0.001%	0.01%	0.1%	1%
2	0.0/6.3/0.0	0.3/**72.8**/0.5	10.7/**97.3**/10.6	73.1/**99.3**/88.6
3	0.0/**9.5**/0.0	0.1/72.7/0.6	2.4/97.2/0.7	45.9/99.3/73.8
4	0.0/9.5/0.0	0.1/61.7/0.3	0.4/93.2/1.2	11.6/97.8/24.7

for qsw-feature in Tables 7 and 8. This is because we observe inherently poor performance on the CASV4-Th dataset, so we selected wolves at FMR=0.05% when applicable. Alpha-wolves achieve upto 9.5% user coverage @FMR=0.001% by AND-ing three identities using *qsw* feature on the CASV4-Th Right dataset, 72.8% user coverage @FMR=0.01% by AND-ing two identities using *qsw* features on the CASV4-Th Right dataset, 67.9% @FMR=0.05% by XOR-ing two identities using *lg* features on the CASV4-Th Right dataset, 97.3% user coverage @FMR=0.1% by AND-ing two identities using *qsw* feature on the CASV4-Th Right dataset, and upto 99.5% user coverage @FMR=1% by XOR-ing two identities using *qsw* features on the CASV4-Th Left dataset.

5.2. Results for Method II: Alpha-mammals

In this section, we present our results on dictionary attacks using the "alpha-mammal" hill climbing algorithm. In Table 9, we present the results obtained on the IITD dataset for the lg and qsw features at $FMR(\%) = \{0.01, 0.1, 1.0\}$.

Table 9. Alpha-mammal attacks on the IITD dataset.

feature-laterality	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR		
	0.01%	0.1%	1%
lg-Left	0.9/0.9/51.6	28.7/57.8/28.7	53.8/76.7/53.8
lg-Right	0.9/0.9/4.1	4.1/4.1/85.1	11.3/12.6/90.1
qsw-Left	0.9/0.9/**59.6**	58.7/58.7/64.1	88.3/88.3/**92.4**
qsw-Right	0.9/0.9/6.3	4.1/4.5/**90.5**	13.5/13.5/92.3

Table 10. Alpha-mammal attacks on the CASV4-Th dataset. Here, we discard mixtures with more than 70% of 1's or 0's.

feature-laterality	Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR		
	0.01%	0.1%	1%
lg-Left	4.8/**59.5**/15.6	6.1/**92.6**/23.6	46.5/99.0/92.7
lg-Right	4.6/6.2/51.8	12.0/19.3/76.6	48.0/75.5/98.6
qsw-Left	0.1/1.6/2.2	3.2/5.1/28.1	83.2/80.3/99.1
qsw-Right	0.4/0.9/74.3	2.2/4.8/71.1	71.1/71.1/**99.5**

Similarly, we show the results on the CASV4-Th dataset in Table 10. It is also important to note that, unlike alpha-wolves, we only obtain one alpha-mammal per search process. Thus this attack is always one-shot. Overall, we saw that for the IITD dataset, the XOR operator has the best performance except for the lg feature and the left eye class. In this specific exception, the AND operator outperforms the XOR operation at $FMR(\%) = \{0.1, 1\}$. We saw some interesting mixtures, especially for the AND operator and lg feature on left IrisCodes, where the mixed IrisCode mask ends up covering a majority of the eye region. This may indicate that specific regions in the alpha-mammal IrisCode that are not occluded contribute significantly towards a successful dictionary attack. We show a specific example of such a scenario in Fig. 2. Therefore, after more than three IrisCodes are combined we limit the lateral movement, *i.e.*, unless we see an improvement in the reward function we terminate the search. This reduces the number of IrisCodes that are combined and thus limits the occluded region.

(a) alpha-mammal IrisCode mask

(b) alpha-mammal IrisCode

Figure 2. Example of an alpha-mammal computed on the IITD dataset wherein a majority of the IrisCode is covered by the mask.

6. Discussion

6.1. Analyzing alpha-mixtures

We discuss our findings and offer insights into the behavior of the *alpha-mixtures*. Surprisingly, we observe that the seed IrisCodes that generate the alpha-mixtures

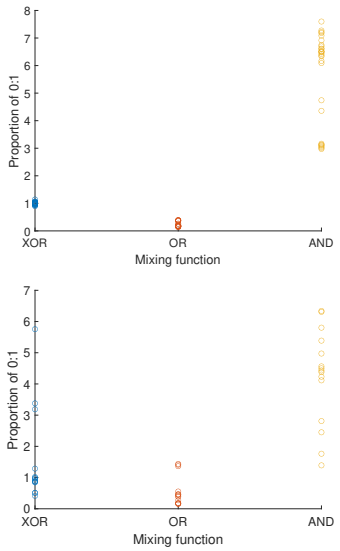


Figure 3. Distribution of 1's and 0's of alpha-wolves using two wolves for IITD lg-Right (left) and CASV4-Th qsw-Right (right).

belong to good quality ocular images with little or no occlusion, thereby alleviating concerns regarding fragile bits [13]. We further observe that increasing the number of wolves/mammals in the mixture does not necessarily increase the chances of higher user coverage. In a majority of cases, mixing two wolves/mammals yields the best user coverage. In terms of mixing operators, XOR appears to produce the highest user coverage in most of the cases. XOR performs logical inequality that produces True/1 only if both bits disagree, implying XOR-mixing inherently combines highly dissimilar wolves, while OR- and AND-mixing combines relatively similar wolves. We use Normalized Compression Distance (NCD) to examine how 'similar' or 'dissimilar' the alpha-wolves (α_W) are compared to the seed wolves (W). NCD computes the similarity between objects by evaluating the binary size of their compressed versions, $C(\cdot)$. $NCD(W, \alpha_W) = \frac{C(W\alpha_W) - min\{C(W), C(\alpha_W)\}}{max\{C(W), C(\alpha_W)\}}$, $0 < NCD < 1$. We observe that NCD is highest for XOR-mixed alpha-wolves (≈ 0.98) compared to AND- and OR-mixed alpha-wolves (≈ 0.85) on the IITD dataset. Alternatively, in the context of information theory, Daugman suggests that there should be no uncertainty about the identity X denoted by a biometric signal Y, *i.e.*, $H(X|Y) = 0$ [9]. However, $Y_{alphamix}$ combines biometric signals from two identities (say, X_1 and X_2), thus increasing the conditional entropy, $H(X_1, X_2|Y_{alphamix}) = H(X_1|Y_{alphamix}) + H(X_2|Y_{alphamix}, X_1) = H(X_2|Y_{alphamix}) + H(X_1|Y_{alphamix}, X_2)$. We speculate that higher conditional entropy in the alpha-mixtures may be responsible for increase in biometric collisions, thereby producing higher false matches.

Table 11. Statistics of bit '1' in the original, seed codes and XOR-mixed alpha-wolves from the IITD and CASV4-Th datasets for IrisCodes and their masks.

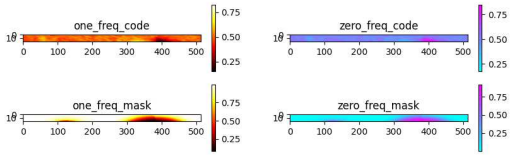
	feature-laterality	IITD code $\mu \pm \sigma$	IITD mask $\mu \pm \sigma$	CASV4-Th code $\mu \pm \sigma$	CASV4-Th mask $\mu \pm \sigma$
original	*lg*-left	0.49±0.03	0.86±0.11	0.5±0.03	0.9±0.16
	lg-right	0.49±0.05	0.85±0.17	0.48±0.05	0.77±0.2
	qsw-left	0.55±0.5	0.83±0.17	0.56±0.07	0.9±0.16
	qsw-right	0.54±0.06	0.84±0.19	0.59±0.09	0.77±0.2
seed codes	*lg*-left	0.49±0.01	0.92±0.03	0.35±0.12	0.36±0.29
	lg-right	0.49±0.01	0.85±0.09	0.47±0.04	0.64±0.24
	qsw-left	0.49±0.005	0.86±0.10	0.45±0.16	0.54±0.31
	qsw-right	0.56±0.03	0.85±0.09	0.68±0.23	0.22±0.18
α_{wolves}	*lg*-left	0.45±0.1	0.06±0.01	0.44±0.09	0.46±0.21
	lg-right	0.47±0.05	0.13±0.07	0.48±0.03	0.31±0.2
	qsw-left	0.49±0.05	0.14±0.09	0.48±0.16	0.4±0.25
	qsw-right	0.46±0.03	0.13±0.07	0.46±0.18	0.31±0.14

We investigate the distribution of 1's and 0's to understand the behavior of the *mixing function*. Fig. 3 analyzes the proportion of 0's to 1's in the alpha-wolves from the IITD right-lg feature and from CASV4-Th right-qsw-based IrisCodes. It is surprising to note that in cases where the XOR mixing function yields the highest coverage (IITD right-lg), the alpha-wolves are tightly clustered around one, indicating equal proportion of 1's and 0's. However, whenever the AND-mixing function produces higher user coverage compared to XOR (CASV4-Th right-qsw), we observe the proportion of 0's to 1's in XOR-ed alpha-wolves to be scattered. We present a statistical analysis (mean and standard deviation) of the bits of the IrisCodes and their masks for the original IrisCodes, seed wolves and alpha-wolves in Table 11. We observe that the XOR-mixed IrisCodes are consistent with the original IrisCodes, but the IrisCode masks become sparser, *i.e.*, contain more 0's. We qualitatively analyze the frequency of '1' and '0' bits in original and alpha-wolf IrisCodes and masks in Fig. 4.

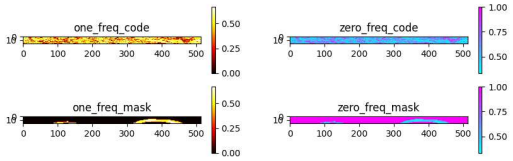
6.2. Additional analysis

Is there an overlap among the seed IrisCodes across feature encoding and eye laterality? We observe that identical subjects (but different samples) appear as wolves for both *lg* and *qsw* within the same laterality (right) on the IITD dataset. For example, *lg* has seed wolves {074_09, 150_06}, while *qsw* has {074_06, 150_09}. XX_YY denotes subject XX and sample YY. We note minimal overlap among the seed wolves on the CASV4-Th dataset.

Can we assess the viability of the mixed IrisCodes at the image level? To examine the viability of the mixed IrisCodes, we use an off-the-shelf image-to-image translation network [14] to accept a binary IrisCode as input and generate the corresponding iris image as output. We use

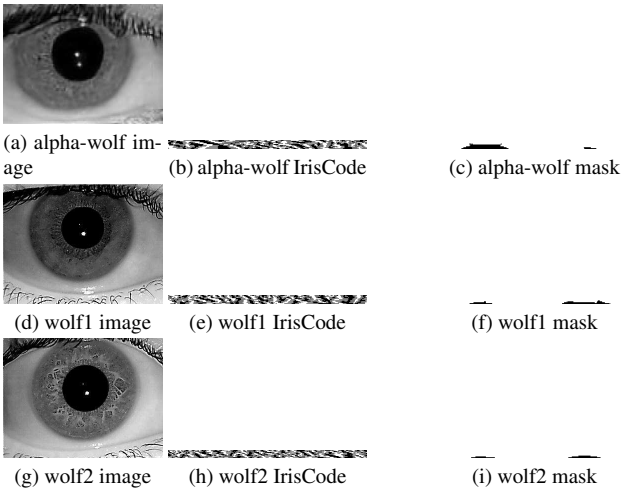


(a) The heatmap of IITD left eyes with *lg* feature.



(b) The heatmap of alpha-wolf of IITD left eyes with *lg* feature.

Figure 4. Examples of the heatmap of '0's and '1's frequency.



(a) alpha-wolf image (b) alpha-wolf IrisCode (c) alpha-wolf mask

(d) wolf1 image (e) wolf1 IrisCode (f) wolf1 mask

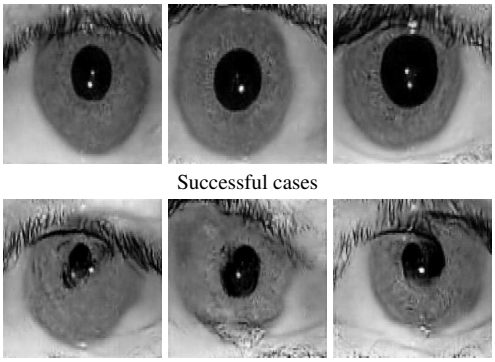
(g) wolf2 image (h) wolf2 IrisCode (i) wolf2 mask

Figure 5. Illustration of alpha-wolf translated iris image (top row) and the respective constituent seed wolves (bottom two rows) along with their IrisCodes and masks.

a simple network, pix2pix [3], as our main objective is to inspect the viability of the mixed IrisCodes as biologically plausible "human" iris pattern. We incorporate the Deep Image Structure and Texture Similarity (DISTS) [10] index as an auxiliary term to the standard GAN expectation and L_1 loss terms in the formulation to account for textural and structural details preservation in the generated image. Therefore, the final generator loss function is as follows.

$$\mathcal{L}_{\mathcal{G}} = \lambda_{GAN} * \mathcal{L}_{\mathcal{GAN}} + \lambda_{L1} * \mathcal{L}_{L1} + \lambda_{DISTS} * \mathcal{L}_{DISTS} \quad (2)$$

In Eqn. 2, $\lambda_{GAN} = 1, \lambda_{L1} = 100, \lambda_{DISTS} = 0.1$. We trained the model using an ADAM optimizer with initial learning rate=0.0002, momentum term=0.5, for 200 epochs and batch size=1. We used the entire dataset of IITD with the original IrisCodes and the ocular images for the translation network. However, we trained separately for each eye category and feature extractor, resulting in four models (left/right $\times$ *lg*/*qsw*). Our test set comprises of the mixed IrisCodes (alpha-wolves) as inputs. See the generated images, codes and masks corresponding to the two wolves in Fig. 5. More examples of generated iris images corresponding to alpha-wolves (mixture of wolves) are presented in Fig. 6. The image translation step filters out improbable alpha-wolves via manual inspection. Next, we recompute the user coverage with the successful alpha-wolves. We report them for XOR-mixed alpha-wolves (best-performing) in Table 12. At FMR=0.001%, we observe an absolute decrease in the user coverage by 1.8% on IITD left-*lg*, while decreasing the number of attack attempts by 16%; a decrease in the user coverage by 2.7% on IITD right-*lg*, while decreasing the number of attack attempts by 39%; no decrease in attacks on IITD left-*qsw*, while decreasing the number of attack attempts by 50%; and finally, a decrease in the user coverage by 13.4% on IITD right-*qsw*, while decreasing the number of attack attempts by 50%.



Successful cases

Failure cases

Figure 6. Examples of outputs from the viability check.

Table 12. User coverage after filtering alpha-wolves using viability check on IITD dataset.

feature-laterality	Proportion of Users (%) Covered @ False Match Rate for XOR			
	0.001%	0.01%	0.1%	1%
lg-left	8.9	16.9	79.0	89.7
lg-right	15.2	17.8	62.5	72.3
qsw-left	20.9	25.0	84.4	93.3
qsw-right	18.3	20.9	83.0	87.1

Can synthetic IrisCodes be used to launch dictionary attacks? In this experiment, we use CASIA-IrisV4-Synthetic dataset [1] that comprises 10,000 images from 1,000 synthetic identities (no left or right eye category). Refer to [6, 17] for synthesis details. We use a subset of 4,000 images (first 4 samples) from the dataset without masks.

Intra-dataset performance: We observe a maximum user coverage of 4.3% with *lg* and 4.0% using *qsw*, both

Table 13. Cross-attacks using alpha-wolves on IITD dataset.

α_W → $\mathcal{D}_{IC}$	Thres-holds	Left Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR				Right Proportion of Users (%) Covered @ False Match Rate for OR/AND/XOR			
		0.001	0.01	0.1	1	0.001	0.01	0.1	1
$lg \rightarrow qsw$	τ_{α_W}	0.0/0.0/10.27	0.0/0.0/18.75	2.23/1.79/89.73	2.23/1.79/92.41	0.0/0.0/16.96	0.0/0.0/20.09	1.79/2.23/87.50	1.79/2.23/89.73
	$\tau_{D_{IC}}$	0.0/0.0/12.5	0.0/0.0/22.70	2.23/1.79/**92.86**	3.57/2.23/**95.09**	0.0/0.0/20.09	0.0/0.0/23.21	2.23/2.68/89.73	4.02/4.02/89.73
$qsw \rightarrow lg$	τ_{α_W}	0.0/0.0/21.43	0.0/0.0/23.66	1.79/2.68/89.29	2.23/4.02/93.30	0.0/0.0/**29.02**	0.0/0.0/**34.82**	2.23/2.68/87.95	2.68/3.57/89.29
	$\tau_{D_{IC}}$	0.0/0.0/20.98	0.0/0.0/22.77	1.79/2.23/78.57	1.79/2.68/89.29	0.0/0.0/26.79	0.0/0.0/29.02	1.79/1.79/78.12	2.23/1.79/87.50

@FMR=0.1% by OR-ing two synthetic IrisCodes.

Cross-dataset performance: On CASV4-Th left dataset, we observe @FMR=0.1%, a maximum user coverage of 1.0% using *lg* by AND-ing two synthetic IrisCodes and 22.7% using *qsw* by OR-ing two IrisCodes. On CASV4-Th right dataset, we observe a maximum user coverage of 2.6% using *lg* by AND-ing two IrisCodes, and 47.7% using *qsw* by OR-ing two IrisCodes. We achieve a maximum user coverage of 16.8% using *lg* by XOR-ing two IrisCodes when tested on *qsw*-based IrisCodes @FMR=1%.

We also adopt a 2-state Hidden Markov Model (HMM) with a transition probability $\alpha = 0.9$ as suggested in [9] to generate 10 synthetic IrisCodes. We mix them using the bitwise operators and then test them on real datasets (IITD and CASV4-Th). We observe AND-mixing achieves best user coverage of 0.4% on the IITD dataset and 12.1% on the CASV4-Th dataset, thus, indicating synthetic IrisCodes can be used as dictionary attacks against real IrisCodes.

Are the attacks effective assuming limited knowledge? Previous experiments consider that the adversary has full knowledge of the feature encoding scheme and their corresponding decision thresholds. In this experiment, we use lg-based alpha-wolves, α_W (from 2 wolves) to launch dictionary attacks against qsw-based IrisCodes, $\mathcal{D}_{IC}$, and vice-versa. We compute the user coverage considering multiple thresholds, τ_{α_W} (threshold corresponding to the feature of the alpha-wolves) and $\tau_{D_{IC}}$ (threshold corresponding to the feature of the target IrisCode). This experiment assumes limited knowledge on part of the adversary about the encoding employed by the target system. Results in Table 13 indicate that even with partial knowledge, template level attacks achieve an extremely high coverage of 29.02% @FMR=0.001% when XOR-mixed qsw-based wolves are used against lg-based IrisCodes on the IITD right dataset.

Can we learn the *best* possible way to combine IrisCodes? We have a fixed set of logical operators as the mixing function but that does not guarantee optimal mixing. Therefore, we employ an existing image fusion technique, namely IFCNN [32] to perform mixing. IFCNN uses a 4-layer CNN trained on over 100K RGB and depth-images for fusing multi-modal, multi-spectral and multi-exposure images using MSE and perceptual losses extracted from a pre-trained ResNet 101 model. We selected this network as it allows fusion of variable number of inputs. We supply seed wolves as templates in one setup and as iris images in another setup to perform mixing at both template and image level. The best user coverage from the fused wolf IrisCodes is 21.9%, and from the fused wolf iris images is 17.4% @FMR=1% after fusing 2 codes/images.

Summary: We study the feasibility of dictionary attacks on iris recognition systems for the first time. Although the practicality of this attack is currently restricted at the template (IrisCode) level, we observe vulnerabilities on high quality iris datasets and we suspect that the risk might be further compounded in the presence of non-ideal imaging (low resolution, inadequate illumination, etc.). Our findings surprisingly indicate that mixing IrisCodes using simple bitwise operators can be highly effective as dictionary attacks against a large number of unseen identities. We observe that, in particular, XOR-operator increases the user coverage, *e.g.*, from 1.34% on IITD left-lg wolves to 20.5% @FMR=0.01% with XOR-mixed alpha-wolves (only wolf samples) and 51.6% @FMR=0.1% with XOR-mixed alpha-mammals (with or w/o wolves). Even synthetic IrisCodes can be used as alpha-wolves to launch dictionary attacks on real datasets with 47.7% coverage @FMR=0.1%. We further validate the viability of alpha-mixtures at image level via a conditional generative network.

7. Conclusion

In this work, we explore dictionary attacks at the template level (IrisCodes) on iris recognition systems that use log Gabor (*lg*) and spatial Gabor (*qsw*) features. We show that mixing IrisCodes using AND, OR and XOR operators results in the so-called Master IrisCodes that can fortuitously match with a large number of other identities. The IrisCodes are strategically selected: they can be either wolves, resulting in alpha-wolves, or selected via search optimization, resulting in alpha-mammals. We empirically analyze the efficacy of these attacks on three datasets, *viz.*, IITD, CASIA-IrisV4-Thousand and Synthetic, and achieve a user coverage of upto 71 identities @FMR=0.001% using real alpha-wolves, upto 133 identities using alpha-mammals at FMR=0.1% and upto 477 identities @FMR=0.1% using synthetic alpha-wolves. Our method is effective on cross-attacks across different IrisCode encoding schemes. Future work will extend to image-level dictionary attacks.

References

[1] CASIA Iris V4 Database. https://hycasia.github.io/dataset/casia-irisv4/. [Online accessed May 2023]. 3, 7

[2] IIT Delhi Iris Database. https://www4.comp.polyu.edu.hk/~csajaykr/IITD/Database_Iris.htm. [Online accessed Jan 2023]. 3

[3] pix2pix tensorflow implementation. https://github.com/affinelayer/pix2pix-tensorflow. [Online accessed June 2023]. 7

[4] UIDAI AADHAR Program. https://uidai.gov.in/en/ecosystem/authentication-devices-documents/biometric-devices.html. [Online accessed August 2023]. 1

[5] Philip Bontrager, Aditi Roy, Julian Togelius, Nasir Memon, and Arun Ross. Deepmasterprints: Generating masterprints for dictionary attacks via latent variable evolution. In *IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–9, 2018. 1

[6] Jiali Cui, Tieniu Tan, and Zhenan Sun. An iris image synthesis method based on pca and super-resolution. volume 4, pages 471– 474, 09 2004. 7

[7] John Daugman. How iris recognition works. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004. 2

[8] John Daugman. *Iris Recognition at Airports and Border-Crossings*, pages 819–825. Springer US, Boston, MA, 2009. 1

[9] John Daugman. Information theory and the iriscode. *IEEE Transactions on Information Forensics and Security*, 11(2):400–409, 2016. 2, 6, 8

[10] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022. 7

[11] George Doddington, Walter Liggett, Alvin Martin, Mark Przybocki, and Douglas Reynolds. SHEEP, GOATS, LAMBS and WOLVES A Statistical Analysis of Speaker Performance in the NIST 1998 Speaker Recognition Evaluation. *5th International Conference on Spoken Language Processing (ICSLP)*, 1998. 1, 2

[12] Gizem Erdogan. Contact Lenses in Iris Recognition - MS Thesis. Technical report, West Virginia University, 2013. 2

[13] Karen P. Hollingsworth, Kevin W. Bowyer, and Patrick J. Flynn. The best bits in an iris code. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):964–973, 2009. 6

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-Image Translation with Conditional Adversarial Networks. *Proc. of Computer Vision and Pattern Recongition*, 2017. 6

[15] Anil K. Jain, Arun A. Ross, and Karthik Nandakumar. *Introduction to Biometrics*. Springer Publishing Company, Incorporated, 2011. 2

[16] Jong-Gook Ko, Youn-Hee Gil, Jang-Hee Yoo, and Kyo Chung. A novel and efficient feature extraction method for iris recognition. *ETRI Journal*, 29:399–401, 06 2007. 2

[17] Li Ma, Tieniu Tan, Yunhong Wang, and Dexin Zhang. Personal identification based on iris texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1519–1533, 2003. 2, 7

[18] Mirko Marras, Paweł Korus, Anubhav Jain, and Nasir Memon. Dictionary attacks on speaker verification. *IEEE Transactions on Information Forensics and Security*, 18:773–788, 2023. 1

[19] Libor Masek. Recognition of human iris patterns for biometric identification - m.s. thesis report. Technical report, School of Computer Science and Software Engineering,University of Western Australia, 2003. 2

[20] Donald M. Monro, Soumyadip Rakshit, and Dexin Zhang. Dct-based iris recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):586–595, 2007. 2

[21] Kien Nguyen, Hugo Proença, and Fernando Alonso-Fernandez. Deep learning for iris recognition: A survey. *arXiv*, 10 2022. 2

[22] George W. Quinn, James Matey, Elham Tabassi, and Patrick J. Grother. IREX V: Guidance for Iris Image Collection. Technical report, NIST Interagency/Internal Report (NISTIR) - 8013, 2014. 1

[23] Christian Rathgeb and Christopher Busch. On the feasibility of creating morphed iris-codes. In *IEEE International Joint Conference on Biometrics (IJCB)*, pages 152–157, 2017. 2, 3

[24] Christian Rathgeb and Andreas Uhl. Secure iris recognition based on local intensity variations. In Aurélio Campilho and Mohamed Kamel, editors, *Image Analysis and Recognition*, pages 266–275, 2010. 2

[25] Christian Rathgeb, Andreas Uhl, Peter Wild, and Heinz Hofbauer. Design decisions for an iris recognition sdk. In Kevin Bowyer and Mark J. Burge, editors, *Handbook of Iris Recognition*, Advances in Computer Vision and Pattern Recognition. Springer, second edition edition, 2016. 2

[26] Aditi Roy, Nasir Memon, and Arun Ross. Masterprint: Exploring the vulnerability of partial fingerprint-based authentication systems. *IEEE Transactions on Information Forensics and Security*, 12(9):2013–2025, 2017. 1

[27] Renu Sharma and Arun Ross. Image-level iris morph attack. In *IEEE International Conference on Image Processing (ICIP)*, pages 3013–3017, 2021. 2

[28] Ron Shmelkin, Tomer Friedlander, and Lior Wolf. Generating master faces for dictionary attacks with a network-assisted latent space evolution. *16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pages 01–08, 2021. 1

[29] Philipp Terhorst, Florian Bierbaum, Marco Huber, Naser Damer, Florian Kirchbuchner, Kiran B. Raja, and Arjan Kuijper. On the (limited) generalization of masterface attacks and its relation to the capacity of face representations. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–9, 2022. 1

[30] Masashi Une, Akira Otsuka, and Hideki Imai. Wolf attack probability: A theoretical security measure in biometric authentication systems. *IEICE Transactions on Information and Systems*, E91D, 05 2008. 1

[31] Neil Yager and Ted Dunstone. The biometric menagerie. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):220–230, 2010. 1

[32] Yu Zhang, Yu Liu, Peng Sun, Han Yan, Xiaolin Zhao, and Li Zhang. IFCNN: A General Image Fusion Framework Based on Convolutional Neural Network. *Information Fusion*, 54:99–118, 2020. 8