IRIS RECOGNITION: ENHANCING SECURITY AND IMPROVING PERFORMANCE

By

Renu Sharma

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science – Doctor of Philosophy

2022

ABSTRACT

IRIS RECOGNITION: ENHANCING SECURITY AND IMPROVING PERFORMANCE

By

Renu Sharma

Biometric systems recognize individuals based on their physical or behavioral traits, viz., face, iris, and voice. Iris (the colored annular region around the pupil) is one of the most popular biometric traits due to its uniqueness, accuracy, and stability. However, its widespread usage raises security concerns against various adversarial attacks. Another challenge is to match iris images with other compatible biometric modalities (i.e., face) to increase the scope of human identification. Therefore, the focus of this thesis is two-fold: firstly, enhance the security of the iris recognition system by detecting adversarial attacks, and secondly, accentuate its performance in iris-face matching.

To enhance the security of the iris biometric system, we work over two types of adversarial attacks - presentation and morph attacks. A presentation attack (PA) occurs when an adversary presents a fake or altered biometric sample (plastic eye, cosmetic contact lens, etc.) to a biometric system to obfuscate their own identity or impersonate another identity. We propose three deep learning-based iris PA detection frameworks corresponding to three different imaging modalities, namely NIR spectrum, visible spectrum, and Optical Coherence Tomography (OCT) imaging inputting a NIR image, visible-spectrum video, and cross-sectional OCT image, respectively. The techniques perform effectively to detect known iris PAs as well as generalize well across unseen attacks, unseen sensors, and multiple datasets. We also presented the explainability and interpretability of the results from the techniques. Our other focuses are robustness analysis and continuous update (retraining) of the trained iris PA detection models. Another burgeoning security threat to biometric systems is morph attacks. A morph attack entails the generation of an image (morphed image) that embodies multiple different identities. Typically, a biometric image is associated with a single identity. In this work, we first demonstrate the vulnerability of iris recognition techniques to morph attacks and then develop techniques to detect the morphed iris

images.

The second focus of the thesis is to improve the performance of a cross-modal system where iris images are matched against face images. Cross-modality matching involves various challenges, such as cross-spectral, cross-resolution, cross-pose, and cross-temporal. To address these challenges, we extract common features present in both images using a multi-channel convolutional network and also generate synthetic data to augment insufficient training data using a dual-variational autoencoder framework. The two focus areas of this thesis improve the acceptance and widespread usage of the iris biometric system.

Dedicated to Mummy, Daddy, Maa and Bapa

ACKNOWLEDGMENTS

The journey of my Ph.D. research wouldn't be possible without the support of a number of people, and let me take this opportunity to thank them. First, I would like to bestow my sincere thanks and gratitude to my Ph.D. advisor, Dr. Arun Ross, whose acceptance email began this journey. He motivates me to imbibe various research qualities—from critical and deep thinking to efficient communication. His guidance, feedback, support, and motivation give directions to my Ph.D. research at every stage. He also provides me an opportunity to exhibit my research outside the lab through commercial projects, summer school, conferences, journals, and workshops. Next, I would like to thank my doctorate committee—Dr. Xiaoming Liu, Dr. Vishnu Boddeti, and Dr. Selin Aviyente—for their continuous support and valuable feedback.

I want to convey my special thanks to all the CSE faculty members, especially my course instructors (Dr. Eric Torng, Dr. Xiaoming Liu, Dr. Jiayu Zhou, Dr. Arun Ross, Dr. Vishnu Boddeti, Dr. Sandeep Kulkarni, and Dr. Ashoke Sinha) for enhancing my knowledge in various subjects. I am also thankful to all the CSE and MSU administrative staff for helping me out with administrative affairs, want to mention Brenda Hodge, Steve Smith, Amy King, Erin Dunlop, and Vincent Mattison.

I am fortunate to have wonderful labmates who made my Ph.D. an enjoyable journey, whether there was a technical hurdle or a research plateau. Their eagerness to help and reliable persona made everything easier for me. I loved those planned outings and sudden lunches with them. Thank you all-Thomas, Sudipta, Anurag, Melissa, Steven, Denny, Yaohui, Aaron, Vahid, Achsah, Ishita, Shivangi, Darshika, Cunjian, Austin, Ryan, Parisa, Sushanta, Debasmita, Raul, Morgan, Redwan, Pegah, Katie, Sai, Madison, Protichi, Ana.

On the personal front, how could I express my gratitude through words to my parents (mummy and daddy)? Their unconditional love and belief hold me at every stage of my life. I am also grateful to my parents-in-law (maa and bapa), whose proud eyes push my limits further. The inspiration and love from my sister and brother keep this journey going. I am also thankful to my sisters-in-laws,

brothers-in-laws, and cheery kids for sprinkling various colors in my life.

I am thankful to the second family I had here in Michigan as my friends—Apoorva, Sneha, Gauri, Aditya, Affan, Shalin, and Tanvi. I cherish the numerous unforgettable memories we made together. Thanks for providing me with a helping hand whenever required.

And lastly, I would like to thank my better half, Sushanta, for always being with me. His strong belief, support, and love made this thesis possible. Thanks to my kiddo, Siddhant, for having such an infectious smile :).

I am blessed to be starting my Ph.D. journey in the beautiful green and white landscape of the MSU campus.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Biometrics

Human authentication is very much required in our day-to-day activities, for instance, log in to a laptop, computer, or mobile; access control to a building; attendance system; ATM transactions, credit card payment; or border crossing through airports. Traditional ways of human authentication - knowledge-based (personal identification number (PIN) or secret pattern) and token-based (card)- are not able to keep pace with the increasing demands of authentication. The traditional ways require remembering a large number of passwords or carrying a lot many cards or keys, which is inconvenient for users and also limits the security. Biometrics, which refers to the measurement and calculation of body characteristics, meet the high demands of authentication requirement as users need not remember the password or carry the token. It recognizes humans based on their physical (face, fingerprint, iris), behavioral (signature, gait), and psychophysiological (ECG, EEG) traits [126]. These traits should be unique, universal, permanent, and measurable. Iris, the annular region around the pupil, is one of the most popular biometric traits due to its high accuracy, fast matching, and great stability.

1.2 Anatomy of Iris

Iris is an anterior part of the uveal tract covered by the cornea from the front and supported by the lens from the back [40]. It separates the anterior and posterior chambers of the eye. At its base, it is connected to the eye's ciliary body and another end leads to the pupil. The frontal view of the iris is a colored annular region surrounding the pupil of an eye and surrounded by the sclera of an eye (Figure 1.1 (a)). The diameter of the iris is approximately 12 mm and the circumference is 38mm. The frontal iris pattern ((Figure 1.1 (b)) consists of two zones - pupillary and ciliary zone - separated by collarette. The pupillary zone (Figure 1.1 (c)) lies in the proximity

of the pupil and contains sphincter muscles, connecting crests, and pigment or pupillary ruff. The sphincter muscles encircle the pupil and controlled by parasympathetic nerve endings. It constricts the pupil in the presence of high illumination (miosis). The ciliary zone (Figure 1.1 (c)) consists of crypts, contraction furrows, dilator muscles. The dilator muscle fibers run radially and control by sympathetic nerve ending. It dilates the pupil in low illumination (mydriasis). Figure 1.2 shows the frontal view of the iris captured by different imaging techniques.

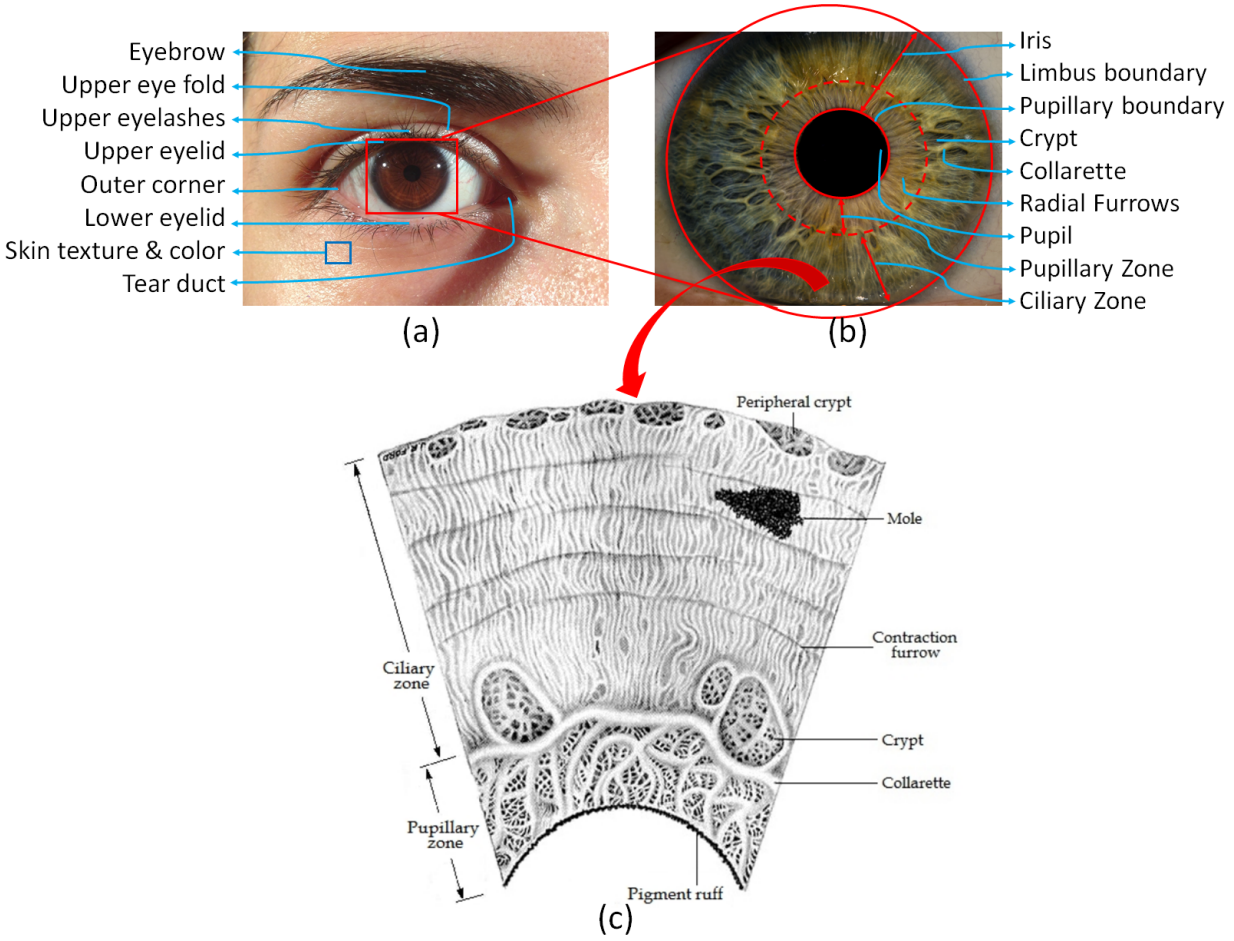


Figure 1.1: Frontal view of the iris: (a) ocular image, (b) focused view of the iris pattern, and (c) frontal anatomy of the iris [40].

The cross-sectional view of captured using Optical Coherence Tomography (OCT) imaging is shown in Figure 1.3 (a). The arc-like structure is the cornea, whereas the cloud-like structure corresponds to the iris tissue structure. Figure 1.3 (b) shows iris cross-sectional anatomical view. It consists of four layers from anterior to posterior: (a) anterior border layer, (b) stroma layer,

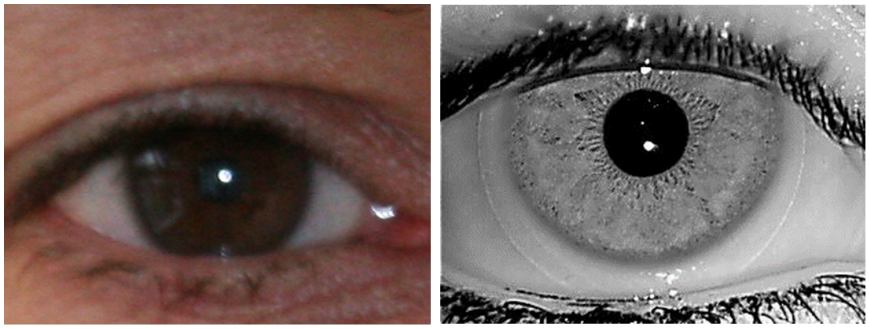


Figure 1.2: Iris images captured using different imaging techniques: (a) captured in the visible spectrum illumination, and (b) captured in the near-infrared spectrum illumination.

(c) anterior epithelium layer, and (d) posterior pigmented epithelium layer. The anterior border layer contains interconnected connective tissues (fibroblasts) and beneath them, pigment cells (melanocytes) derived from the anterior stroma. The border layer is absent in the crypts and contraction furrows. It is thickest in the pupillary zone and at the periphery of the ciliary zone. The stroma layer consists of a loose collagenous network of sphincter pupillae muscles, blood vessels, nerves, fibroblasts, melanocytes, clump cells, and mast cells. Melanocytes present in the anterior border and stroma layer mainly contribute to the color of the iris. Dark-colored iris (brown, black) are profuse in melanocytes, whereas light-colored iris (blue, green, or gray) are sparse in melanocytes. The anterior epithelium layer is about 12.4 μm in thickness and mainly consists of dilator pupillae muscle. The last posterior pigment epithelium is a layer of cells derived from the internal layer of the optic cup. It mainly consists of pigmented cells.

1.3 Automated Iris Recognition System

The complex structure of the iris results in randomness in the iris pattern, which in turn supports the uniqueness at the planetary scale of the global human population [68]. Iris patterns of left and right eyes of an individual and iris patterns of monozygotic twins are also not similar. Due to its uniqueness and universality, it is considered as a suitable and reliable biometric trait employed for authentication.

Biometric authentication occurs in two stages — enrollment and recognition [126]. During

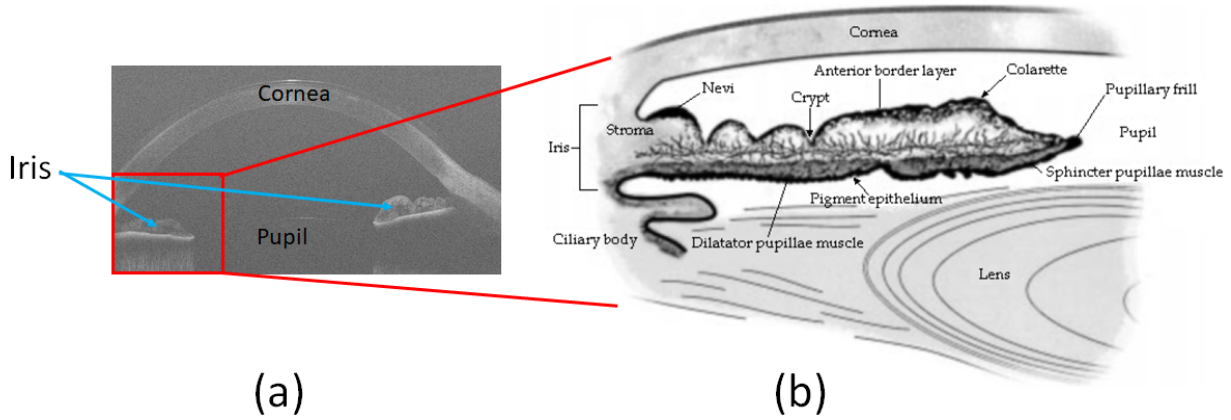


Figure 1.3: Cross-sectional view of the iris: (a) image captured from optical coherence tomography imaging and (b) transverse anatomy of the iris [295].

the enrollment stage, an individual presents their biometric sample (iris) to the acquisition unit. The sample is then transformed into a biometric template which gets stored in the enrollment database (also known as reference or gallery set). Each enrolled template is associated with a unique identifier. During the recognition stage, an individual again presents their biometric sample for authentication, which gets transformed into a biometric template. The query template is then matched against enrolled templates either in identification or verification mode. In identification, the query template is matched against all enrolled templates (one-to-many matching) and returns an identifier associated with it. In verification, the query template is input along with a claimed identity. Hence, the query template is matched against the claimed identity enrolled template (one-to-one matching) and returns the boolean value (verified or not verified). The brief of iris recognition pipeline (Figure 1.4) is provided in the following steps:

- **Acquisition:** The acquisition module acquired an iris image from an individual using a specialized iris sensor. Most commercial iris sensors capture iris images in near-infrared (NIR) illumination range (700–900 nm), though most smartphones capture in visible spectrum (VIS) range (400-700 nm). Different spectral bands can potentially be used to capture different components of the iris. NIR illumination predominantly captures the stromal features (fibrovascular layer) of the iris, whereas VIS captures information about the pigment melanin.

- **Iris Segmentation:** The iris segmentation module locates the iris region from the acquired image. It involves the detection of pupillary (the inner boundary between the pupil and the iris) and limbic (the outer boundary between the iris and the sclera) boundaries. However, researchers also include detection of occlusions to the iris region, such as specular reflections, upper and lower eyelashes, upper and lower eyelids. Iris segmentation techniques in the literature can be categorized as edge-based, region growing-based, active contour-based, and learning-based (machine learning and deep learning). The edge-based approaches first detect edge points in an iris image and then fit circular or elliptical models to detect pupillary and limbic boundaries. Most popular edge-based methods used for segmenting iris region are integrodifferential operators [69], Hough transforms [295]. Tan et al. [264] proposed a combination method of region clustering, semantic refinements, and integrodifferential operators. These techniques assume the circular inner and outer boundaries for the iris region. Other researchers fitted elliptical models [80, 158, 183, 196, 236, 319]. The region growing approaches detect various regions in the entire image and perform semantic refinements to obtain the iris region. Various works that fall under this category are [8, 87, 131, 307, 316]. Later, the boundaries shape assumption is relaxed in active contour-based approaches by fitting irregular boundaries. Various research works based on active contour to detect iris boundaries are [24, 31, 67, 127, 134, 147, 187, 247, 247, 282]. The last learning-based approaches of segmentation aim to classify image pixels into iris and non-iris categories using machine learning techniques, such as SVM classifier [231, 263], AdaBoost [157], triplet Markov fields (TMF) [27], fast-structured random forest [103] and graph-cuts [209]. With the success of deep learning techniques in other computer vision tasks, it has also been utilized in iris segmentation [19, 104, 112, 160, 238, 275].

  Researchers also focus on segmenting the iris region in visible spectrum images [20, 207]. NICE I [4] presented a competition of iris segmentation techniques for the visible spectrum images. A detailed description of iris segmentation techniques are included in [35, 130, 218].

- **Iris Normalization:** After the iris segmentation, the circular iris region is mapped to a

fixed dimension region by the iris normalization module. Typically, the rubber sheet model proposed by Daugman [62] is used for iris normalization. It maps the segmented circular iris region defined in cartesian coordinates (x, y) to rectangular polar coordinates (r, θ). It helps in minimizing variations in the area of the iris region due to the dilation and contraction of the pupil.

- **Feature Extraction:** The feature extraction module is responsible for extracting salient or discriminative features from the normalized or unnormalized iris images and represents the images in a compressed form (template), commonly known as IrisCode. Daugman [69] utilized phase information from normalized iris using quadrature 2D-Gabor wavelets to create a feature template of 2,048 phase bits. Iris feature extraction techniques defined in the literature can be categorized as texture filtering approaches, texture analysis approaches, patch-based approaches, sparse coding, and deep learning representation techniques. Texture filtering approaches extract iris texture using standard texture filters, such as Gabor filters [12, 62, 69, 173], wavelet transform [30, 32, 184, 283], ordinal features, [257], discrete cosine transform (DCT) [179], local intensity variation [151, 169], phase correlation approach [177, 320], Zernike moments [261], and binarized statistical image features (BSIF) [59, 226]. Texture analysis approaches explore underlying iris texture representation using different methods, such as gray-level co-occurrence matrices (GLCM) [77], local-global graph methodology [310], probabilistic graphical models [139], SURF features [175], dynamic programming [210], SIFT features [15, 200, 258], geometric key-based iris encoding [262], bayesian estimation [268, 291], and shape-based features [48]. However, the performance of these methods is not showing much improvement over the traditional texture filtering approaches. Another category of feature extraction techniques is patch-based. The patch-based approaches tessellate an iris image into smaller patches, extract features from patches, and combine them. The research works that fall under this category are [25, 162, 179, 205]. The patch-based encoding helps in handling occlusion and non-linear deformations. The sparse coding-based techniques are useful in handling iris images cap-

tured in non-ideal conditions (low resolution, blur and defocus) [148,149,202,317]. Recently, deep learning-based techniques are applied to extract discriminative features from the iris images [38,91,92,164,189,208]. A detailed description of iris feature extraction techniques is included in [35,37,188].

- **Comparator:** The comparator module performs matching of two iris templates (query template and enrolled template) to establish an identity of an individual and outputs a match score. The match score could be a similarity measure or dissimilarity measure. However, in iris biometrics, hamming distance [62] is generally used to calculate the match score. It is an XOR operation between the two IrisCodes after masking the non-iris bits (eyelids or eyelashes). Hamming distance of two different irides should be equal to 0.5, whereas the hamming distance of two IrisCodes from the same iris should be 0.

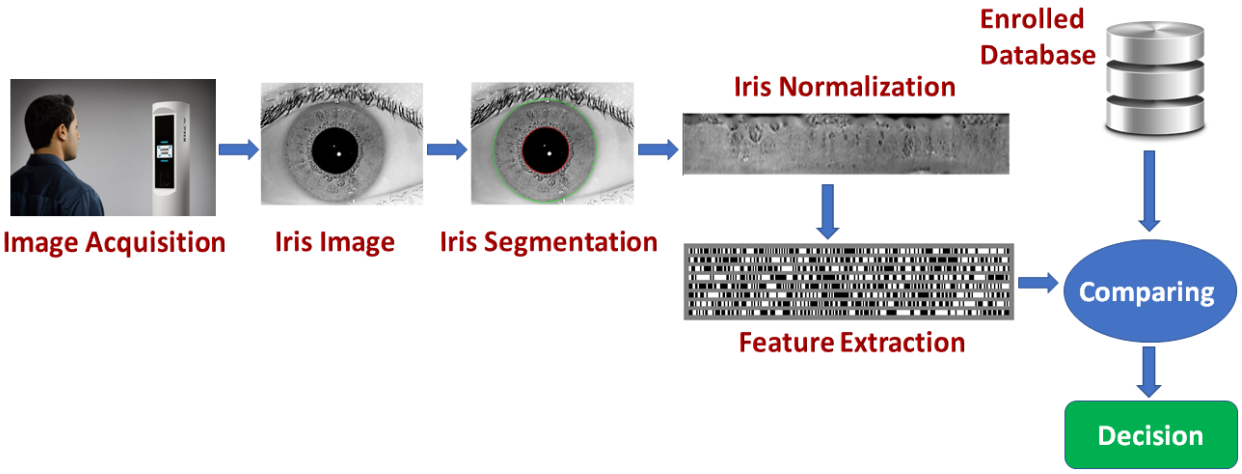


Figure 1.4: Various steps of the automated iris recognition process. It consists of the acquisition of an iris image from an individual, segmentation of iris region, iris region normalization, extraction of features from the iris image, and then the matching of the iris template against the enrolled templates.

1.3.1 Applications

Iris recognition systems have been deployed in a wide range of applications around the world. Few prominent applications of iris recognition are listed below:

- The United Arab Emirates implemented an iris biometric system for border control.[1] Entry through any means land, air, or seaports is now through iris biometrics. There are 1.2 million records in the database and 14 billion matches per day.

- The Amsterdam Schiphol Airport, Netherlands, also implemented iris recognition to expedited, passport-free border security checks since 2001.[2]

- The Hashemite Kingdom of Jordan deployed an iris-enabled automated teller machine (ATM) at Cairo Amman Bank in 2009.[3] In June 2012, UNHCR registered Syrian refugees in Jordan on Cairo Amman Bank ATMs.[4] The system is used to provide financial assistance to refugees.

- India's Aadhaar project is the world's largest biometrics-based identification system with more than 1.28 billion enrollments (April 2021) [6]. The project is to assign a 12-digit unique identification (UID) number to all the residents of India. The UID number is associated with an individual's demographic and biometric information such as a photograph, ten fingerprints, and two iris scans. Various services are also linked with the UID number: electronic-Know Your Client (e-KYC) service, government subsidies distribution, telecom services, income tax services, and financial services.

- Biometric e-passport has been endorsed by 120 countries (since June 2017) [1]. It supports facial, fingerprint, and iris recognition.

- Various mobile platforms also implemented iris recognition for locking and unlocking the mobile devices[5]: Samsung Galaxy S8 and S9 series, Microsoft 950 XL, Fujitsu NX F-04G, Vivo X5Pro, ZTE Grand S3, Alcatel Idol 3, and UMI Iron.

[1] https://www.cl.cam.ac.uk/ jgd1000/UAEdeployment.pdf
[2] https://www.schiphol.nl/en/privium/how-the-iris-scan-works/
[3] https://www.cab.jo/services/IRIS%20Recognition
[4] https://www.unhcr.org/innovation/using-biometrics-bring-assistance-refugees-jordan/
[5] https://webcusp.com/list-of-all-eye-scanner-iris-retina-recognition-smartphones/

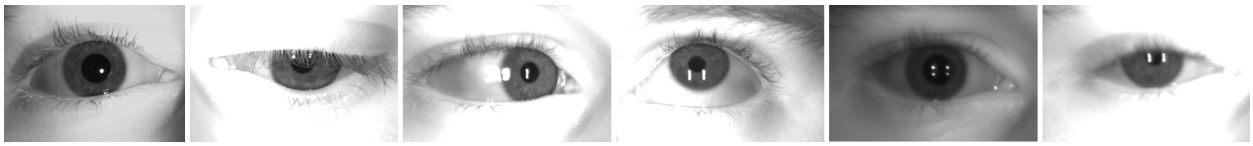
(a) (b) (c) (d) (e)

Figure 1.5: Various applications of iris recognition system: (a) UAE border control system, (b) India's national ID project, (c) Hashemite Kingdom of Jordan's iris-enabled ATM, (d) Biometric e-passport, and (e) Mobile access control.

1.3.2 Challenges

Despite the wide deployment of iris recognition systems, there are various open challenges in iris recognition systems as listed below:

1. **Low-quality Images:** Iris recognition performs considerably well when iris images are captured in a controlled environment. However, when the iris images are captured in non-ideal conditions, it negatively impacts the performance of recognition. Non-ideal scenarios result in degradation of image quality in various ways, such as occlusion by eyelids, occlusion by eyelashes, specular reflections, motion blur, off-angle gaze, low resolution, illumination variations, eyeglasses, or contact lenses. Figure 1.6 shows a good quality iris image along with few low-quality iris images. Various works showed the impact of low-quality images on the performance and also proposed mitigation measures [21, 36, 128, 159, 209, 235, 237, 243, 271]. The NIST IREX II-IQCE report [260] evaluated various iris image quality assessment algorithms.



Good Quality Iris Image | Occlusion + Saturation | Off-angle | Gaze up | Low illumination + Blur | Saturation + Blur

Figure 1.6: Samples of a good quality iris image and few low-quality iris images.

2. **Pupil Dilation:** Generally, pupil diameter lies in the range of 1.5 mm to 8 mm and can be dilated to over 9 mm. Visual features also get affected on dilation, pupillary ruff become

thinner or even disappear, crypts become oblique, vessels become more tortuous, and the contraction and peripheral furrows deepen [40]. Pupil dilation occurs due to light intensity change, alcohol consumption, drug usage, age, disease, eye drops, a person's emotional state, and perceptual events. Several studies [18, 36, 100, 115, 270] show the negative impact of pupil dilation on iris recognition.

3. **Iris Aging:** The resting pupil size decreases with age due to fibrotic changes in the sphincter and atrophy of the dilator muscles [40]. However, iris recognition is considered relatively stable despite these changes [99, 177, 179, 268]. On the contrary, in [36, 84, 269], the authors observed the decrease in iris recognition performance. Later, the works by [29, 111, 239, 273, 294] analyzed various other covariates along with time-lapse responsible for the degradation of iris recognition performance, such as sensor aging (commonly denoted as pixel defects), segmentation errors, quality measures (blur, illumination variation, noise, occlusion), and geometrical factors (pupil dilation). The maximum time span used in the experiments to validate the influence of aging on iris recognition is nine years. Therefore, there is a need to perform experiments on a larger dataset with a longer time span to validate the effect of aging on automated iris recognition.

4. **Iris Diseases:** Ophthalmic disorders not only degrade the iris recognition performance but also increase the failure to enroll rate. These disorders may include cataract, glaucoma, iridis, rubeosis iridis, acute anterior uveitis, aniridia, ciliary body leiomyoma, lisch nodules, iris melanoma, heterochromia synechiae, hyphema, iris cysts, iris prolapse, corneal pathologies, and iridodialysis. American Academy of Ophthalmology[6] reported 24.4 million cataract patients, 7.7 million diabetic retinopathy patients, 2.7 million glaucoma patients, and more than 2.1 million age-related macular patients. There occur more than 7.63 million cataract surgeries (aged 50+ years) in India [101] in the year 2020. Various research works focus on the impact of these disorders on iris recognition in [28, 142, 190, 274].

[6]https://www.aao.org/newsroom/eye-health-statistics

5. **Sensor Interoperability:** The challenge of sensor interoperability arises with the use of different iris sensors where users enrolled using one model of iris sensor and probe images are acquired using another new iris sensor. The situation of cross-sensor matching also arises when the iris image of one application (e.g., national ID) matches with iris images of another application (e.g., law enforcement) where capturing sensors are different. Bowyer et al. [36] showed higher false non-matches when images of different sensors are matched compared with the images of the same sensor. The cross-sensor matching accounts for the differences in the relative position of illumination sources and human eyes, camera characteristics, and spectrums (visible or near-infrared). Various research works focus on cross-sensor matching in [91, 186, 201].

6. **Security:** In [90], researchers reconstructed iris images from iris templates and used them to attack commercial iris recognition systems, with a success rate of around 80%. Samsung's new Galaxy S8 smartphone has also been defeated by German hackers using dummy eyes. In another case, eye drops were used to cause excessive mydriasis and bypass the iris recognition-based border control in UAE. Biometrics considers an integral part of the human body, so the compromise in the individual biometric template compromises his/her identity. Therefore, it is essential to maintain the integrity of the biometric system and protect it from various types of attacks. Section 1.4 provides details of different attacks employed on biometric systems.

7. **Cross-modal Matching:** Cross-modal matching associates data of one biometric modality to another modality. It is required when the legacy database or enrolled template of query identity is not available. It is also beneficial when we need to map genomic data to phenotypic traits [163]. However, the matching involves various challenges due to the difference in modalities, sensors, spectrums, and resolutions. Matching of iris images with face images is discussed in Section 1.5.he challenge of sensor interoperability arises with the use of different iris sensors where users enrolled using one model of iris sensor and probe images are acquired using another new iris sensor.

The focus of this thesis is on the last two (security and cross-modal matching) challenges of the iris recognition system.

1.4 Security of Iris Recognition System

In [3, 224], authors identified nine points of attack in the generic biometric system (Figure 1.7): (1) at the acquisition unit (presentation of a plastic eye to the iris sensor); (2) at the communication channel between the acquisition channel and feature extraction unit (modification of captured biometric sample); (3) at the feature extraction unit (trojan horse attack on feature extractor); (4) at the communication channel between feature extraction unit and comparison unit (attack on TCP/IP protocol and alteration of feature template); (5) at the data storage unit (tampering of enrolled templates); (6) at the communication channel between the data storage unit and comparison unit (attack on TCP/IP protocol and alteration of an enrolled template); (7) at the comparison unit (modification of matching algorithm or match score); (8) at the communication channel between comparison unit and decision unit (attack on TCP/IP protocol and modification of match score); (9) at the decision unit (change or flip of the decision). Cryptography or encryption techniques could prevent attacks at the communication channels (4, 6, 8 attack points). The attacks on the feature extraction (3 attack point), data storage (5 attack point), and comparison (7 attack point) units can be mitigated by keeping these units at a secure location. The attacks before feature extraction (1, 2 attack points) can be resolved by the implementation of additional modules for the detection of fake presentation or modification of a biometric sample. The focus of the thesis is on the attacks employed at attack points 1 (sensor-level) and 2 (image-level) shown in orange boxes of Figure 1.7. These attacks are easier to deploy as they do not require any internal knowledge of the system. The two significant attacks fall in these categories are *presentation attacks* and *morph attacks*.

1.4.1 Iris Presentation Attacks

According to ISO/IEC 30107-1:2016 [3], a Presentation Attack (PA) is a "presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric

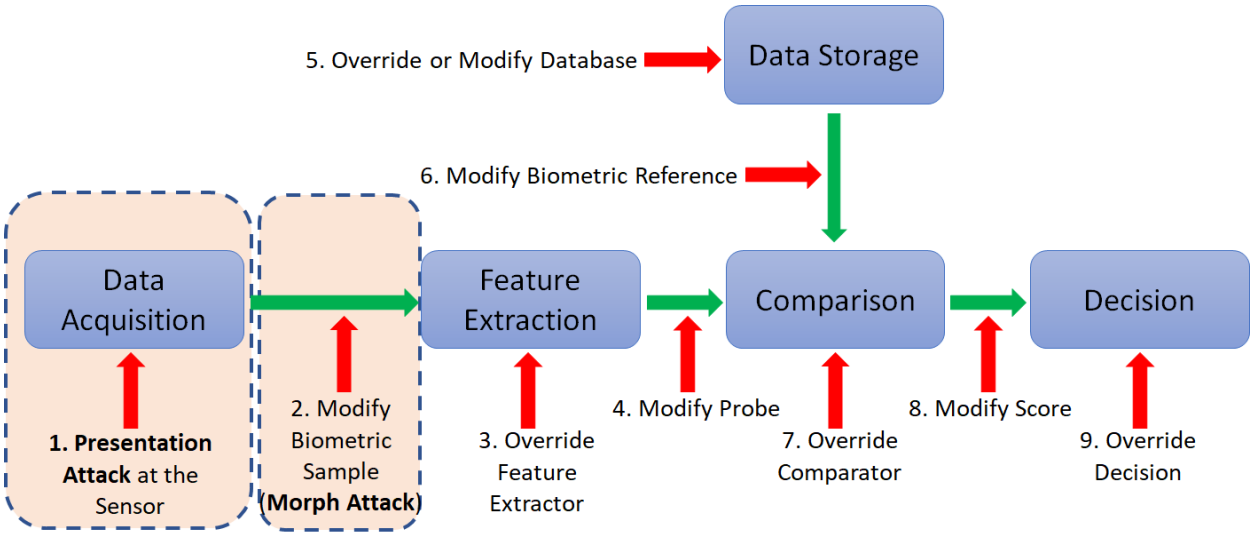


Figure 1.7: Generic biometric system and the various points of attacks launched on the system [224]. The attacks shown in orange boxes (presentation and morph attacks) are our focused research area.

system". The biometric characteristics or materials used to launch a presentation attack are termed Presentation Attack Instruments (PAIs). Examples of iris PAIs are printed iris images [57, 64, 113, 211], artificial eyes (plastic, glass, or doll eyes) [113, 152], cosmetic contacts [123, 212, 300], video display of an eye image [58, 214], cadaver eyes [58, 172], robotic eye models [143], holographic eye images [193], mannequin eye, and eye presentation under coercion. Figure 1.8 shows few samples of iris PAIs. In [3], iris PAIs are categorized as artificial, human-based, and other natural (animal or plant-based). The artificial PAIs are further categorized as complete (print or plastic eye) and partial (cosmetic contacts). The human-based PAIs are categorized as lifeless (cadaver eyes), altered (iris surgeries), non-conformant (off-gaze iris or occlusion by eyelids), coerced (unconscious or under duress), and conformant (zero effort impostor attempt). The objective is to detect the aforementioned presentation attacks along with future unknown attacks.

Typically, presentation attack detection (PAD) techniques follow are similar procedures as used by the biometric recognition system. It consists of capturing raw data from the sensor (additional or same recognition sensor), feature extraction from the raw data, and classification of features into detection or not detection classes based on pre-specified decision criteria. The PAD process can be performed simultaneously (additional sensor) or sequential (biometric recognition sensor).

In the literature, existing iris PAD techniques are categorized as *hardware-based* or *software-based*. The software-based techniques utilize the iris image captured from the conventional iris sensors, whereas the hardware-based techniques employ additional hardware to detect the liveness characteristics (eye blinking, pupil dilation or contraction, etc.). Czajka and Bowyer [58] proposed another categorization based on the type of input (image or video) and type of response (active or passive). The categorization includes four categories: static iris passively imaged (static features from still iris image), static iris actively imaged (features from multi-spectral iris images), dynamic iris passively imaged (features from pupil hippus), and dynamic iris actively imaged (dynamic features from stimulated pupil reflex). Various competitions and assessments of these iris PAD techniques can be found in [58, 61, 304–306].

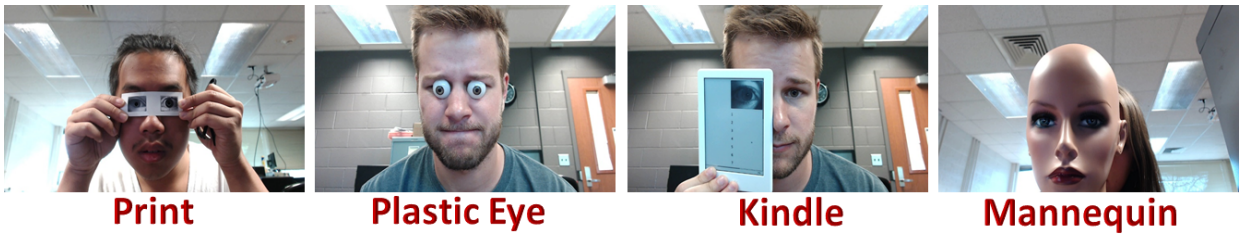


Figure 1.8: Various iris presentation attacks instruments.

1.4.2 Morph Attacks

Morph attack is another focus of our thesis. It is generally employed at the image-level on the raw biometric image captured from the acquisition sensor and before the image transfer to the feature extraction module. Though, it can also be performed at feature-level [85, 225], but requires knowledge of the feature extraction module. The morph attack at the image-level can be directed by presenting the modified biometric image to the sensor or by digitally uploading it to the biometric system. The morph attack entails the generation of an image (morphed image) that embodies multiple different identities. Typically, a biometric image is associated with a single identity; however, the morphed image can successfully match with multiple identities (Figure 1.9). It violates the fundamental uniqueness property of the biometric system. Morph attack is

mainly studied in the context of face recognition, where a single passport with a morphed face image allowed two individuals to pass through the border control security. It has not been widely investigated in iris recognition.

The morphed biometric images are generated using morphing techniques. In general, morphing [26] involves the creation of seamless transformation from one image into another. It creates intermediary morphed images by combining the two images in different proportions. It is a well-known field of research for the entertainment, education, or medical industry. However, in the context of biometric systems, it is recently being used to generate morphed biometric image that has the potential to attack the biometric system [86].

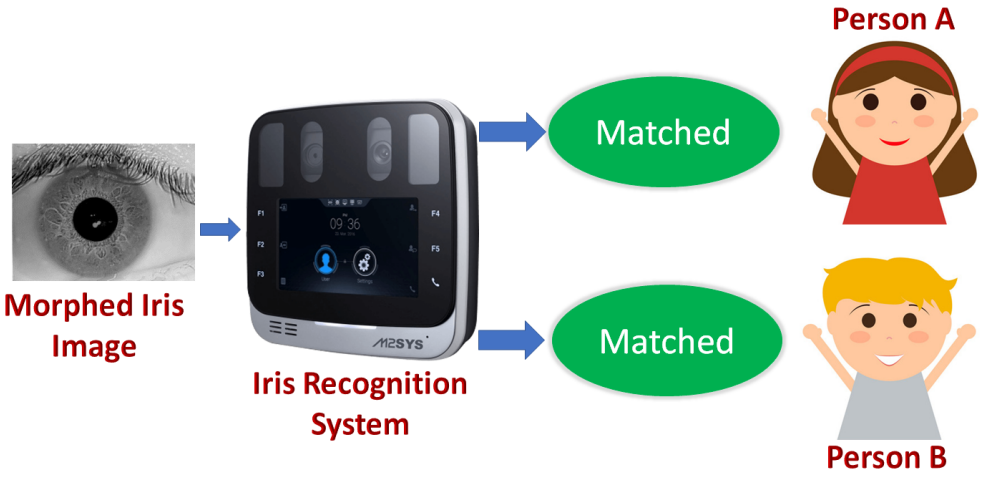


Figure 1.9: Pictorial diagram of iris morph attack. A single morphed iris image can authenticate two or more individuals, which violates the fundamental uniqueness characteristics of the biometric system.

1.5 Cross-modal Biometrics

Typically, a biometric system matches samples of the same modality for human recognition. However, cross-modal biometrics refers to the matching of different modalities to establish the identity of an individual. For instance, matching of iris images with face images [129], deducing phenotypic traits from genomic data [163], or mapping face image with voice signal [168, 185]. Figure 1.10 shows some of the examples of cross-model matching. The motivation of performing cross-modal matching comes from the various scenarios: (a) identify an individual when the legacy

database or corresponding enrolled template is not available, (b) improve recognition confidence even if the legacy dataset is available, (c) match noisy probe image when it cannot be matched with its legacy database, i.e., masked face image, and (d) connect databases of different biometric modalities to identify an individual globally.

Our focus is on the matching iris images with face images which constitutes the following challenges:

1. *Cross-Modality*: This involves matching iris modality images to face modality images. Though the ocular region is common in both the modalities, but the focus while acquisition is different in different modalities. In iris image acquisition, the focus is on the iris pattern, whereas in face image acquisition, the focus is on the entire face.

2. *Cross-Sensor*: Different sensors are used to capture the face and iris images. Sensors add various noises to the images, for instance, fixed pattern noise, pixel response non-uniformity (PRNU), random noise, etc.

3. *Cross-Spectrum*: Generally, face images are acquired using sensors typically operating in the visible spectrum (VIS) in contrast to iris images acquired using sensors operating in the near-infrared (NIR) spectrum. When considering the iris region only, NIR illumination (700-900nm) captures the stromal features (fibrovascular layer) of the iris, whereas VIS illumination (400-700nm) captures melanin pigment and a meshwork of ligament features.

4. *Cross-Resolution*: Iris or ocular regions of the face images are generally of very low resolution as compared to iris images.

1.6 Thesis Contributions

The main contributions of this proposal are as follows:

1. We propose an effective and robust software-based iris presentation attack (PA) detector called D-NetPAD using a single near-infrared (NIR) iris image (Figure 1.11 (a)). It is

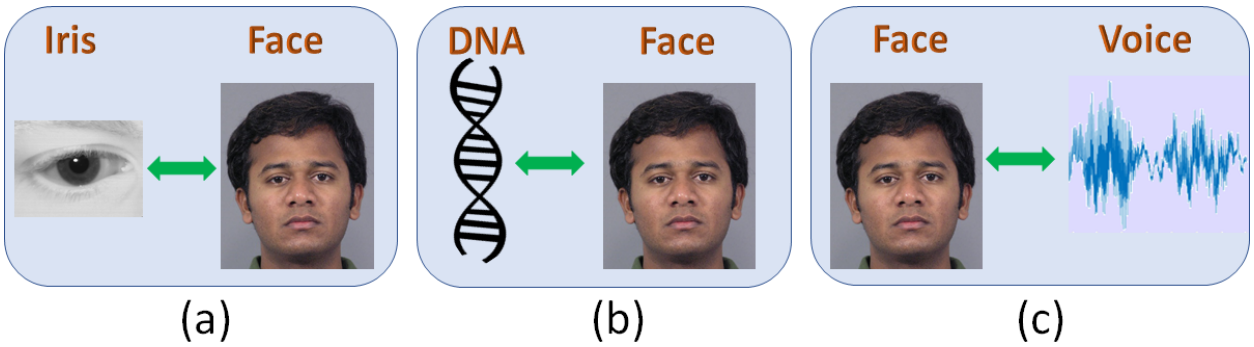


Figure 1.10: Examples of cross-modal matching: (a) iris modality matches with face, (b) deducing phenotypic traits from genomic data, and (c) mapping face image with the voice signal.

based on the densely connected convolutional neural network architecture. It demonstrates generalizability across PA artifacts, sensors, and datasets. We conduct experiments on a proprietary dataset and two publicly available datasets (LivDet-Iris 2017 and LivDet-Iris 2020) that substantiate the effectiveness of the proposed method for iris PA detection. The proposed method results in a true detection rate of 98.58% at a false detection rate of 0.2% on the proprietary dataset and outperforms state-of-the-art methods on the LivDet-Iris 2017 and LivDet-Iris 2020 datasets. We also explore the explainability and interpretability of our method using t-SNE plots and Grad-CAM which help in visualizing intermediate feature distributions and fixation heatmaps, respectively. Further, we conduct a frequency analysis to explain the nature of features being extracted by the network.

2. We design a hybrid (a combination of hardware and software-based) iris presentation attack detector utilizing short videos (approx. 4 secs) captured from a webcam (Figure 1.11 (b)). The videos are in the visible spectrum focusing on the user interaction with the iris sensor. To extract discriminative features from the scene, we develop various spatial-temporal feature extraction techniques. Evaluation is performed on a proprietary dataset (IPV dataset) of 121 subjects. We also extend it for detecting PAs in face modality. For the face modality, experiments are performed on three publicly available datasets (SiW, SiW-M, and OULU-NPU). The proposed approach generalizes well across different environments (e.g., changes in illumination or background), presentation attacks, and modalities.

3. We propose a hardware-based iris PA detector utilizing Optical Coherence Tomography (OCT) imaging (Figure 1.11 (c)). The OCT imaging provides a cross-sectional view (internal structure) of an eye, whereas traditional imaging provides 2D iris textural information. Its viability is assessed by comparing its performance with respect to traditional iris imaging modalities, viz., near-infrared (NIR), and visible spectrum. PA detection is performed using three state-of-the-art deep architectures (VGG19, ResNet50, and DenseNet121) to differentiate between bonafide and PA samples for each of the three imaging modalities. Experiments are performed on a proprietary dataset of 2,169 bonafide, 177 Van Dyke eyes, and 360 cosmetic contact images acquired using all three imaging modalities under intra-attack (known PAs) and cross-attack (unknown PAs) scenarios. Promising results demonstrate OCT as a viable solution for iris presentation attack detection.

4. We assess the robustness of the iris PA detector against architectural parameter perturbations. The robustness analysis involves three state-of-the-art architectures (VGG, ResNet, and D-NetPAD), three types of parameter perturbations (Gaussian noise, weight zeroing, and weight scaling), and two settings (entire network and layer-wise). We conduct evaluations on the LivDet-Iris-2017 and LivDet-Iris-2020 datasets. Based on the robustness analysis, we propose improved models simply by perturbing parameters of the network without further training. Then we combine these perturbed models to improve performance over the original model. The ensemble models show a 47.59% average improvement on LivDet-Iris-2017 dataset and 5.44% on the LivDet-Iris-2020 dataset.

5. We propose a retrain methodology to maintain the performance of iris PAD in non-stationary environment. The methodology involves building a new expert model using new oncoming training data, and makes a final decision for a probe sample by a weighted sum of old and new iris PAD models scores. We assign the weights dynamically at the run-time for each probe sample using in-domain models (separate from iris PAD models). The in-domain model provides information about the membership of a probe sample to the training data. To

evaluate the proposed method, we experiment with three setups. The first two setups are in the application of detecting presentation attacks in iris biometric modality. The third setup compares the proposed method against state-of-the-art continual learning methods on the split MNIST dataset.

6. We investigate the other adversary attack called morph attacks in the context of iris biometrics. We perform iris morphing at the image level and generate morphed iris images from two available datasets (IIITD and WVU-Multimodal). We then demonstrate the vulnerability of three different iris recognition methods (VeriEye, USITv3.0, and CNN-Pairwise) to morph attacks with a success rate of over 90% at a false match rate of 0.01%. Finally, we provide preliminary results on the detection of morphed iris images.

7. We propose learning-based methods to perform cross-modal matching (iris modality images match against face modality images). Cross-modal recognition mainly encounters two challenges: (i) a large domain gap due to different sensors, spectra, and resolutions, and (ii) an imbalance in the training data. We propose three deep learning approaches to address these two challenges. The first approach is at the feature-level, where we jointly extract discriminative features from two modality images to reduce their domain gap and termed it Multi-channel CNN. The second approach is at the image-level, where one domain image is transformed into another utilizing various GAN architectures. The third approach is at the training-level that resolves the imbalanced training data by generating samples of under-represented class using the Dual Variational Generation (DVG) framework. We conduct experiments on BioCop-2008, BioCop-2009, WVU multi-modal, and cross-spectrum PolyU datasets to substantiate the effectiveness of the proposed approaches.

1.7 Thesis Organization

The remaining document is organized as follows:

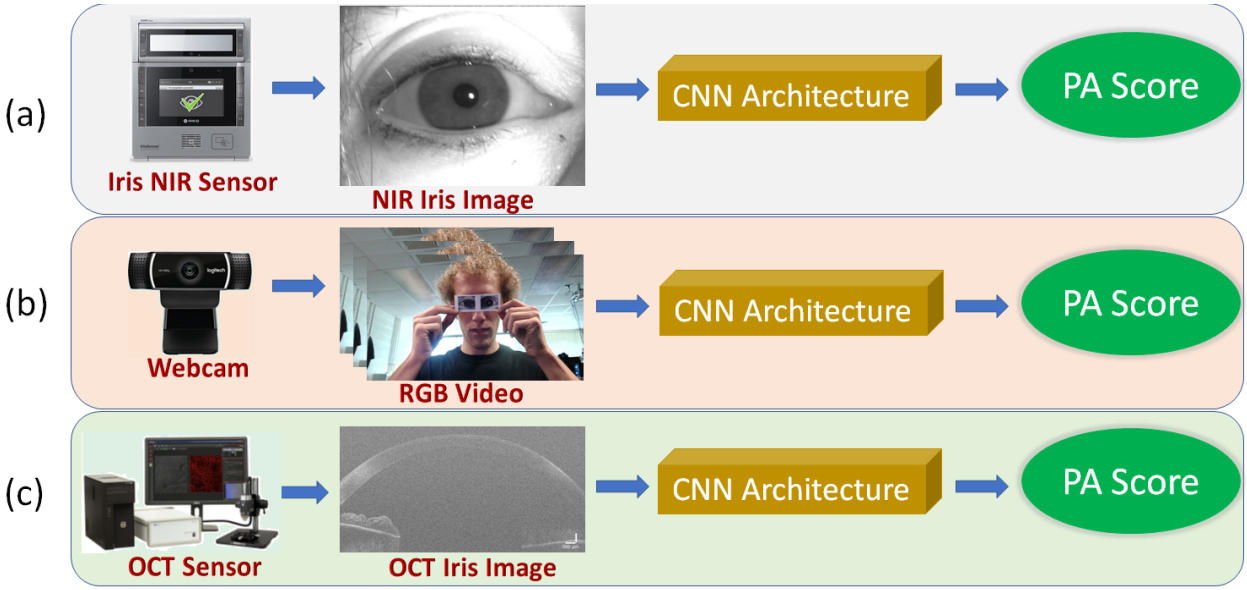


Figure 1.11: Different categories of techniques applied to detect iris presentation attacks: (a) technique utilizing a single NIR iris image captured from conventional iris recognition sensor and (b) technique utilizing a video captured from webcam (c) technique utilizing a single iris OCT image. All these techniques generate a Presentation Attack (PA) score between 0 and 1, where '0' corresponds to bonafide input sample and '1' corresponds to PA input.

- Chapter 2 describes the software-based iris PA detector (D-NetPAD) which utilizes a single NIR iris image. This chapter covers the first contribution.
- Chapter 3 entails the hybrid iris PA detector which utilizes a scene video captured from a webcam. This chapter covers the second contribution.
- Chapter 4 describes the hardware-based iris PA detector which utilizes OCT imaging. This chapter covers the third contribution.
- Chapter 5 assess the robustness of the iris PA detector along with other deep neural networks. This chapter covers the fourth contribution.
- Chapter 6 explores the retraining strategies for iris PA detectors to keep the performance over time. This chapter covers the fifth contribution.
- Chapter 7 introduces the iris morph attacks, their potential to attack iris biometric systems, and their detection technique. This chapter covers the sixth contribution.

- Chapter 8 provides insight into cross-modality matching of iris images with face images. This chapter covers the seventh contribution.
- Chapter 9 concludes the thesis and provides some directions for future work.

CHAPTER 2

IRIS PRESENTATION ATTACK DETECTION USING A SINGLE NIR IMAGE

Parts of this chapter appeared in the following publications:

R. Sharma and A. Ross, "D-NetPAD: An Explainable and Interpretable Iris Presentation Attack Detector," International Joint Conference on Biometrics (IJCB), September 2020.

P. Das et al., "Iris Liveness Detection Competition (LivDet-Iris) - The 2020 Edition," International Joint Conference on Biometrics (IJCB), September 2020.

2.1 Introduction

In this chapter, we present a iris presentation attack detection (PAD) method which utilizes a near-infrared (NIR) iris image captured from the conventional iris sensor. Therefore, the method does not impose additional overhead when integrated with the existing iris recognition system. The method is based on a densely connected convolutional neural network (DenseNet). The DenseNet architecture has a unique property that each layer is connected to every other layer in a feed-forward fashion and the features across different layers correspond to different resolutions. The aggregation of these multi-resolution features efficiently characterizes the iris pattern as the iris pattern is arguably stochastic in nature and the intricate features of the iris stroma are manifested in multiple resolutions [63]. The source code and trained model of the proposed method are available at https://github.com/iPRoBe-lab/D-NetPAD.

The main contributions of the work are as follows:

1. We propose an effective and robust iris PA detector named as D-NetPAD that is based on the DenseNet architecture [122]. We also demonstrate that the proposed detector exhibits generalizability across different PAs, sensors, and datasets.

2. We evaluate the performance of D-NetPAD on a proprietary dataset and two publicly available dataset (LivDet-2017 and LivDet-2020).

3. We perform visualizations using t-SNE plots [280] and Grad-CAM [245] to explain the performance of the proposed method. The t-SNE plots provide visualization of features obtained from the intermediate layers of the model. The Grad-CAM produces heatmaps emphasizing the salient regions in an iris image that are used by the network to detect iris PAs.

4. We also conduct a frequency analysis to understand the frequencies learned by the model and, based on that, interpret its performance.

The rest of the chapter is organized as follows: Section 2.2 provides the brief description of the related work. Section 2.3 discusses the architecture of the proposed method. Section 2.4 describes the experimental setup and results on all the datasets. Section 2.5 provides a detailed analysis of the results obtained from the D-NetPAD. Section 2.7 concludes the chapter.

2.2 Related Work

Existing techniques in the literature used to counter iris PAs can be categorized as being either hardware-based or software-based. Hardware-based techniques typically require physical devices in addition to the conventional iris sensor to aid in PA detection. The additional hardware assists in capturing intrinsic properties of the eye (e.g., corneal reflection, red-eye effect, etc.), involuntary behavioral characteristics (pupil hippus, eye blinking, etc.), or challenge-response behavior which could be voluntary (eye-tracking) or involuntary (pupil dynamics with external light). Daugman [66] suggested the use of spectrographic properties of the eye (tissue, blood) and four Purkinje images generated by the reflection from the outer and inner surface of the cornea and lens. Further, Lee *et al.* [152] analyzed the changes in the reflectance ratio between the iris and sclera under multi-spectral illumination. Czajka *et al.* [56] utilized IrisCUBE camera to capture pupil dynamics, whereas Kanematsu *et al.* [135] used CCD camera with two white LEDs to initiate

and record pupillary reflex. Hughes *et al.* [123] created 3D structural modeling of an eye using stereo imaging. Raghavendra and Busch [219] explored the inherent characteristics of Light Field Camera (LFC) in the VIS spectrum for iris detection. Komogortsev and Karpov [144] used the EyeLink II eye tracker to capture Oculomotor Plant Characteristics as a cue for detecting PA. Sharma and Ross [250] introduced the use of Optical Coherence Tomography (OCT) imaging for iris PA detection.

On the other hand, software-based techniques extract salient features from the *digital* iris image in order to classify it as a bonafide or a PA.[1] Daugman [65] distinguished patterned contacts from real iris images using the amplitude spectrum of the 2-D Fourier Transform. Other researchers also analyzed frequency spectrum using 2-D Fourier spectra [108], Wavelet Transform [109] and Laplacian pyramids [217]. Various handcrafted features are used to detect the iris PA, for instance, HVC [315], LBP [119], BSIF [220], and SID [97]. However, more recently, a number of deep-learning based methods have been proposed [46, 114, 176, 194, 301, 302]. Menotti *et al.* [176] proposed a deep architecture for PA detection called SpoofNet. Pala and Bhanu [194] developed a deep framework built upon triplet convolutional networks. Hoffman *et al.* [113] focused on detecting iris PAs utilizing a patch-batch convolutional neural network (CNN) that is observed to perform well in the cross-sensor and cross-dataset scenarios. They extend their work [114] by analyzing the importance of utilizing the periocular region in detecting iris PAs. Chen and Ross [46] proposed a multi-task CNN for first detecting the iris region and then classifying it. In their other work [45], they explored IrisCodes for PA detection, so that commercially used IrisCodes could be authenticated. Yadav *et al.* [302] utilized a Relativistic Average Standard Generative Adversarial Network (RaSGAN) as a one-class classifier to detect unseen or unknown iris PAs. In another work, Yadav and Ross [303] proposed Cyclic Image Translation Generative Adversarial Network (CIT-GAN) for augmenting under-represented iris PAs in the training set. Chen and Ross [47] worked in the direction of explaining the model predictions by incorporating attention mechanisms in the CNN network which provides visual explanations to the predictions. The Liveness Detection-Iris

[1] A “bonafide" image is sometimes referred to as a “live" image in the literature.

Competition (LivDet-Iris) held in 2013 [305], 2015 [306], 2017 [304] and 2020 [61] provided a comprehensive comparative report of different iris PA detection techniques. Czajka and Bowyer [58] also presented a detailed assessment of various state-of-the-art iris PA detection (PAD) algorithms. While most of these methods resulted in very high PA detection rates, generalizability across PAs, sensors, and datasets is still a challenging problem [79, 212, 300].

2.3 D-NetPAD: Description and Rationale

Dense Network Presentation Attack Detection (D-NetPAD) is based on the Densely Connected Convolutional Network 121 (DenseNet121) [122] architecture. The architecture consists of 121 convolutional layers of kernel size 7×7, followed by a max-pooling layer and a series of Dense blocks and Transition layers. There are four Dense blocks, and three Transition layers lie between successive Dense blocks. Each Dense block consists of two convolutional layers of kernel size 1×1 and 3×3. Both convolutional layers are followed by a non-linear ReLU activation layer. The Transition layer consists of one convolutional layer of kernel size 1×1 and an average pooling layer. It reduces the size of feature-maps, which is kept constant within a Dense block. The last layer is a fully connected layer. The work in [301] exploits the DenseNet architecture of depth 22 with three densely connected blocks.

The most notable characteristic of DenseNet is that each layer connects to every other layer in a feed-forward fashion. In other words, each layer obtains feature-maps from preceding layers and passes its feature-maps to subsequent layers. The features from preceding layers are combined by concatenation as opposed to the summation performed in the ResNet [105] architecture. The concatenation removes the constraint of having the same dimension across the feature-maps. In this way, the architecture ensures the maximum flow of information in the forward direction and also resolves the most prevalent challenge of vanishing gradient in the backward direction. Another major advantage of DenseNet121 is that it supports such densely and deeply connected network with fewer trainable parameters (7,978,856) as compared to its counterpart ResNet50 (35,610,216) [105] or VGG19 (143,667,240) [255]. This is because DenseNet uses a small set of

filters in each layer (e.g., 12 filters/layer) compared to the traditional convolutional networks (~128 or 256 filters/layer). DenseNet preserves the feature-maps and reuses it in the subsequent layers instead of relearning feature-maps every time. The reusability of feature-maps helps in alleviating the over-fitting problem, especially in the case of limited training data. These architectural tweaks help in generating an efficient feature representation for the highly textured iris pattern. Feature-maps at each layer capture specific spatial and frequency information and consolidation of these feature-maps result in the extraction of multi-resolution features. These features are efficient in characterizing the stochastic nature of the iris pattern. The intricacy of a bonafide iris pattern is not present in the spoofed iris (print eye, artificial eye, or cosmetic contact), and this difference is efficiently captured by the features generated from DenseNet. The consolidation of feature-maps at the last layer also smoothens the decision boundaries, resulting in better generalization across PA artifacts, sensors, and datasets.

Figure 2.1 shows the flowchart of the proposed architecture. The iris sensor acquires an ocular image which is input to the iris detection module. In our implementation, we use the VeriEye iris detector, which outputs the centers of the iris and pupil along with their radii. The iris region is cropped from the ocular image using the center and radius of the iris. The cropped iris region is then resized to 224×224 and input to the pre-trained DenseNet121 network. The ImageNet dataset [72] is used to pre-train the network. The network produces a single presentation attack (PA) score, which lies between 0 and 1. A score approaching '1' indicates that the input sample is a PA, whereas a score approaching '0' indicates that the input sample is a bonafide. We determine the threshold by fixing the False Detection Rate to 0.2% in order to get the final classification. If the PA score is less than the specified threshold, the input sample is labeled as a bonafide; otherwise, it is a PA. During training, the learning rate used is 0.005, the batch size is 20, the optimization algorithm used is the stochastic gradient descent with a momentum of 0.9, the number of epochs is 50, and the loss function is cross-entropy.

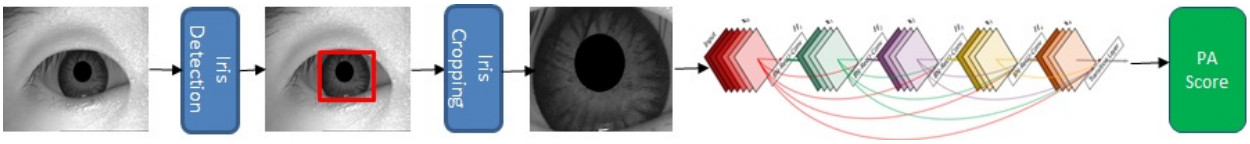


Figure 2.1: Flowchart of the D-NetPAD algorithm. Iris region (red box) is detected and cropped from the ocular image and input to the D-NetPAD architecture. The base architecture used in D-NetPAD is DenseNet121 [122]. It produces a single PA score within a range of 0-1, which determines whether an input image is a bonafide (value towards 0) or a PA (value towards 1).

2.4 Evaluation and Results

We performed experiments on a proprietary dataset as well as two publicly available benchmark dataset (LivDet-2017, LivDet-2020) to evaluate the performance of D-NetPAD. The proprietary dataset has several subsets and is, therefore, referred to as the “Combined Dataset" in the rest of the document. The Combined and the LivDet2020 datasets correspond to the cross-PA scenario, whereas the LivDet-2017 dataset creates a test-bed for cross-PA, cross-sensor, and cross-dataset testing scenarios. In the cross-PA scenario, we use PA instruments (PAIs) that were not used during the training. In the cross-sensor scenario, we evaluate images from different sensors than those used during the training. The cross-dataset scenario incorporates testing under different PAIs, sensors, data acquisition environments (indoor/outdoor, varying illumination conditions), subject populations, and platforms (desktop and mobile). The cross-dataset scenario accounts for large variations, making it the most challenging test scenario.

2.4.1 Combined Dataset: Description and Results

The Combined Dataset was collected under the IARPA Odin program (Presentation Attack Detection) [2]. The IrisAccess iCAM7000 sensor was used to collect the data. The dataset is a combination of various component datasets collected at different locations and times using different units of the same sensor. Table 2.1 provides the description of the component datasets. There are a total of 13,851 iris images out of which 9,660 are bonafide and 4,291 are PAs. The PA samples in the dataset correspond to the following attack instruments: print, artificial eye, cosmetic contacts, kindle replay, and transparent dome on print. Figure 2.2 shows sample images from the dataset.

The test set JHU-APL03 (Table 2.1) comprises two types of artificial eyes and 10 different types of cosmetic contacts. It corresponds to the cross-PA scenario as it contains six additional cosmetic contacts that are not used during training. As the process of collecting cosmetic contact images is a tedious and time-consuming process, its quantity is limited in the training set. Therefore, we utilize cosmetic contact images from NDCLD-2015 [267] to overcome the shortcoming. The bonafide images in the NDCLD-2015 dataset are not used as the Combined dataset has a large number of bonafide images. The NDCLD-2015 dataset was collected using the IrisGuard AD100 and IrisAccess LG4000 sensors.

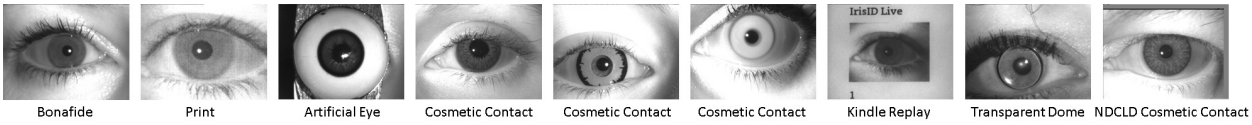


Figure 2.2: Sample images of bonafide and different types of PAs (print, artificial eye, cosmetic contact, kindle replay, and transparent dome on print) taken from the Combined dataset. The last cosmetic contact image is taken from the NDCLD-2015 dataset.

Table 2.1: Description of different components of the Combined Dataset. Details of the train and test set of the Combined and NDCLD 2015 datasets are also provided in terms of the number of bonafide and PA images. Here, MSU stands for Michigan State University, CU stands for Clarkson University, and JHU-APL stands for Johns Hopkins University-Applied Physics Laboratory.

Dataset	**Train**									**Test**
	MSU IrisPA01	CU IrisPA01	CU IrisPA02	PB IrisPA01	PB IrisPA02	PB IrisPA03	JHU-APL01	JHU-APL02	NDCLD 2015	JHU-APL03
Bonafide	381	962	1,107	446	518	518	1,394	1,371	-	2,963
Print	991	660	415	14	-	-	-	-	-	-
Artificial Eye	318	34	-	21	9	12	49	111	-	175
Cosmetic Contacts	-	-	208	-	21	94	78	120	2,236	177
Kindle Replay	51	79	-	-	-	-	-	-	-	-
Transparent Dome	-	-	503	9	-	-	42	-	-	-
Acquisition Time Period	Nov 2017	Nov 2017	Dec 2018	April 2018	Feb 2019	Sept 2019	May 2018	May 2019	2015	Nov 2019

We evaluate the performance of the D-NetPAD in terms of True Detection Rate (TDR) at a False Detection Rate (FDR) of 0.2%. TDR is the percentage of PA samples correctly detected, whereas FDR is a percentage of bonafide samples incorrectly classified as PA.[2] The D-NetPAD is compared against two deep learning-based methods ([46] and [114]) as these are state-of-the-art (SoTA) methods. It is also compared with VGG19 [255] and ResNet101 [105] deep architectures. Table

[2]Other commonly used evaluation measures for presentation attack detection are Attack Presentation Classification Error Rate (APCER) and Bonafide Presentation Classification Error Rate (BPCER). TDR is 1−APCER, and FDR is the same as BPCER.

2.2 presents the results of all five algorithms. **The D-NetPAD results in a TDR of 98.58% and outperforms the SoTA methods [114], [46], VGG19 and ResNet101 by 25.27%, 6.44%, 2.41% and 1.75%, respectively**. Low performance of [114] (a network of eight convolutional layers) substantiates the use of deeper architectures. We also experiment to emphasize the importance of iris localization in the proposed method. When we input the entire ocular image into the DenseNet121 network, it resulted in a lower TDR of 94.59% at 0.2% FDR. Next, we analyze the failure cases (misclassified cases) of our method. At 0.2% FDR, there are four bonafide samples misclassified as PAs and five PAs samples misclassified as bonafide. Figure 2.3 shows the misclassified images. In the case of misclassified bonafide images, subjects wearing transparent contact lenses and glare of the light reflected from the glasses, resulting in misclassification. In the case of misclassified PA images, the D-NetPAD fails for a particular type of cosmetic contact lens (Halloween-style Extreme contact lens), where the pattern appears only at the periphery of the cosmetic contact. Segmentation ignores the outer region of the iris containing some artifacts of these cosmetic contacts. This resulted in a smaller region of the artifact being fed into the DenseNet for PA detection, leading to a misclassification.

Table 2.2: The results of D-NetPAD in term of TDR (%) at 0.2% FDR on the Combined dataset. The method is compared with four other algorithms.

Algorithms	[114]	[46]	VGG19	ResNet101	D-NetPAD
TDR (%) @ 0.2% FDR	78.69	92.61	96.26	96.88	**98.58**

2.4.2 LivDet-2017 Dataset: Description and Results

Another dataset used for evaluation is the LivDet-2017 [304] dataset. The LivDet-2017 dataset is a combination of four datasets: Clarkson, Warsaw, Notre Dame, and IIITD-WVU datasets. Figure 2.4 shows few samples of LivDet-2017 dataset. Table 2.3 describes the types of PAs present in the datasets, and the number of images in the train and test sets of all four datasets. The Clarkson dataset represents the cross-PA testing scenario. The test set consists of 5 additional cosmetic contacts and prints of visible spectrum iris images captured using an iPhone 5. The Warsaw dataset helps in

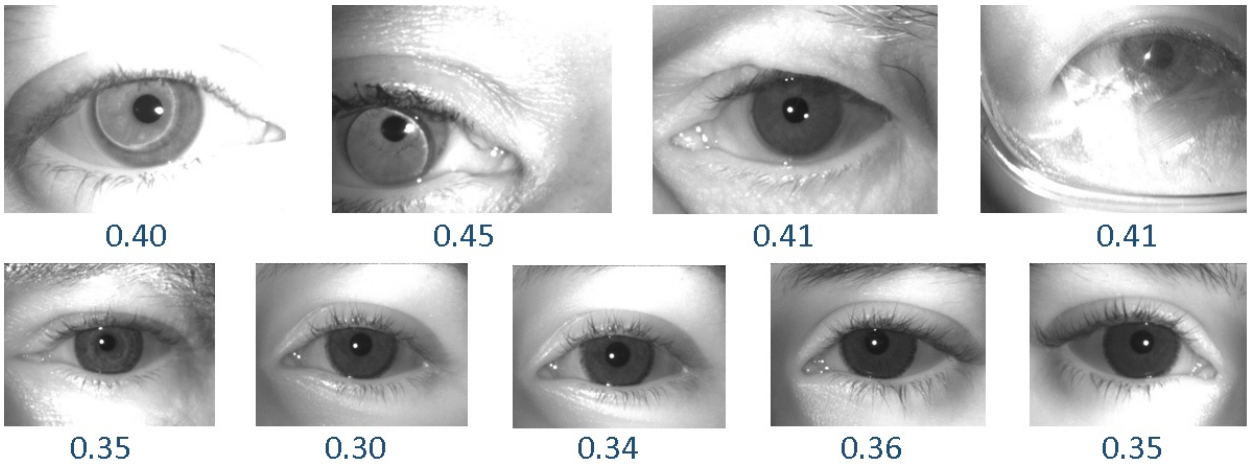


Figure 2.3: Misclassified images by the D-NetPAD algorithm on the JHU-APL03 test set. The first row shows bonafide images that are misclassified as PA. The second row shows PA images that are misclassified as bonafide. The PA score is displayed at the bottom of each image. The threshold for classification is 0.40, where a PA score below the threshold is considered to be a bonafide.

evaluating the cross-sensor testing scenario. It consists of two test sets: a “known" sensor and an “unknown" sensor. The IrisGuard AD100 sensor is used to capture the images of the training set and the “known" component of the test set. Images of the “unknown" component of the test set are captured by a setup composed of Aritech ARX-3M3C camera, SONY EX-View CCD sensor, Fujinon DV10X7.5A-SA2 lens, and B+W 092 NIR filter. The Notre Dame dataset corresponds to the cross-PA scenario. It also contains two test sets (“known" and “unknown"). The “unknown" test set includes cosmetic contacts not used in the training set. The IIITD-WVU dataset consists of data collected by IIITD and WVU. The IIITD data is used for training, whereas the WVU data is used for testing. The dataset corresponds to the cross-dataset scenario, where the test set incorporates variations in the sensors, data acquisition environment, subject population, and PA generation procedures. The training set is captured in a controlled environment using two iris sensors: Cogent dual iris sensor (CIS 202) and VistaFA2E single iris sensor. The test set is captured using the IriShield MK2120U mobile iris sensor at two different locations: indoors (controlled illumination) and outdoors (varying environmental conditions). The cross-dataset testing scenario represents the most difficult case.

For a detailed evaluation of the D-NetPAD, we created three models of the D-NetPAD network,

Table 2.3: Description of the train and test sets of all four subsets of the LivDet-2017 dataset along with the number of bonafide and PA images present in the datasets. The information about the sensors is also provided. Each subset represents different testing scenarios. The Clarkson and Notre Dame test sets correspond to the cross-PA scenario, whereas the Warsaw data corresponds to the cross-sensor scenario. The IIITD-WVU represents a cross-dataset scenario. Here, “K. Test" means a known test set of the dataset, and “U. Test" means an unknown test set.

Dataset	**Clarkson (Cross-PA)**		**Warsaw (Cross-Sensor)**			**Notre Dame (Cross-PA)**			**IIITD-WVU (Cross-Dataset)**	
	Train	Test	Train	K. Test	U. Test	Train	K. Test	U. Test	Train	Test
Bonafide	2,469	1,485	1,844	974	2,350	600	900	900	2,250	702
Print	1,346	908	2,669	2,016	2,160	-	-	-	3,000	2,806
Cosmetic Contacts	1,122	765	-	-	-	600	900	900	1,000	701
Sensor	IrisAccess EOU2200		IrisGuard AD100		Aritech ARX-3M3C, Fujinon DV10X7.5A, DV10X7.5A-SA2 lens B+W 092 NIR filter	IrisGuard AD100, IrisAccess LG4000			Cogent CIS 202, VistaFA2E	IriShield MK2120U

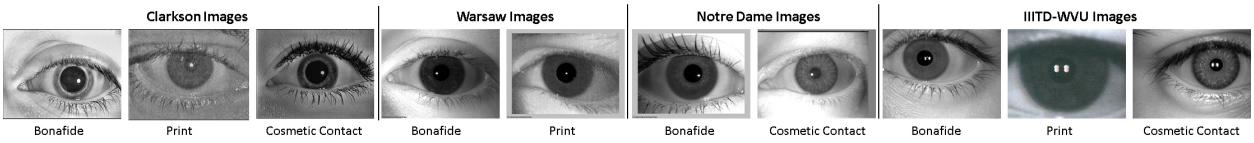


Figure 2.4: Sample images of bonafide and different types of PAs (print, cosmetic contact) taken from each subset of the LivDet-2017 dataset.

which differ in their training process: (i) **Pre-trained D-NetPAD**: The model trained on the Combined dataset is directly used; (ii) **Scratch D-NetPAD**: The model is trained from scratch on the LivDet-2017 train sets; and (iii) **Fine-tuned D-NetPAD**: The model that is pre-trained on the Combined dataset is fine-tuned using the LivDet-2017 train sets. The performance measure used is the same as used in [304]: Attack Presentation Classification Error Rate (APCER) and Bonafide Presentation Classification Error Rate (BPCER). The APCER is the proportion of PA samples misclassified as bonafide, whereas the BPCER is a proportion of bonafide samples misclassified as PAs. The D-NetPAD is compared against the top three winners of the LivDet-2017 competition. Table 2.4 summarizes the results of all algorithms. While the pre-trained D-NetPAD model and the model trained from scratch perform at par with the state-of-the-art methods, the fine-tuned D-NetPAD model outperforms the other methods.

We also measured the performance of D-NetPAD in terms of its TDR at 0.2% FDR on the LivDet-2017 dataset. Table 2.5 compiles the results of D-NetPAD on all four datasets of the

LivDet-2017 [1] dataset. A summary of the results is provided below:

Clarkson Test Dataset: The pre-trained D-NetPAD fails on the test set of Clarkson. The Clarkson dataset corresponds to the cross-sensor and cross-PA scenarios. The images captured from IrisAccess EOU2200 is visually quite different from the images captured by the iCAM 7000 iris sensor, which results in poor performance (28.63%). But, the result improves (92.05% and 93.51%) when the training set (scratch or fine-tuned) includes the Clarkson train set (sensor information).

Warsaw Test Dataset: The pre-trained D-NetPAD achieves competent performance on the Warsaw dataset. The sensors and types of PA used in the Warsaw dataset are different from the one used in the training, but the images captured by the test sensors are visually similar, which results in comparable TDR. Fine-tuning the pre-trained D-NetPAD using the train set of Warsaw dataset results in 100% TDR.

Notre Dame Test Dataset: The dataset represents the cross-PA scenario, where the test set uses additional cosmetic contacts. The pre-trained D-NetPAD model trained on diverse cosmetic contacts generalizes well across previously unseen cosmetic contacts (93.55% and 91%). Its performance drops on the unknown test set (66.55%) when the model is trained from scratch as the diversity of cosmetic contacts is limited in the Notre Dame train set. Fine-tuning the model with the Notre Dame train set achieves 100% TDR.

IIIT-WVU Test Dataset: The dataset is the most challenging dataset where the test set images are captured using the IriShield MK2120U mobile iris sensor and under different capturing environments (indoor and outdoor). The dataset also included unseen PAs, resulting in very low TDRs for all three models (42.91%, 29.30%, and 48.85%). We further analyze the results of IIIT-WVU by plotting the PA score distributions of the bonafide and PAs, and estimating the d-prime distance between them (Figure 2.5). Though the TDR is quite low in the case of fine-tuned D-NetPAD, its histogram shows a better separation ($d' = 2.64$) between the score distributions of bonafide and PAs.

The D-NetPAD algorithm demonstrates robustness across PAs and sensors testing scenarios

after the fine-tuning but fails in the case of cross-dataset which is a combination of cross-PA, cross-sensor, cross-environment, and cross-platform scenarios. Here, cross-platform implies training on images of iris sensor meant for desktop (e.g., IrisAccess iCAM7000) and testing on images of iris sensor meant for mobile devices (e.g., IriShield MK2120U).

Table 2.4: D-NetPAD performance reported in terms of APCER and BPCER on all subsets of the LivDet-2017 dataset. The method is compared with three state-of-the-art algorithms in [304], which are the winners of the LivDet-2017 competition.

Algorithm	Clarkson		Warsaw		IIITD-WVU		Notre-Dame		Averaged	
	APCER	BPCER	APCER	BPCER	APCER	BPCER	APCER	BPCER	APCER	BPCER
CASIA	9.61	5.65	3.4	8.6	23.16	16.1	11.33	7.56	11.88	9.48
Anon1	15.54	3.64	6.11	5.51	29.4	**3.99**	7.78	0.28	14.71	3.36
UNINA	13.39	**0.81**	0.05	14.77	23.18	35.75	25.44	0.33	15.52	12.92
Pre-Trained D-NetPAD	16.73	19.46	1.66	0.83	16.05	15.24	1.00	2.22	8.86	9.43
Scratch D-NetPAD	5.78	0.94	**0**	**0.04**	36.41	10.12	10.38	3.23	13.14	3.58
Fine-tuned D-NetPAD	**2.99**	2.97	**0**	0.54	**1.88**	8.84	**0.33**	**0.27**	**1.3**	**3.15**

Table 2.5: D-NetPAD performance reported in terms of the TDR (%) @ 0.2% FDR on different subsets of the LivDet-2017 dataset. Three models of D-NetPAD are generated by varying their training data.

Algorithm	Clarkson	Warsaw		Notre-Dame		IIITD-WVU
	Test	K. Test	U. Test	K. Test	U. Test	Test
Pre-Trained D-NetPAD	28.63	92.95	98.56	93.55	91.00	42.91
Scratch D-NetPAD	92.05	100	100	100	66.55	29.30
Fine-tuned D-NetPAD	93.51	100	100	100	99.77	48.85

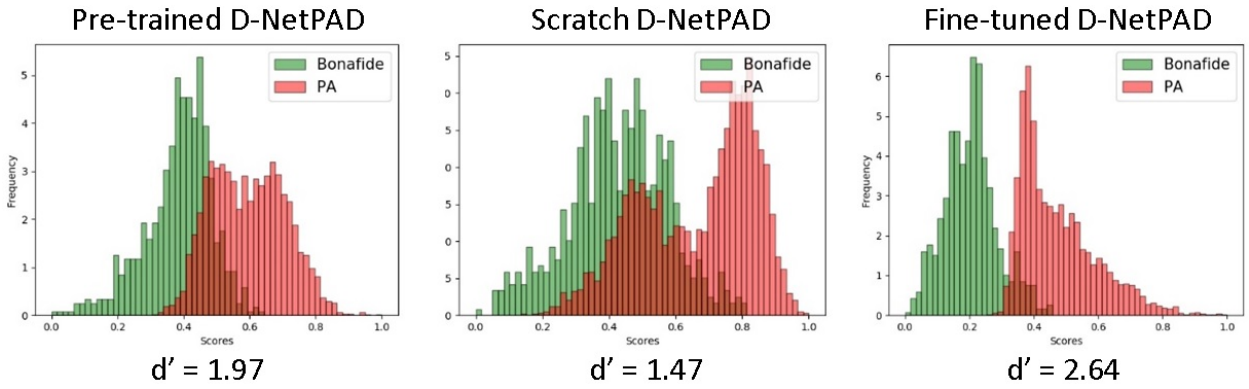


Figure 2.5: Histograms of the three trained models of D-NetPAD on the IIITD-WVU test set. For accurate classification, there should be minimal overlap between the two (red and green) distributions. This plot indicates the efficacy of the fine-tuned D-NetPAD.

2.4.3 LivDet-2020 Dataset: Description and Results

The last dataset used for evaluation is the LivDet-2020 dataset [61]. The dataset is from the LivDet-Iris-2020 competition launched in May 2020 by five organizations: Clarkson University (USA), University of Notre Dame (USA), Warsaw University of Technology (Poland), IDIAP Research Institute (Switzerland), and Medical University of Warsaw (Poland). The LivDet-2020 dataset is different from previous editions ([304–306]) in that the organizers did not announce any official training set, only a testing set is provided. The testing set employed in the competition is a combination of data from all three organizers: Clarkson University, University of Notre Dame, and Warsaw University of Technology. The dataset consists of 12,432 images (5,331 bonafide and 7,101 PA samples). Five presentation attack instruments (PAI) categories included in the dataset are printed eyes (1,049), cosmetic contact lenses (4,336), kindle/electronic displayed eyes (81), fake/prosthetic/printed eyes with add-ons (541), and cadaver eyes (1,094). The fake eyes with add-ons include five subcategories: cosmetic contacts on printed eyes, cosmetic contacts on doll eyes, clear contacts on printed eyes, eye dome on printed eyes, and doll eyes. Table 2.6 provides the number of images in each category of PAIs along with the sensor information used to capture those images. Figure 2.6 shows few samples of the LivDet-2020 dataset.

For the evaluation of the D-NetPAD on the LivDet-Iris-2020 dataset, D-NetPAD is trained on the Combined Dataset along with the partial data from the Warsaw PostMortem v3 dataset [275] (1,200 cadaver iris images from the first 37 cadavers). The base architecture used is DensetNet161, which consists of 4 dense blocks and 161 convolutional layers. The evaluation measures used are APCER and BPCER. The APCER is calculated for the individual PA types as well as for overall PAs. The proposed method is compared with the top three winners of the LivDet-2020 competition. Table 2.7 summarizes the results of all algorithms. The D-NetPAD outperforms the other methods by a large margin. D-NetPAD resulted in a weighted APCER of 2.76% at a BPCER of 1.61%, whereas the winner of the competition shows 59.10% weighted APCER at 0.46% BPCER. When considering individual PAs ACPER, attacks by cadaver eyes and fake eyes are easier to detect, whereas attacks by cosmetic contacts are challenging attacks to detect. The

Table 2.6: Description of the test set of the LivDet-iris-2020 dataset. It includes the number of images in each category and the sensor used to capture them.

Classes	**Presentation Attack Instruments**	**Sample Count**	**Sensor**
Bonafide	-	5,331	LG 4000, AD 100, Iris ID iCAM7000
PA	Printed Eyes	1,049	Iris ID iCAM7000
PA	Cosmetic Contact Lens	4,336	LG 4000, AD 100, Iris ID iCAM7000
PA	Kindle Display	81	Iris ID iCAM7000
PA	Fake/Prosthetic/Printed Eyes with Add-ons	541	Iris ID iCAM7000
PA	Cadaver Iris	1,094	IriTech IriShield

Table 2.7: D-NetPAD performance reported in terms of APCER and BPCER on the LivDet-2020 dataset. The results also include APCER on the individual type of PAs. The method is compared with the winners of the LivDet-2020 competition. Here, **PE** is Printed Eyes; **CL** is Cosmetic Contact Lens; **ED** is Electronic Display; **F/P** is Fake/Prosthetic/Printed Eyes with Add-ons; and **CI** is Cadaver Iris.

Algorithms	**APCER**					**Overall Performance**		**ACER**
	PE	**CL**	**ED**	**F/P**	**CI**	**APCER**$_{avg.}$	**BPCER**	
USACH/TOC	23.64	66.01	9.87	25.69	86.10	59.10	0.46	29.78
FraunhoferIGD	14.87	72.80	53.08	19.04	0	48.68	11.59	30.14
Competitor-3	72.64	43.68	83.95	73.19	89.85	57.8	40.31	49.06
D-NetPAD	2.38	3.85	1.23	0.18	0.18	2.76	1.61	2.18

high ACPER in cosmetic contact PAs is also due to the unseen types of cosmetic contact used in the testing set.

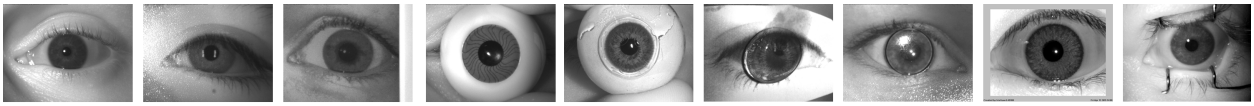


Figure 2.6: Sample images of bonafide and PAs (print, kindle display, artificial eye, cosmetic contact, and cadaver eyes) from the LivDet-2020 dataset.

2.4.4 GCT5 and GCT6 Datasets: Description and Results

The Government Control Test (GCT) 5 and 6 are also proprietary datasets collected under the IARPA Odin program (Presentation Attack Detection) [2]. Table 2.8 provides the performance of D-NetPAD on both datasets along with the number of bonafide and PA images present in the

Table 2.8: D-NetPAD performance in terms of TDR at 0.2% FDR on the GCT5 and GCT6 datasets. Table also provides information about training and testing data along with base architecture used in both models.

Model (Base Architecture)	DenseNet161	DenseNet201
Train	Combined Dataset, GCT3 (2,963 Bonafide, 352 PAs), Cross-sensor (9,606 Bonafide, 922 PAs), Warsaw Postmortem (2,400 PAs)	Combined Dataset, GCT3 (2,963 Bonafide, 352 PAs), Cross-sensor (9,606 Bonafide, 922 PAs), Warsaw Postmortem (2,400 PAs), GCT4 (337 Bonafide, 332 PAs), LivDet-2020 (5,315 Bonafide, 7,101 PAs), GCT6 Train (1,457 Bonafide, 598 PAs)
Test	GCT5 (1,354 Bonafide, 206 PAs)	GCT6 Test (3,112 Bonafide, 392 PAs)
TDR (%) @ 0.2% FDR	95.63	99.23
Threshold	0.3839	0.4671

training and testing sets. The base architecture of the GCT5 model is DenseNet161, whereas, for GCT6, it is DenseNet201.

2.4.4.1 Failure Analysis

Subsequently, we analyze the failure cases that occurred on the GCT5 data. There are ten misclassifications, one in bonafide and nine in case of PA images. Figure 2.7 shows all the failure cases along with their Grad-CAM heatmaps. The Grad-CAM heatmaps provide information about the correctness of the iris segmentation and high priority regions network utilize to make the decision. There occur no segmentation errors in the failure cases. The misclassified bonafide image contains circular artifacts in the iris region may be due to a soft transparent contact lens or positioning of light sources. The circular artifact resembles cosmetic contact images and results in misclassification. In the case of PA misclassifications, three types of cosmetic contacts contribute to the misclassification, two are unknown cosmetic contacts (m6-009-0007-A77-1 and m6-009-0011-F40-1), and one is known cosmetic contact Extreme-SFX-Intrigue Brown (m6-009-0005-B44-1). The Extreme-SFX-Intrigue Brown cosmetic contacts are misclassified in the GCT3 results as well. The training of the GCT5 model contains only a few images of the specified contact lens. In two misclassified PA images (m6-009-0011-F40-1), artifacts due to the face mask cause misclassification.

We also perform matching of entire PA images with their corresponding bonafide to understand

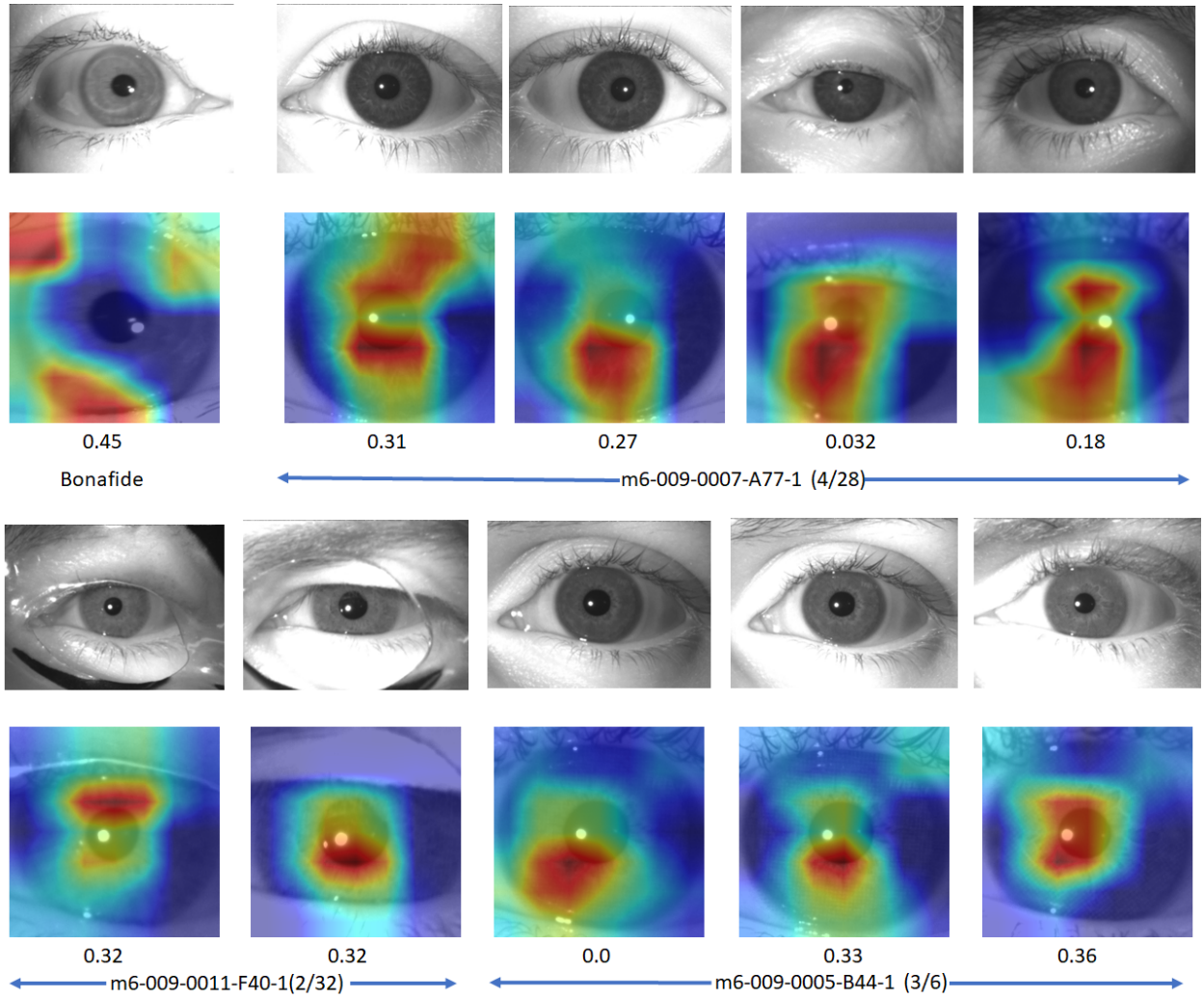


Figure 2.7: Failure cases on the GCT5 dataset. The first image is a bonafide misclassified bonafide image, and the other images are misclassified PA images. Three types of cosmetic contacts get misclassified: m6-009-0007-A77-1, m6-009-0011-F40-1, and m6-009-0005-B44-1. The threshold is 0.38.

the portion of the underlying bonafide pattern present in the PA samples. We use a commercially available high-performing iris matcher called VeriEye. Its threshold is 60 for 0.001% FMR. We observe that 82.52% (170/206) of PA images are higher than the 60 revealing an underlying iris pattern helpful for matching. Figure 2.8 shows a histogram of VeriEye match scores corresponding to correctly classified and misclassified PA images. Match scores of misclassified PA images are high (more than 100) and lie in the right-sided tail of the histogram. It indicates that these cosmetic contacts do not obscure underlying iris patterns and result in misclassification. Figure

2.9 shows misclassified PA images along with their bonafide and corresponding match scores. All misclassified PA images show higher match scores. We also plot the histogram of match scores (VeriEye) corresponding to different types of cosmetic contacts (Figure 2.10). Most misclassifications occurred on m6-009-0005-B44 cosmetic contact PA images (3/6) show in brown color in Figure 2.10, second major misclassifications occur on m6-009-0007-A77 cosmetic contacts (4/28) show in lime green color, and another cosmetic contact m6-009-0011-F40 (2/32) is next misclassified cosmetic contact show in orange color. All these cosmetic contacts have high match scores revealing underlying iris patterns.

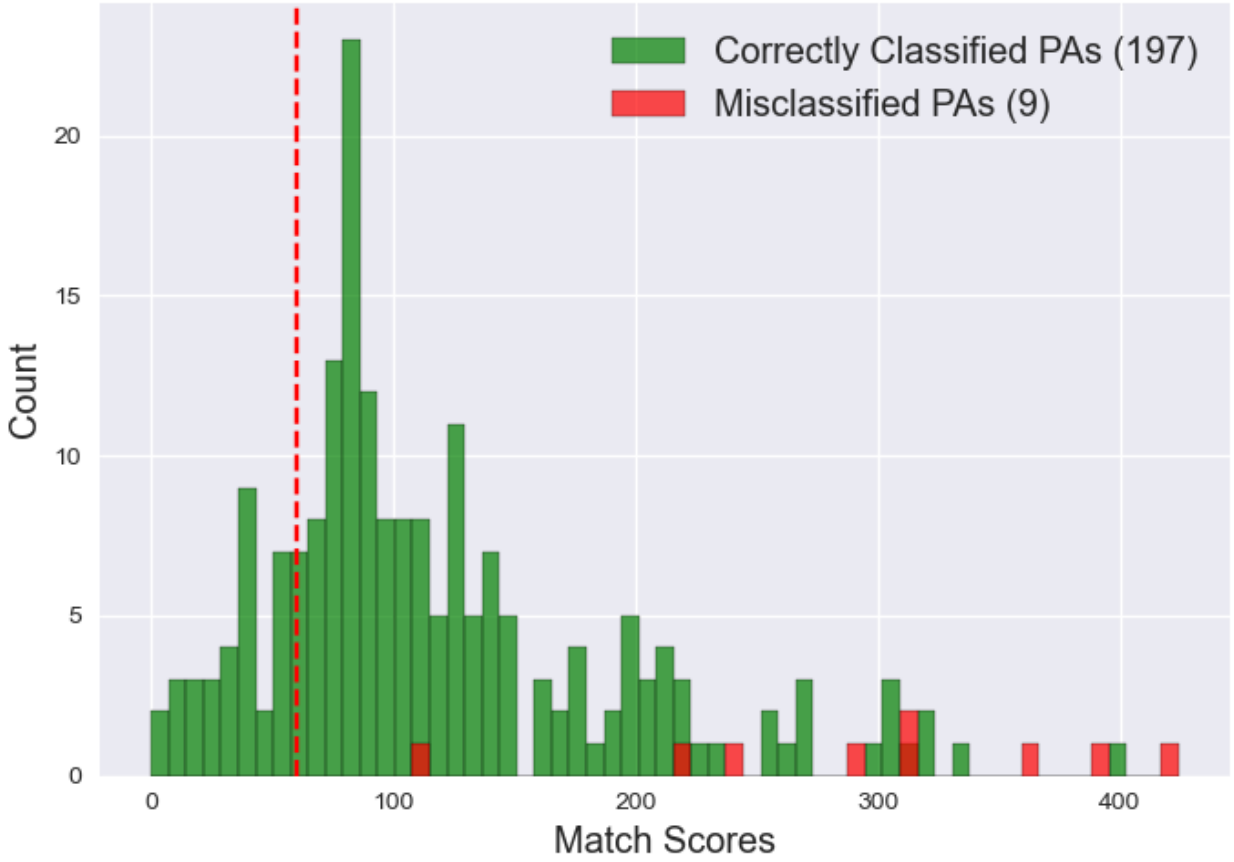


Figure 2.8: Histogram of VeriEye match scores corresponding to correctly classified and misclassified PA images when match with their bonafide images on the GCT5 data.

We perform a similar analysis on the Combined dataset test data. Figure 2.11 shows a histogram of VeriEye match scores corresponding to correctly classified and misclassified PA images. Here, we observe that 47.16% (75/159) of PA images are higher than the 60 (0.001% FMR). There are five

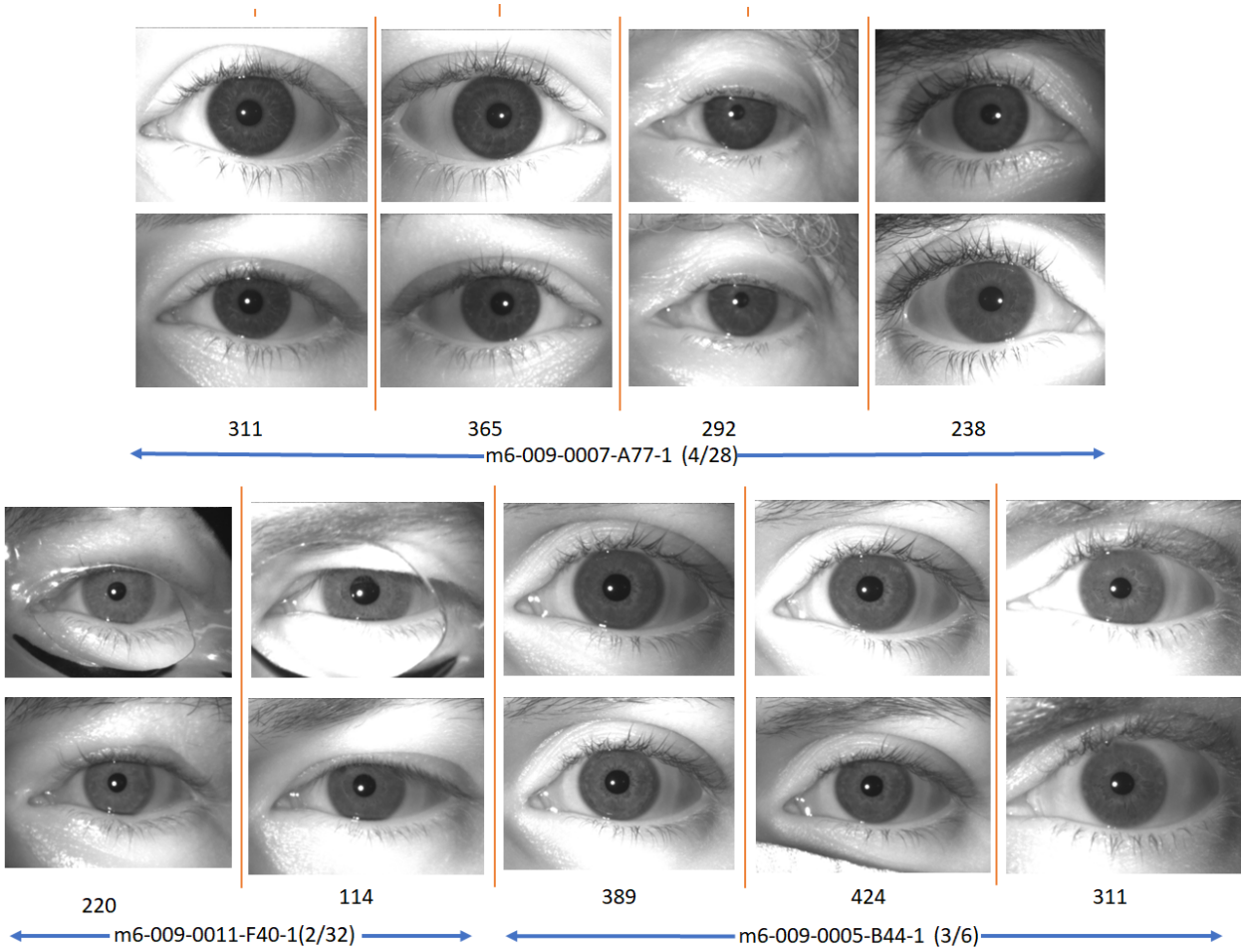


Figure 2.9: Misclassified PA images (bottom row) along with their bonafide images (top row) and their matching score using VeriEye commercial iris matcher.

misclassifications on PA images, and 4 out of 5 match scores are higher than the 60 threshold. We also plot a histogram of match scores corresponding to different types of cosmetic contacts (Figure 2.12), and misclassifications mainly occur on m60090005B44 cosmetic contact (5/23) shown in brown color in Figure 2.12.

2.5 Explainability analysis

2.5.1 Visualization Analysis

We visualize the results of the D-NetPAD using t-Distributed Stochastic Neighbor Embedding (t-SNE) [280] plots and Gradient-weighted Class Activation Mapping (Grad-CAM) heatmaps. We

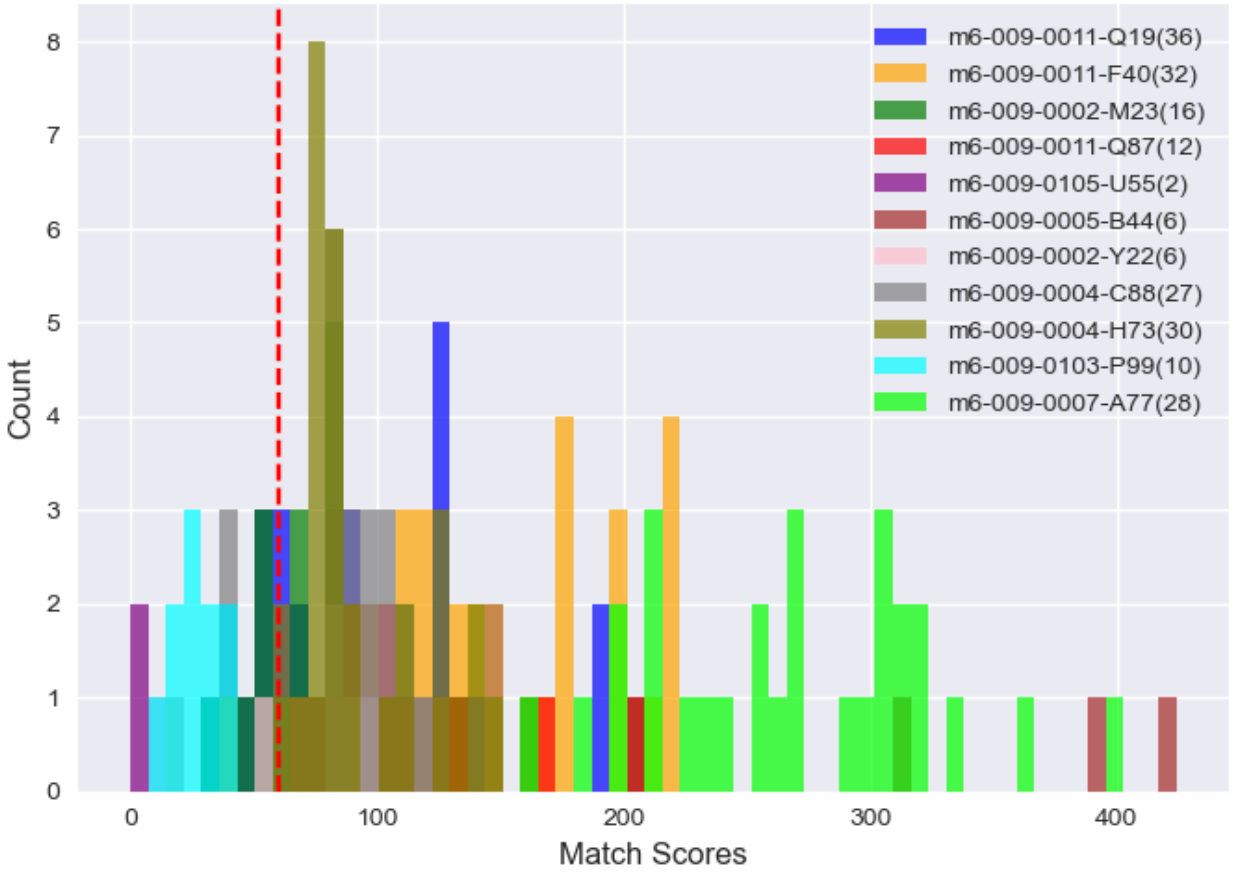


Figure 2.10: Histogram of VeriEye match scores corresponding to different cosmetic contact PA types on the GCT5 data.

utilize the D-NetPAD model trained on the training set of the Combined dataset for this purpose, and use the samples in the JHU-APL03 test set to generate these visualizations. The t-SNE helps in visualizing the features extracted from the D-NetPAD. It reduces the high-dimensional features extracted from the D-NetPAD to a lower dimension (two in our case), which are then used to construct a scatter plot. The architecture of the D-NetPAD consists of four Dense blocks. We capture the high-dimensional features at the end of each Dense block for visualization (Figure 2.13). For instance, the feature set captured at the end of Dense block 4 has a size of $1 \times 1024 \times 7 \times 7$, which is flattened to $1 \times 50{,}176$. The 50,176-dimensional row vector is then reduced to a two-dimension vector. We draw three key observations from these plots:

1. The distributions of bonafide, artificial eye and cosmetic contact features overlap after the initial Dense blocks, but separate for the later Dense blocks. As the depth of the network increases, the

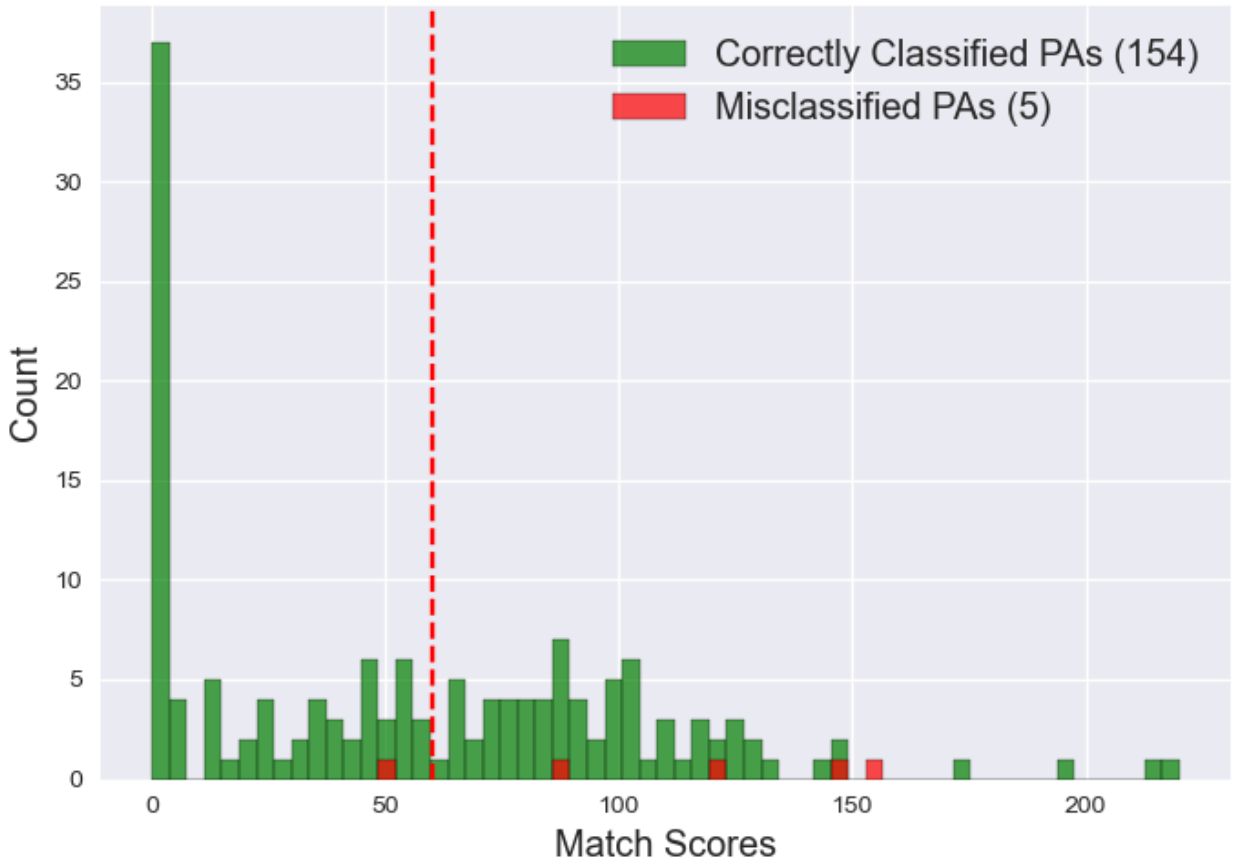


Figure 2.11: Histogram of VeriEye match scores corresponding to correctly classified and misclassified PA images when match with their bonafide images on the GCT3 data.

features of different categories are better separated. This substantiates the high performance of D-NetPAD (Table 2.2).

2. The features of different categories are sufficiently discriminated at the end of Dense Block 4, which justifies the use of four Dense blocks in the architecture as opposed to three in [301].

3. The plots shows two bonafide clusters which correspond to the left and right eyes. The left and right irides exhibit differences due to the orientation of upper and lower eyelids, location of specular reflection, the relative position of pupil center to iris center, and background illumination variation. The D-NetPAD captures these variations in its features.

We further visualize the CNN activations using the Grad-CAM [245] heatmaps. The Grad-CAM produces a coarse localization map highlighting the salient regions in an image that were used by the network to generate its inference. These are regions that produce high activations in

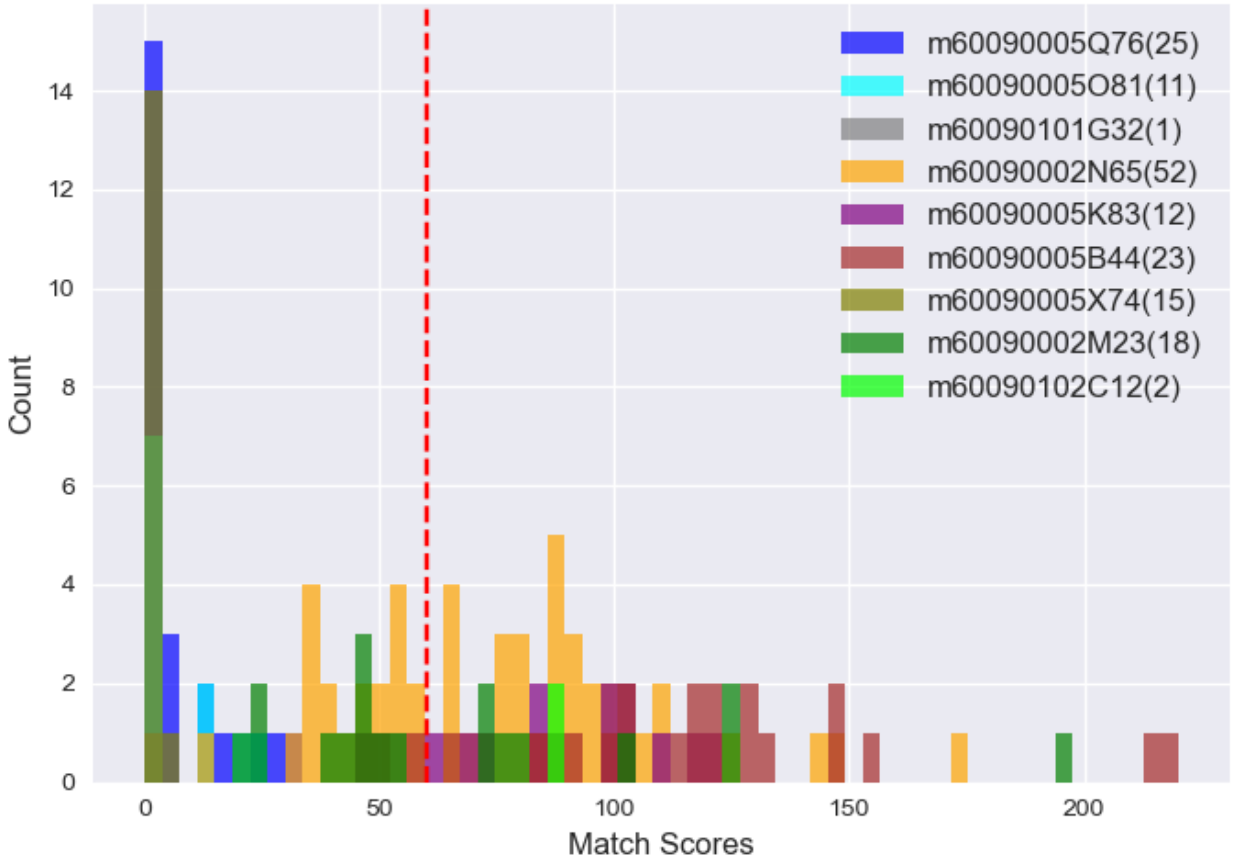


Figure 2.12: Histogram of VeriEye match scores corresponding to different cosmetic contact PA types on the GCT3 data.

the neural network. It is estimated using the gradient of a loss function, which backpropagates through the convolutional layers to the input image [245]. Figure 2.14 presents the CNN activation heatmaps on bonafide, artificial eye, and cosmetic contact images taken from the JHU-APL03 test set. The last column represents the average heatmaps of each category considering the entire test set. The red regions indicate high activation, whereas the blue regions represent low activation. The first row of Figure 2.14 shows the heatmap of bonafide sample images along with the average bonafide heatmap, where the high activation region is at the pupillary zone of the iris pattern. The second row of Figure 2.14 corresponds to the heatmap of artificial eye images, where the focus seems to be mainly on the left and right sub-regions of the iris. The last row shows the heatmaps of cosmetic contact images, where the lower sub-region of the iris pattern is focused. **The average heatmaps show the distinctive regions of focus in each category, which helps in**

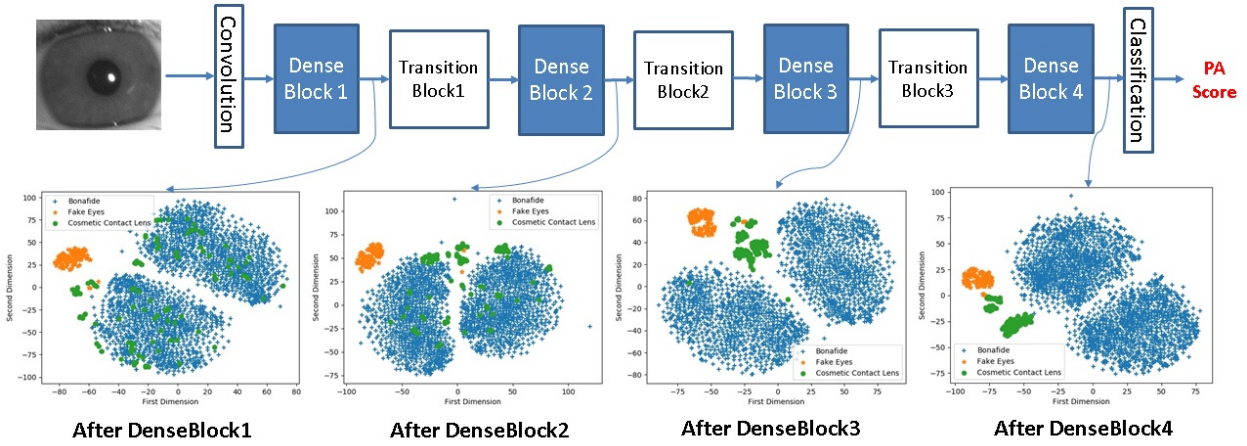


Figure 2.13: The architecture of D-NetPAD consists of four Dense blocks. We capture the features at the end of each Dense block, which are then visualized using t-sne plots (shown below each Dense block). The two-dimensional features of bonafide, artificial eyes, and cosmetic contacts overlap in the initial layers, but get separated in the last layer. The two blue clusters in each category correspond to the left and right eyes.

discriminating bonafide from PAs. We quantify the results of Grad-CAM by training a network with Grad-CAM heatmaps corresponding to three categories to show the region's distinctiveness in each category. We utilize DenseNet121 architecture as a backbone architecture and perform training on randomly selected 60% of the JHU-APL03 heatmaps from the Combined dataset. We repeated the experiments five times, and the top-1 accuracy is 97.90% ± 0.001. The performance validates our claim that D-NetPAD focuses on distinctive regions for each category.

2.5.2 Spatial Frequency Analysis

The iris is a highly textured pattern exhibiting numerous spatial frequencies. To understand what frequencies the D-NetPAD model has learned and how it impacts iris PAD performance, we perform a spatial frequency analysis on the D-NetPAD model. We attain the objective with the assumption that the performance of the model only gets affected by the manipulation of learned frequencies. We start by manipulating higher frequencies for two reasons. First, when we visually examine low- and high-pass filtered images (Figure 2.15), it is observed that a high-pass filter (suppression of low frequencies) considerably obscures the iris pattern. Second, deep learning-

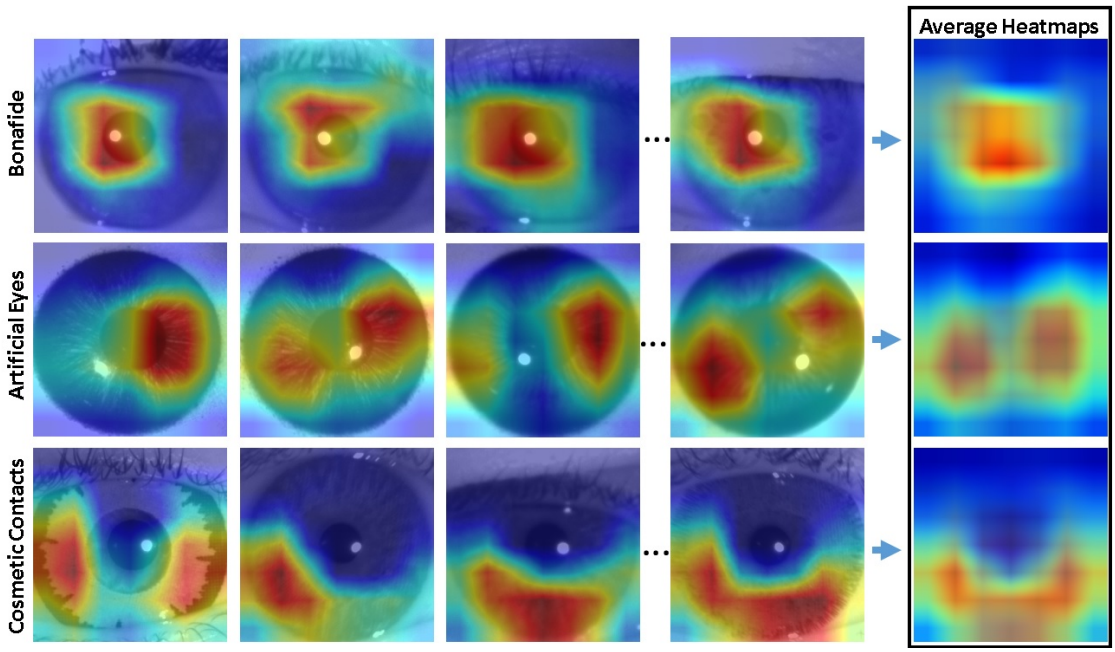


Figure 2.14: Grad-CAM [245] heatmaps corresponding to bonafide (first row), artificial eye (second row), and cosmetic contact (last row). The last column represents the average heatmaps of each category. The heatmaps represent focused regions of the image by the D-NetPAD algorithm. Red-colored regions represent highly focused regions by the D-NetPAD, whereas blue regions represent low priority ones.

based models learn low frequencies first (initial epochs) and then high frequencies (later epochs) in the training process [213, 299]. In other words, the number of weight parameters contributing towards expressing low frequencies is larger than the one expressing high frequencies [213]. Due to this, a small manipulation in low frequencies results in large shifts in performance. In the case of high frequencies, the more the architecture learns high frequencies, the more it tunes its parameters towards learning the intricacies of the training images, which may cause overfitting. So, learning of high frequencies determines the effectiveness of model-fitting on the training data (i.e., efficiently fit or overfit).

For high frequency suppression, we use a low-pass filter with various cutoff frequencies. Cutoff frequency represents a radius from the center in the Fourier transforms (second row of Figure 2.15). A low-pass filter allows frequencies below the cutoff frequency and attenuates higher frequencies. Figure 2.17 shows the performance of the D-NetPAD model as well as the VGG19 and ResNet101 models on various low-pass filter cutoff frequencies. We use the train and test set of the Combined

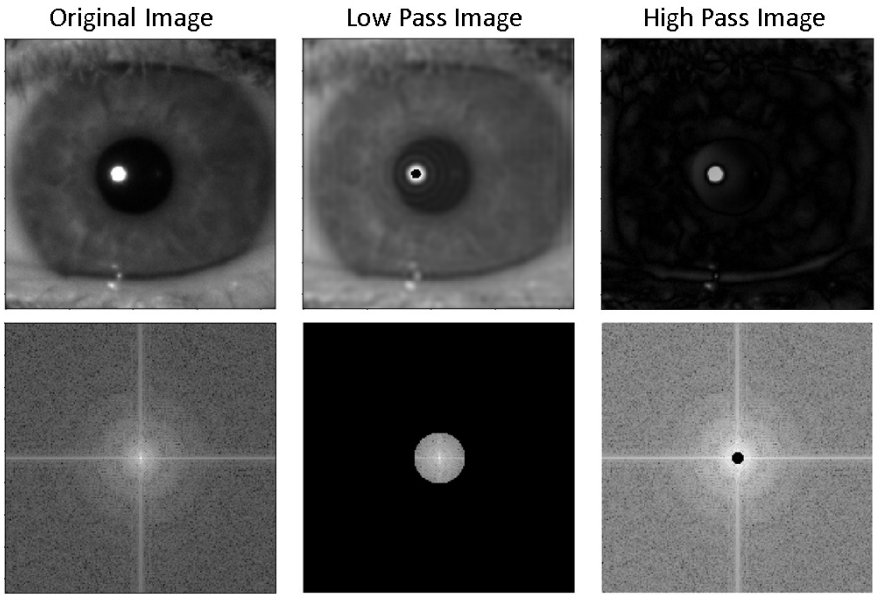


Figure 2.15: Frequency analysis of an input iris (bonafide or PA) image. In the first row, the left-most image is the original image, the center image is a low-pass filtered image with a cutoff frequency of 20 (higher frequencies are suppressed), and the right-most is a high-pass filtered image with a cutoff frequency of 5 (lower frequencies are suppressed). The second row represents their corresponding fourier transforms.

dataset for the experiments. The manipulation is only applied over the test images. There are two noteworthy observations. First, D-NetPAD shows a relatively lower drop in performance compared to VGG19 and ResNet101 models. Second, the performance of the D-NetPAD model becomes steady beyond the 30 cutoff frequency, which implies that the model has not overfitted to higher frequencies beyond 30. Beyond a cutoff frequency of 60, the performance becomes constant implying that it has not learned from frequencies beyond 60. Another way of manipulating high frequencies is their addition to the input images, which we did by contaminating input images with salt and pepper noise. We also analyze the models when Gaussian noise (noise values are Gaussian-distributed) is added to the input images. Figure 2.16 shows an example of an input image subject to high-frequency manipulation, (b) - (e), and the addition of Gaussian noise, (f). The performance is measured using a relative decrease in TDR (%) at 0.2% FDR. Table 2.9 provides the results of VGG19, ResNet101, and D-NetPAD architectures when input images are manipulated.

The D-NetPAD model shows a lower decrease in TDRs compared to VGG19 and ResNet101

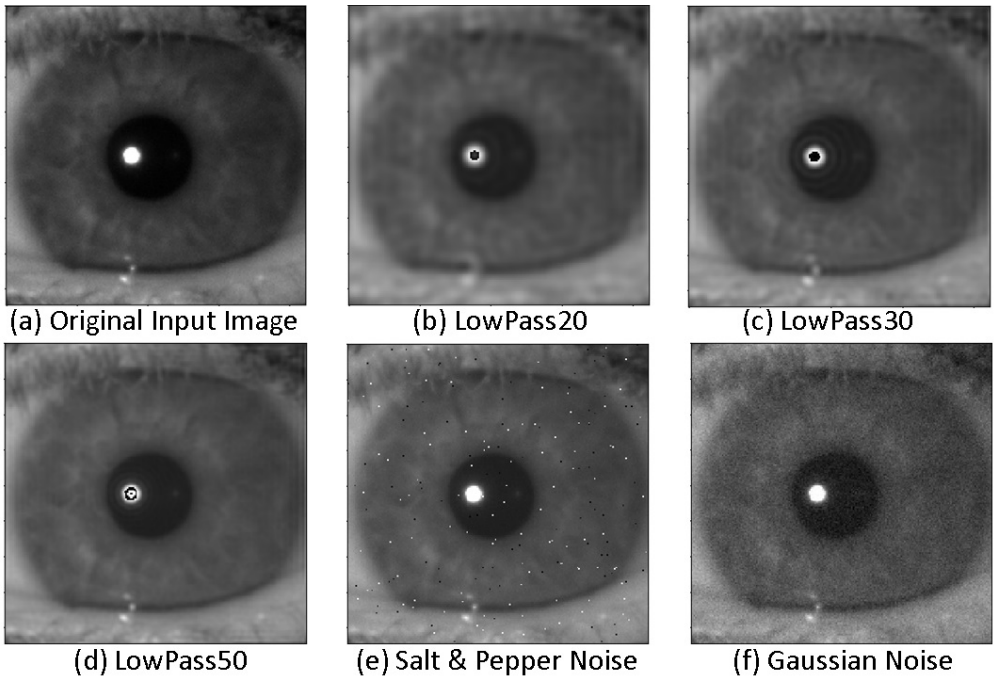


Figure 2.16: Different manipulations applied over the original input image (first image): low-pass filtered images with 20, 30, and 50 cutoff frequencies, additive salt and pepper noise, and additive Gaussian noise. Only test images are subject to these manipulations.

models when higher frequencies in input images are manipulated (either by suppression or addition). The VGG19 and ResNet101 models have a large number of trainable parameters that result in the overfitting of these models to the training data. The overfitted models learn higher frequencies considerably well and, therefore, are more sensitive towards them. On the contrary, efficient learning of frequencies by the D-NetPAD makes it more robust towards manipulations to the high frequencies and also substantiates its generalizability across PAs, sensors, and datasets. Gaussian noise randomly affects both lower and higher frequencies, resulting in a higher drop in performance of all the networks, including D-NetPAD.

2.6 Deployment of D-NetPAD on Desktop and Mobile

We deploy the proposed D-NetPAD model on desktop as well as mobile platforms. In the desktop version, we are capturing iris images from the iCAM7000 iris sensor. The configuration of the GPU used in the desktop is Nvidia GeForce GTX 1070 with 8GB RAM. The D-NetPAD took 0.037 sec to process a single image. Figure 2.18 shows screenshots of the desktop application.

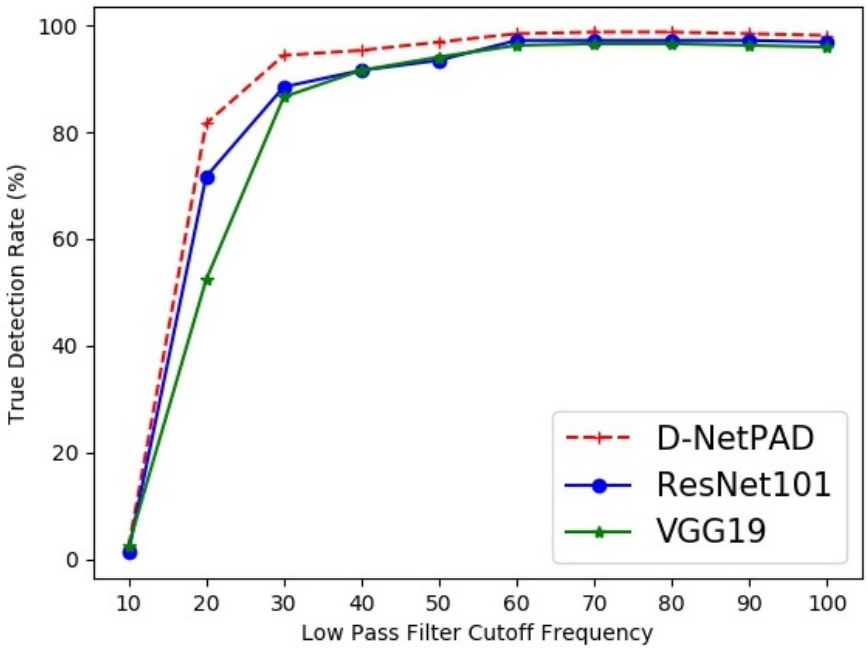


Figure 2.17: The plot of TDR (%) @ 0.2% FDR against low-pass filter cutoff frequencies. Note the cutoff frequency beyond which the performance of D-NetPAD becomes stable (30 in this case). This cutoff frequency indicates that the D-NetPAD has not learned frequencies beyond this cutoff frequency. The performance steadiness of D-NetPAD is better than VGG19 and ResNet101.

Table 2.9: Results (TDR and a relative decrease in TDR) for VGG19, ResNet101, and D-NetPAD models, when high frequencies are manipulated or Gaussian noise is applied to the input test images.

Input Test Images	VGG19		ResNet101		D-NetPAD	
	TDR(%)@ 0.2% FDR	Relative Decrease TDR (%)	TDR(%)@ 0.2% FDR	Relative Decrease TDR (%)	TDR(%)@ 0.2% FDR	Relative Decrease TDR (%)
Original Images	96.26	-	96.88	-	**98.58**	-
LowPass20 (Suppess high freq.)	52.33	45.63	71.65	26.04	81.61	**17.21**
LowPass30 (Suppress high freq.)	86.60	10.03	88.47	8.68	94.39	**4.25**
LowPass50 (Suppress high freq.)	94.08	2.26	93.45	3.54	96.88	**1.72**
Salt & Pepper (Add high freq.)	74.14	22.97	68.22	29.58	80.99	**17.84**
Gaussian Noise	56.07	41.75	62.61	**35.37**	59.19	39.95

For mobile version, we capture both iris images (left and right) from the IriShield BK2121U binocular sensor. We utilize DenseNet201 and MobileNetv2 architectures for mobile deployment.

Table 2.10: Description of two architectures used to detect iris PAs at the mobile platform along with their training data and computational efficiency.

Training Data	Architecture	Input Sensor	Time Taken (secs)	Deploy Environment
Clarkson: 283 Bonafide, 183 Cosmetic Contacts, 1,131 Print ST5: 216 Bonafide, ST6: 524 Bonafide, LivDet-2017_WVU: 702 Bonafide, 2,806 Print, 701 Cosmetic Contacts	DenseNet201	IriShield BK2121U	0.52	Google Pixel 2 Octa-core, 4GB RAM
	MobileNetv2	IriShield BK2121U	0.05	

The training data use to train the models are collected by Clarkson University, taken from Self-test5 and Self-test6 collection, and from the LivDet-Iris-2017 WVU subset. The training images are from the IriShield BK2121U iris sensor except for the images from the LivDet-Iris-2017 WVU subset, which is from IriShield MK2120U sensor. Table 2.10 provides the details of the training data. There are three options for the model compression for mobile devices in PyTorch library:

1. Dynamic quantization: weights are quantized ahead of time, but the activations are dynamically quantized during inference

2. Static quantization: weights quantized, activations quantized, calibration required post-training

3. Quantization aware training: weights quantized, activations quantized, quantization numeric modeled during training

Dynamic quantization reduces the time complexity by a small margin without affecting the performance. The static quantization reduces the time complexity but also reduces the performance. It also requires a calibration using the training data. The third solution requires re-training of the model. Currently, we are using dynamic quantization for compression of the model to deploy at mobile platform. The mobile platform we use is Google Pixel 2 and the time taken by DenseNet201 is 0.52 sec and by MobileNetv2 is 0.05 sec for processing a single image. The loading time on the mobile platform is 0.67 secs. Figure 2.19 shows screenshots of the mobile application developed.

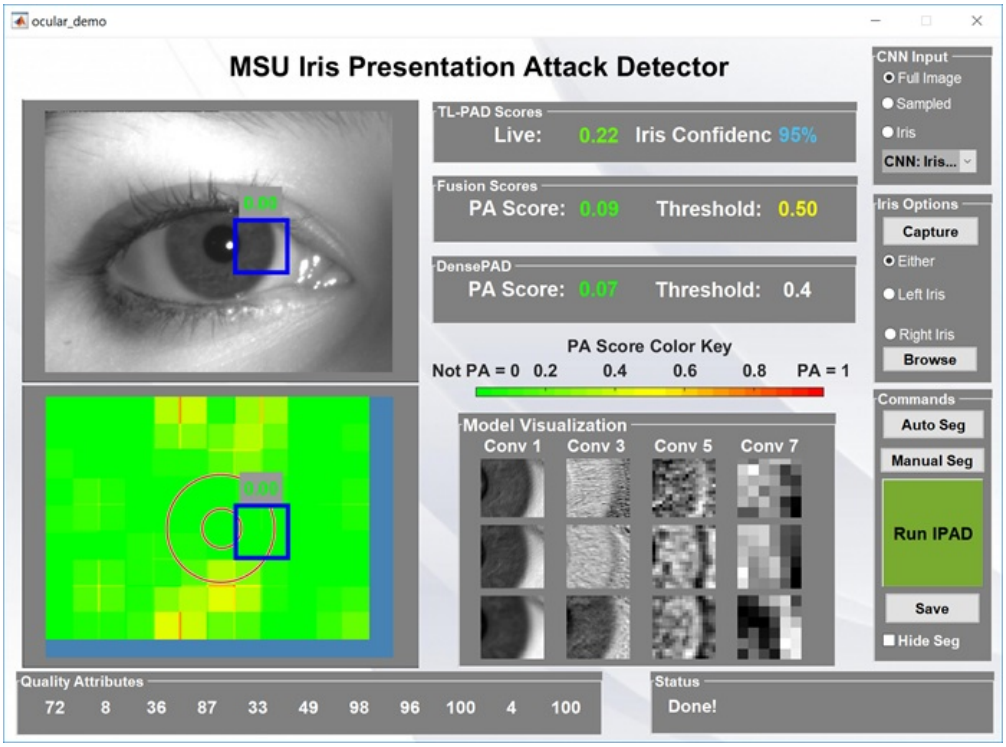


Figure 2.18: Graphical User Interface (GUI) for three iris PA detectors developed by MSU which includes TL-PAD [46], Fusion Method [114] and D-NetPAD [249]. Patch-wise heatmap and filter-maps shown at the bottom of GUI are corresponds to the Fusion Method.

2.7 Conclusion

We propose an effective and robust software-based iris PA detector called D-NetPAD. The D-NetPAD exploits the architectural benefits of DenseNet121. Experiments are performed on five datasets to help assess its effectiveness. The test sets of these datasets correspond to cross-PA, cross-sensor, and cross-dataset scenarios which measure the robustness of the D-NetPAD. We further explained the performance of the D-NetPAD using t-SNE plots, Grad-CAM heatmaps and frequency analysis.

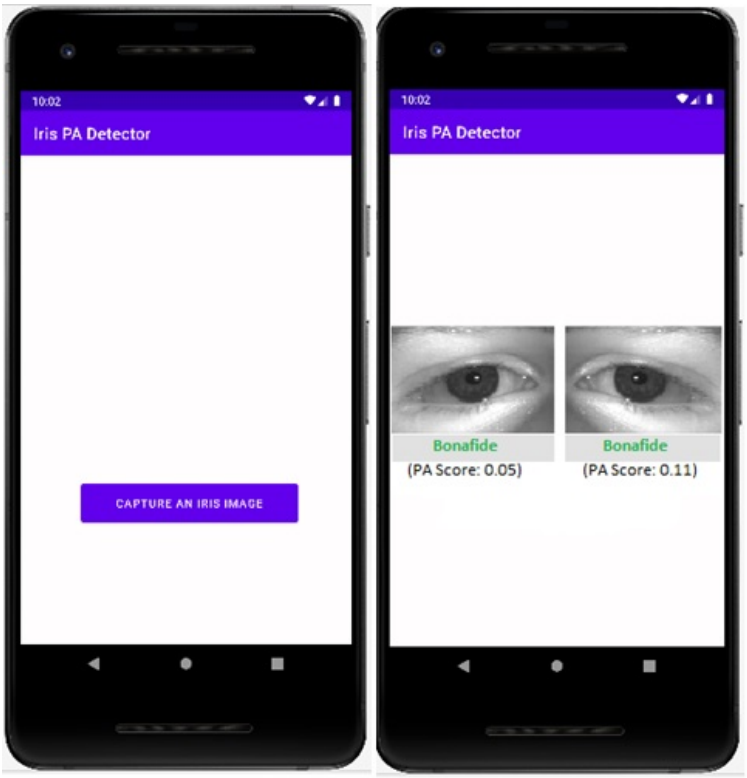


Figure 2.19: Screenshots of Iris PA Detector app on Google Pixel 2. The first image shows the screen on the opening of the app. The second image shows the results after capturing iris images from IriShield USB BK2021U sensor.

CHAPTER 3

IRIS PRESENTATION ATTACK DETECTION USING VISIBLE SPECTRUM VIDEO

3.1 Introduction

In this chapter, we present another iris presentation attack detection (PAD) method which utilizes visible (VIS) spectrum scene video captured from a webcam. Here, the scene refers to the field-of-view of the webcam mounted over the iris sensor (Figure 3.3) capturing user interaction with the sensor. The scene video provides ancillary information such as human posture, actions, objects, human-object interactions, and their temporal changes. Our aim is to extract some of these cues from the multiple frames of the scene video using deep learning techniques. The capturing of the video in the VIS spectrum provides complementary information to the iris image captured in the NIR spectrum (conventionally used in iris recognition). The use of a simple webcam makes the acquisition process cheaper and convenient for users.

The key contributions of the work are:

1. We propose a multi-frame analysis approach for detecting iris PAs from the scene video (VIS spectrum) which seamlessly incorporated into the existing NIR image-based iris recognition systems.

2. We develop various spatial-temporal feature extraction techniques for analyzing the scene.

3. We collect a dataset, Iris Presentation Attacks Video (IPV), consists of 672 iris bonafide and PA videos from 121 subjects and experiments are performed under three scenarios (intra-session, cross-session, and cross-attack).

4. We extend the multi-frame analysis approach for the detection of face PAs and experiments are performed on three publicly available face PA datasets: SiW [165], SiW-M [166] and OULU-NPU [33]. Cross-modality experiments are also performed.

Table 3.1: Description of video-based passive iris PA detection techniques.

Authors	**Hardware and Imaging**	**Algorithmic Details**
Villalbos-Castaldi and Suaste-Góme, 2014 [285]	IR video from custom imaging apparatus	Pupil dynamic features (hippus)
Kiran *et al.*, 2015 [217]	VIS video from iPhone 5S and Nokia Lumia 1020 and NIR images	Laplacian pyramids decomposition followed by frequency responses at different orientations
Kiran *et al.*, 2015 [214]	VIS video from iPhone 5S and Nokia Lumia 1020	Enhanced eulerian video magnification (EVM)
Thavalengal *et al.*, 2016 [265]	VIS and NIR videos from custom mobile device	Multi-spectral features and multi-frame pupil localization
Our Method	VIS video from webcam and NIR images	Various variant of deep features to capture spatial-temporal information

5. We also interpret the PA detection results using Grad-CAM [245] heatmaps. The Grad-CAM heatmap highlights the salient regions in the video that were used by the network to generate the inference.

The rest of the chapter is organized as follows. Section 3.2 discusses the existing work for detecting iris and face PAs. Section 3.3 gives the details of the proposed approach. Section 3.4 describes the datasets used for the experiments. Section 3.5 provides the experimental setup and results. Section 3.6 provides a detailed analysis of the results. Finally, Section 3.7 concludes the chapter.

3.2 Related Work

A brief survey on software and hardware-based iris presentation attack detection (PAD) techniques is provided in Section 2.2. Table 3.1 describes various video-based iris PA detection techniques which requires no stimulation (passive). These techniques are closely related to our acquisition setup.

In the face modality, there are several existing methods that focus on cues from the scene or context. Kim *et al.* [140] detect liveness of a face by combining similarity score of background between reference and input image (region without a face and upper body) and background motion index which indicates the amount of motion in the background compared to the foreground.

Anjos and Marcel [17] measure the correlations between the total amount of movement in the face and background regions. Later on, Anjos *et al.* [16] utilize optical flow for estimating foreground/background motion correlation. Pan *et al.* [195] estimate the context information by comparing the difference of regions around fiducial points between a reference scene image and the input image using local binary pattern (LBP) descriptors. The context information is then combined with blinking information for liveness detection. Yan *et al.* [308] combine three cues namely non-rigid motion, face-background consistency, and imaging banding effect for face PAs detection. The non-rigid motion of a face (i.e., blinking) is estimated using low-rank matrix decomposition. Face-background consistency (motion of face with respect to the background) is calculated using GMM-based motion detection, and imaging banding effect (imaging quality defects) is estimated using wavelet decomposition. Komulainen *et al.* [146] detect face PAs by fusing temporal (using MLP classifier) and texture information (using LBP) at the score level using linear logistic regression. In another work, Komulainen *et al.* [145] utilize a cascade of an upper-body and spoofing medium detectors based on the histogram of oriented gradients (HOG) descriptors and linear support vector machines (SVM). Patel *et al.* [199] integrate deep texture features and face movement cues (eye-blinking) for liveness detection. Deep texture features are learned from both aligned facial regions and the entire frame. Apart from these context information based face PA detection algorithms, other techniques can be found in [89, 221]. Various competitions and assessment reports are [34, 52, 178]. A detailed description of various iris and face PAD techniques are published in [172].

The proposed approach is advantageous over the existing solutions in the following ways: (a) use of the entire frame discards the pre-processing routine (iris segmentation, or iris or face detection) and the error introduce by them; (b) generally, the devices used in hardware-based techniques are expensive or inconvenient for users, whereas the webcam used in the proposed approach is cost-effective and non-obtrusive. In contrast with software-based techniques which detect PAs after the acquisition of an image from the sensor, the proposed approach detects anomalies present in the scene simultaneously with the image acquisition; (c) the approach can easily extend to other

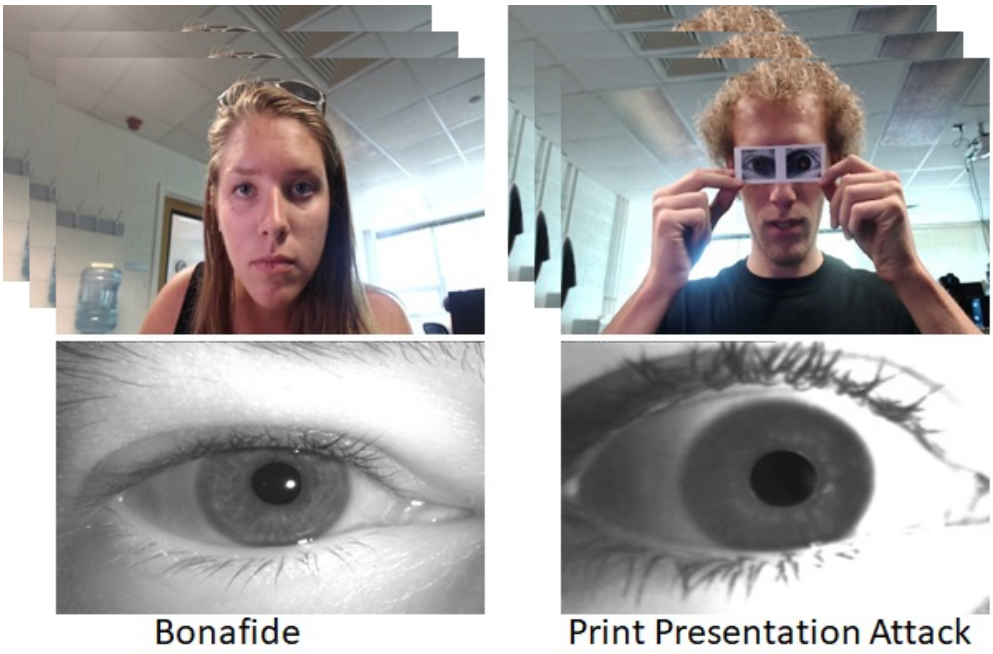


Figure 3.1: Scene video (VIS) and iris image (NIR) of bonafide and PA biometric samples captured by a simple webcam and an iris sensor simultaneously.

Figure 3.2: Different ways of presenting the same attack instrument (paper print) constitute different scenes. These scenes provide different cues for detecting PAs.

modalities (e.g., face or fingerprint) due to the similarities of the way the attacks present (e.g., print attack of the face and iris modalities) as the results show for face PA detection. The approach has a limitation in detecting certain types of presentation attacks, e.g., cosmetic contact lens (iris modality) and face makeup (face modality). It also fails in those acquisition scenarios where scene information is not provided, for instance, OULU-NPU [33] dataset.

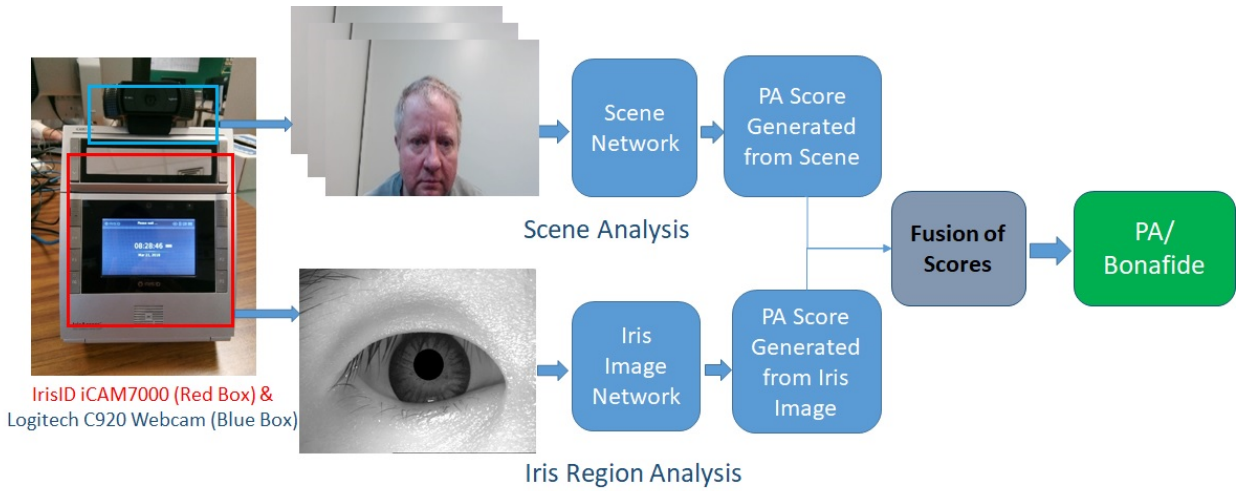


Figure 3.3: The end-to-end architecture of the proposed framework.

3.3 Proposed Method

Figure 3.3 shows the architecture of the proposed framework. Hardware setup consists of a webcam mounted over a standard iris sensor. The webcam captures the scene video, and the iris sensor captures a NIR iris image. Iris image (I) and scene video (V) then undergo different techniques to compute the individual PA scores. A PA score is a confidence score that the given input is a PA. It is in the range [0, 1], where 1 represents high confidence that the input is a PA. To generate a PA score from the iris image (s_I), we adopt the CNN architecture proposed by Hoffman *et al.* [113], though other PA detection techniques can also be used. For computing PA score from the scene video (s_V), we utilize several deep learning techniques (details are given below). Both PA scores are normalized and then averaged to obtain a final PA score by:

$$f(s_V, s_I) = \begin{cases} \text{Bonafide}, & \text{if } \left(\frac{s_V + s_I}{2}\right) \leq T \\ \text{PA}, & \text{otherwise} \end{cases}$$

where, T is the threshold. We propose six different deep learning techniques to extract spatio-temporal information contained in the multiple frames of a scene video and generate a PA (s_V) score. We describe these techniques in the following subsections.

3.3.1 MLP

Firstly, we capture only spatial information from the scene as some of the PAs are strongly associated with the objects present in the scene such as paper print, kindle display, artificial eye, etc. We resize video frames to 80×80 and select 30 equally spaced frames per video. Resized video frames are input into a pre-trained model of Inception-v3 (pre-trained on ImageNet Dataset [73]) to extract CNN features. CNN features are then fed into a multi-layer perceptron (MLP) consisting of two hidden layers with 512 hidden units each and one softmax layer. We also use two dropout layers following each hidden layer with a dropout value of 0.5. The training mini-batch size used is 20. Training stops when there is no further reduction in training loss. During testing, we estimate a final PA score for a video by summing softmax scores obtained from 30 equally spaced frames. The method considers video as a collection of independent frames, exploring only spatial information and ignores the temporal information.

3.3.2 LSTM

To capture temporal information, we feed the same CNN features (Inception-v3) into a two-layered long short-term memory (LSTM) network instead of an MLP network. The first layer has 2048 hidden units, and the second layer is a fully-connected layer with 512 nodes. There is a dropout of 0.5 following the fully connected layer. The mini-batch size and stopping criteria are the same as the previous method. Temporal information is considered in the method, but only at the very end of the network.

3.3.3 LRCN

To capture spatial and temporal information simultaneously, we use a long-term recurrent convolutional network (LRCN) [76], which is an end-to-end trainable architecture. The architecture has a small VGG-16 style network followed by one LSTM layer having 256 nodes and the final softmax layer. The training mini-batch size is 15, and stopping criteria is the same as the previous method.

The method performs training from scratch, which requires a large amount of training data.

3.3.4 C3D

Another way to capture the spatial-temporal information is 3D ConvNet, where the third dimension corresponds to the temporal dynamics. To implement this, we use the architecture (C3D) proposed in [272], which consists of five 3D convolutional layers followed by 3D max-pooling layers, two fully connected layers, and a softmax layer. Unlike [272], we also use two dropout layers following each fully connected layer with a value of 0.5 to prevent over-fitting due to the small training data. The number of filters used for five convolutional layers is 64, 128, 256, 512, and 512 having the same filter size $3 \times 3 \times 3$. Fully connected layers have 4096 hidden units. All max-pooling layers have a kernel size of $2 \times 2 \times 2$ except the first one which has a kernel of size $1 \times 2 \times 2$. The mini-batch size is 15, and stopping criteria is the same as the previous method. Again, due to a large number of parameters and training from scratch, it requires a large amount of training data.

3.3.5 3D ResNeXt-101

This architecture [298] also utilizes 3D convolutional layers, but it is a comparative large architecture pre-trained on Kinetics action recognition dataset [138]. The architecture introduces a new dimension called "cardinality" (the size of the set of transformations), in addition to the depth and width. It consists of 101 convolutional layers depthwise and 32 cardinalities. The input fed into the architecture is 16 equally spaced frames per video resized to 112×112. The mini-batch size is 10, the number of epochs is 100, and the learning rate is 5×10^{-4}.

3.3.6 Two-stream CNN Network

Due to the lack of large training data, we apply another architecture that captures spatial and temporal information separately, but in parallel. The architecture called Two-stream CNN [254] decoupled the scene videos into spatial and temporal information by inputting them into two separate streams

of ConvNets. The final score obtains by averaging softmax scores outputs from the two streams. Details of these two ConvNets are as follows:

3.3.6.1 Spatial ConvNet

The spatial stream performs PA detection utilizing RGB video frames. The backbone architecture is ResNet-101 pre-trained on the ImageNet dataset [73]. Simonyan and Zisserman [254] use a single RGB frame, whereas we select k equally spaced frames from the video and resize them to size 224×224. We then input these frames are into k separate ResNet-101 networks and combine their scores using an aggregation function (S) as:

$$S = \sum_{i=1}^{k} R(F_i; P) \tag{3.3.1}$$

where, $R(F_i; P)$ is the softmax score generated by the network with parameters P and frame F_i as input. The same parameters P are used in all k networks. Subsequently, cross entropy loss is calculated as

$$L(y, S) = -\sum_{i=1}^{C} y_i (S_i - \log \sum_{j=1}^{C} \exp S_j) \tag{3.3.2}$$

where, C is the total number of classes and y_i is the true label of class i. The loss back-propagates through the network and updates the parameters P. Though spatial ConvNet is intended to capture the spatial information, its loss calculation also captures a long-range temporal structure. It is motivated by the concept of Temporal Segment Networks (TSNs) [288] though the TSNs use a sequence of k snippets (set of consecutive frames), we use k frames. The backbone architecture in [288] is the Inception network with Batch Normalization [124], whereas we employ ResNet-101 as a backbone network. Aggregation functions used in [288] are maximum, averaging, and weighted averaging, whereas we use sum as an aggregation function. We aggregate the loss estimated from all the selected frames and then update the parameters. We can update the parameters by using a single frame at a time, which also increases the data used for training but loses the temporal information. This concept is already utilized in the MLP method. We utilize separate networks for individual frames. An alternative option is to feed the frames as multiple channels into the

network, but it increases the number of trainable parameters. C3D and ResNeXt-101 methods used this concept.

We experiment with different numbers of frames per video (k) for training and empirically select $k = 3$. During testing, we select 20 equally spaced frames and combine their corresponding softmax scores to get the final score.

3.3.6.2 Temporal ConvNet

The temporal stream utilizes a stack of optical flows [41] for PA detection. During training, we input 20 optical flow frames from a video into a single ResNet-101 network as multiple channels, where 10 frames correspond to the X-direction, and 10 correspond to the Y-direction [254]. Figure 3.4 shows RGB (first row), X-direction optical flow (middle row), and Y-direction optical flow (last row) frames corresponding to bonafide and PA samples. The optical flow frames are also resized to 224×224. During testing, we randomly select 10 frames and calculate their 10 X-direction and 10 Y-direction optical flow frames to fed into the network. We average 20 softmax scores to get the final decision about the video.

The architecture helped in working with small-sized training data but has a high time complexity to compute the optical flow frames and memory requirement to store them on disk. Both the streams use mini-batch of size 15, the number of epochs is 100, and the learning rate is 0.0005. We empirically select all hyperparameters.

3.4 Datasets

To evaluate iris PA detection, we introduce a proprietary dataset called Iris Presentation Attack Videos (IPV). Existing iris PA video datasets focus solely on the iris region only and does not capture the scene information. This necessitates the collection of a new iris video dataset containing scene information. For face PA detection, we utilize three publicly available datasets: SiW [165], SiW-M [166], and OULU-NPU [33]. Details for all these four datasets are as follows.

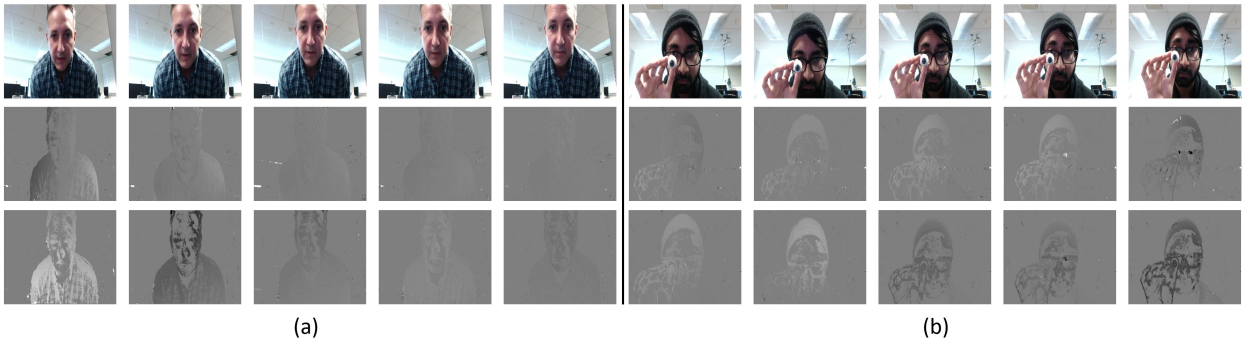


Figure 3.4: Inputs given to the Two-stream CNN network. The top row shows spatial frames, the middle row represents optical flow frames in the X-direction and the bottom row shows optical flow frames in the Y-direction. (a) corresponds to bonafide video frames, and (b) corresponds to PA video frames.

3.4.1 IPV Dataset

We collect the dataset in three sessions with different locations, operators, environments, and timing using a Logitech C920 webcam. Figure 3.3 shows the acquisition setup where a webcam mounts on an IrisID iCAM 7000 sensor. Recording of a video from the webcam starts automatically when the IrisID sensor gives instructions to the user to align the eyes with the IrisID sensor and stops on the capture of an iris image from the IrisID sensor. Videos are approximately 4-5 seconds long with a frame rate of 30 frames/sec. The first session of the dataset collected in lab1 and termed as IPV1. The second session collected in lab2 after five months of the first collection and termed as IPV2. The two labs (lab1 and lab2) are at different locations, thus having different acquisition environments. The third session data collection conducted in the lab1 again after three months and termed as IPV3. Subjects of IPV2 are disjoint from the subjects of IPV1 and IPV3. The IPV1 session data contains videos of only one subject to ensure that PA detection techniques focus on characteristics of bonafide or PA rather than identity information of a user. Different types of PAs and their corresponding number of video collections are paper print (74), artificial eye (156), kindle display (51), funny glasses (166), and mannequin attacks (28). Table 3.2 provides further description of the collected dataset. The dataset has large variations in terms of different PA materials, and different ways to present the PAs. For paper print PAs, we use two different paper types (glossy and matte), and two different types of prints (with and without pupil cut out). We also

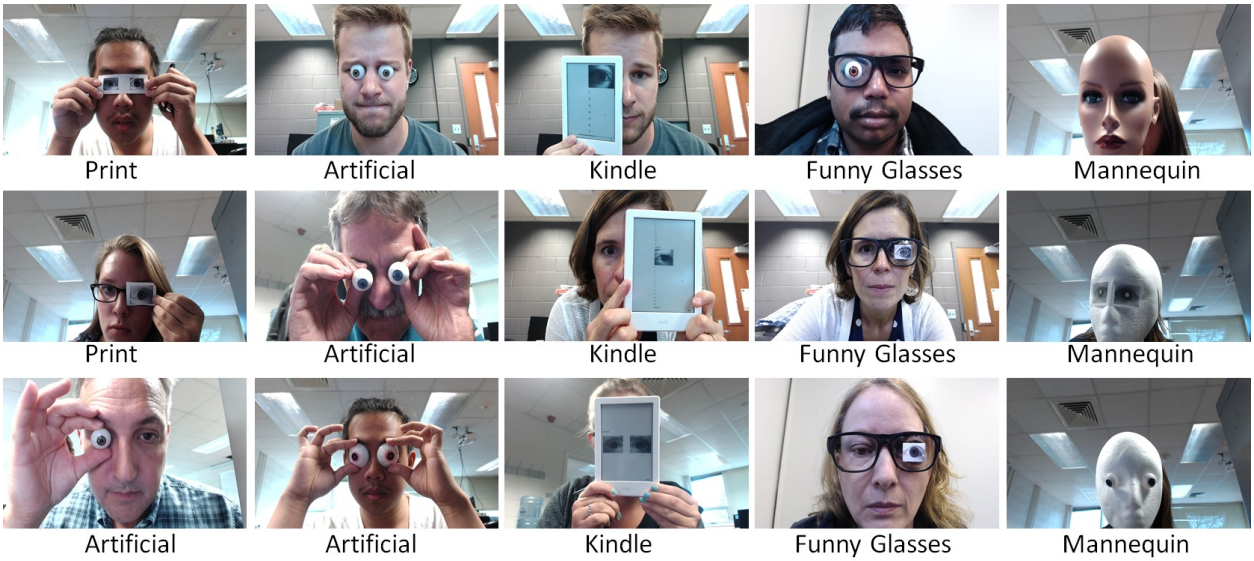


Figure 3.5: Columns show intra-variations among different PAs using a single frame. Paper print PA variations: uses one or both eyes for presenting iris PA. Artificial eye PAs variations: use different materials, e.g., glass, plastic, prosthetic, or rubber eye. Kindle PAs variations: use different sizes and locations of an iris image on the Kindle display. Funny glasses PAs variations: uses plastic or paper print to mount over the funny glasses. Mannequin PAs: use two different materials and print/plastic to mount over them.

use a transparent dome in some paper print PAs to mimic the shape and specular reflections from an eye. Artificial eye PAs contains four different materials (plastic, glass, prosthetic, and rubber). We also create funny glasses PAs, where artificial eye and paper printed PAs mounted over the funny glasses. Mannequin PAs contains two different materials (plastic and polystyrene), mounted with paper print and artificial eyes. More variations introduced in the dataset by alternating the use of one or another eye or both eyes to present a PA. Figure 3.5 shows few variations of the dataset.

3.4.2 SiW Dataset

The Spoof-in-the-Wild (SiW) [165] dataset contains bonafide and spoof videos of 165 subjects. There are 8 bonafide and up to 20 spoof videos from each subject. The dataset is collected in four sessions with different PIE variations. The videos are captured using two high-quality cameras: Canon EOS T6 and Logitech C920 webcam. The videos are 15 seconds in length, 30 fps frame rate, and 1080p HD resolution. The dataset provides two print and four replay video attacks for

Table 3.2: Description of the dataset collected for multi-frame analysis on scene videos captured from a regular webcam.

Session	IPV1	IPV2	IPV3
No. of Bonafide Videos	17	69	111
No. of PA Videos	67	20	388
No. of Subjects	1	80	42
Type of PAs Collected	Paper print, Artificial eye, Kindle display, Mannequin	Funny glasses	Funny glasses, Paper print, Artificial eye, Kindle display
Acquisition Time Period	October 2017	April 2018	August 2018

each subject. For print attack, two quality images (5184×3456 and 1920×1080) are printed using an HP Color LaserJet M652 printer. To generate replay video attacks, four spoof mediums (Samsung Galaxy S8, Apple iPhone 7, Apple iPad Pro, and PC Asus MB168B) are used. Figure 3.7 (second block) shows few samples of the dataset.

3.4.3 SiW-M Dataset

The Spoof-in-the-Wild database with Multiple Attack Types (SiW-M) dataset [166] is built to benchmark the face PA detection algorithms for detecting unseen attacks (cross-attack). There are a total of 1,630 videos of 493 subjects with 13 different spoof attacks. The videos are 5-7 seconds in length, 30 fps frame rate, and 1080p HD resolution. The videos are recorded using a Logitech C920 webcam and a Canon EOS T6 in three sessions. The spoof attacks included in the dataset are 5 3D mask attacks, 3 partial attacks, 3 makeup, one replay, and one print attack. The 3D mask attacks include half mask, silicone, transparent, paper-craft, and mannequin masks. The makeup attacks constitute obfuscation, impersonation, and cosmetic makeup. The partial attacks include funny eye, paper glasses, and partial paper. Figure 3.7 (first block) shows a few samples of the dataset.

3.4.4 OULU-NPU dataset

The OULU-NPU [33] dataset is built to assess the generalizability of face PAD techniques in mobile scenarios. The dataset consists of a total of 4,950 bonafide and PA videos of 55 subjects. The videos were recorded using the front cameras of six mobile devices (Samsung Galaxy S6 edge, HTC Desire EYE, MEIZU X5, ASUS Zenfone Selfie, Sony XPERIA C5 Ultra Dual, and OPPO N3) in three sessions with different illumination and locations. The presentation attacks included in the dataset are print and replay attacks. Print and replay attacks are created using two different printers and display devices respectively. During the capture, special attention is given to avoid the background scene difference between the bonafide and PA videos. So, the print and replay attack videos do not contain the bezels of the screens or edges of the prints. Figure 3.7 (third block) shows few video frame samples of the dataset.

3.5 Experimental Results and Analysis

To analyze the effectiveness of the proposed approaches, we have performed various experiments to detect PAs in iris and face modalities. For the iris modality, we conduct experiments in intra-session, cross-session, and cross-attack scenarios on the IPV dataset. We also conduct a baseline experiment in iris modality, where the complementary nature of the scene cues evaluates with cues obtained from the iris region. For the face modality, we perform face PA detection experiments on SiW [165], SiW-M [166], and OULU-NPU [33] datasets. Finally, we perform cross-modality PA detection experiments where training is on iris PAs and testing on face PAs and vice versa. We reported results in the Average Classification Error Rate (ACER), which is an average of Attack Presentation Classification Error Rate (APCER) and Bonafide Presentation Classification Error Rate (BPCER). APCER is the proportion of PAs samples misclassified as bonafide, whereas BPCER is the proportion of bonafide samples misclassified as PAs.

3.5.1 Iris Modality

To evaluate the proposed approaches, we perform 12 experiments on the IPV dataset in three settings: intra-session, cross-session, and cross-attack scenario. Experiments 01-05 correspond to intra-session, 06 to cross-session, and 07-11 correspond to a cross-attack scenario. One more experiment (exp. 12) performs where the PA score generated from a scene video is fused with the one proposed in [113] (which uses only the iris image) to show the complementary nature of both the cues.

3.5.1.1 Intra-session

In experiments 01-05, we select training and testing data from all three sessions to analyze the intra-session scenario. So, there are 150 videos from each category for training, and the rest used for testing. Table 3.3 provides the details of the selection of videos in each session. Table 3.5 (columns 02-06) presents results of all intra-session experiments. Two-stream CNN is an average fusion of Spatial and Temporal ConvNets. Spatial ConvNet performs the best. LSTM and MLP are also producing comparable ACER. These three methods capture spatial information and work on a pre-trained model. Other methods (LRCN, C3D, and Temporal ConvNet) are not performing well. These models are trained from scratch except Temporal ConvNet, which uses a pre-trained model trained on RGB images instead of optical flow frames. Due to the small size of collected data, training of these methods are prone to over-fitting as there is a large number of trainable parameters in the networks. 3D ResNeXt-101 has the same concept as C3D, except it is a large pre-trained model. As a result, it performs better than the C3D.

3.5.1.2 Cross-session

Experiment 06 aims to analyze the cross-session scenario. In the experiment, training performs on the data collected during IPV03 session, and testing performs on the data collected during IPV01 and IPV02 sessions. Table 3.3 provides further details about the experiment 06. Table 3.5 (columns

07) presents the results of cross-session experiment. This is a difficult test condition as one must account for the variations in data acquisition environment, subject population, and PA generation procedures. The ACER increases for every method. However, MLP, LSTM, 3D ResNeXt-101, and Spatial ConvNet methods manage to perform reasonably well. On the other hand, the lack of training data and adverse test scenario drop the ACER of C3D, LRCN, and Temporal ConvNet methods drastically.

3.5.1.3 Cross-attack

To further evaluate the proposed approaches in a cross-attack scenario, we conduct five more experiments (Exps. 08-12) based on a leave-one-out strategy. The strategy allows methods to train on all types of PAs except one for testing as an unseen attack. Table 3.4 provides the training and testing setup of these experiments. It is an even more difficult testing condition than the cross-session scenario. However, Table 3.5 (columns 08-12) shows reasonably good ACER for alomst all methods except the C3D method and a few PAs. The proposed approaches reliably detect unseen PAs as the method not only captures the characteristics of PA material, but also focus on its presentation, and other contextual information. The 3D ResNeXt-101 method fails to detect unseen artificial eyes and funny glasses attacks, and the LSTM method fails to detect unseen funny glasses attacks. It could be due to the presence of bonafide videos where users wear corrective eyeglasses. The mannequin attack is a difficult unseen attack for the majority of the techniques, whereas paper print is the simplest attack. **Overall, Two-stream CNN method performs the best in the cross-attack scenario.** In another observation, temporal information modeled better in this scenario (compared to intra-session and cross-session), as can be seen from the results of LRCN and Temporal ConvNet. In the leave-one-out strategy, there are more samples for training compared to intra-session and cross-session experimental setup. This also supports our over-fitting hypothesis with C3D, LRCN, and Temporal ConvNet methods for poor results under intra-session and cross-session scenarios.

Table 3.3: Training and testing setup for intra-session (Exp. 01-05) and cross-session (Exp. 06) experiments on the IPV dataset.

Experiments	**Category**	**IPV1**		**IPV2**		**IPV3**	
		Train	**Test**	**Train**	**Test**	**Train**	**Test**
Exp. 01-05	Bonafide	10	7	50	19	90	21
	PA	15	17	10	10	125	219
Exp. 06	Bonafide	0	17	0	69	111	0
	PA	0	32	0	20	111	0

3.5.1.4 Baseline Experiments

From the results obtained from all three scenarios, one can deduce that video of a scene *does contain* cues for detecting PAs in iris modality and it can generalize across unseen attacks. We perform another experiment (Exp. 12) to examine the complementary nature of scene cues regarding cues from iris region. We perform the score-level fusion of cues obtained from the iris region and scene video. The one PA score is obtained from the existing iris PA detection technique [113] trained on BERC-IF dataset [154] using a single iris image and the other PA score from the proposed methods applied over the corresponding scene video. For testing, there are 89 bonafide videos along with their corresponding 178 bonafide iris images (89×2) and 136 PA videos along with their 177 iris images ($41 \times 2 + 95$) where both or either iris image is PA. Table 3.5 (the last column) shows results of the fusion. The ACER calculated using only iris region for PA detection [113] is 10.9%, whereas when combined with scene cues, it reaches 0% (fusion with MLP method), thus demonstrating the complementary nature of the cues provided by the scene video. Figure 3.6 shows ACER of all methods across all experiments for better visualization.

3.5.2 Face Modality

After the successful detection of iris PAs from scene video, we extend its use for detecting face PAs on three publicly available datasets SiW [165], SiW-M [166], and OULU-NPU [33]. SiW [165] and SiW-M [166] datasets do contain scene (contextual) information, whereas in OULU-NPU [33] dataset special attention is given to avoid the scene information. Due to the computational

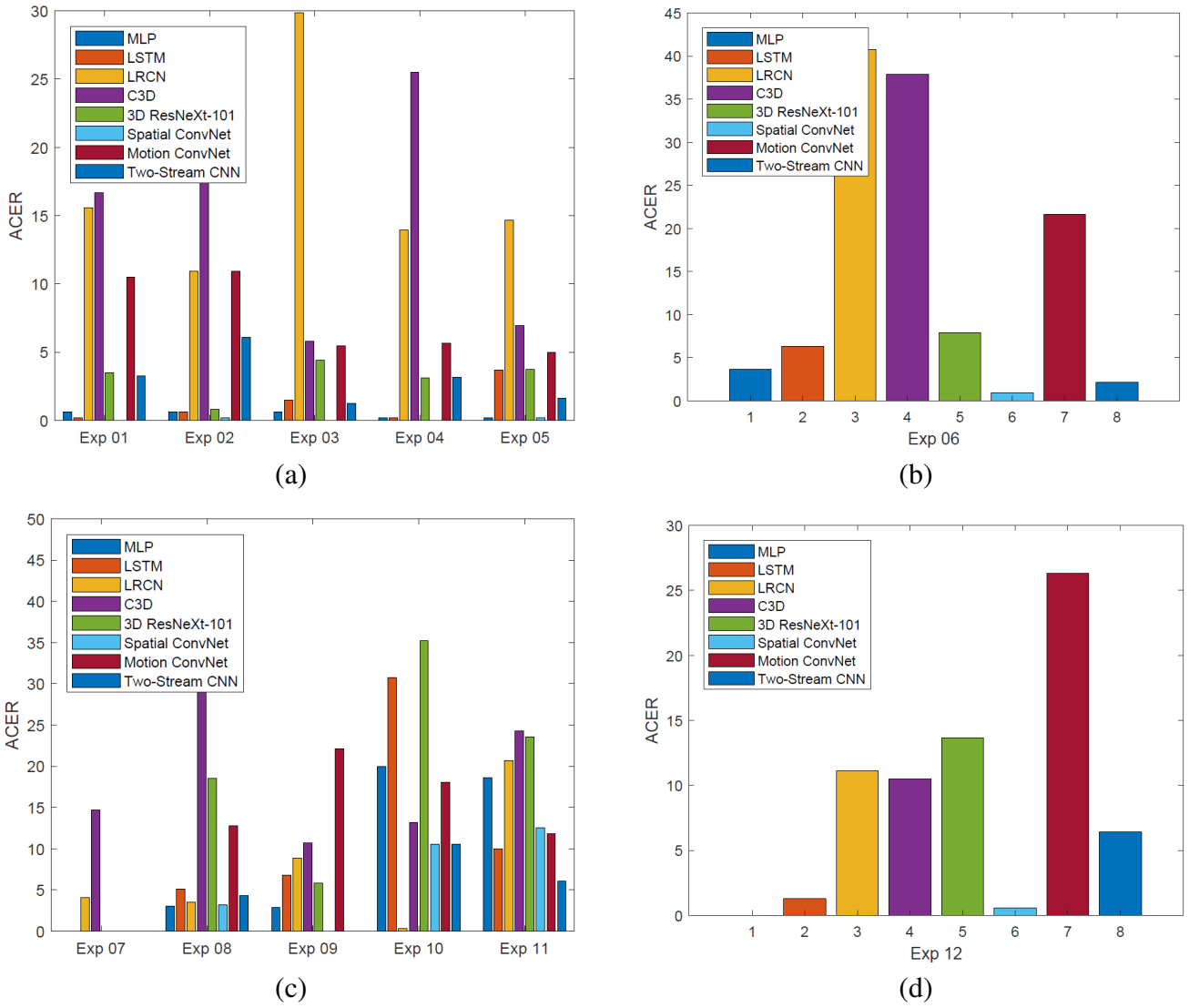


Figure 3.6: Comparison of ACERs of (a) Intra-session experiments (Exp.01-05), (b) Cross-session experiments (Exp.06), (c) Cross-attack experiments (Exp.07-11), and (d) Baseline experiment (Exp.12) on the IPV dataset.

Table 3.4: Training and testing setup for cross-attack (Exp. 07-11) and baseline (Exp. 12) experiments on the IPV dataset.

Experiments	Unseen PA	Category	Train	Test
Exp. 07	Paper print	Bonafide	197	35
		PA	401	74
Exp. 08	Artificial eye	Bonafide	197	35
		PA	319	156
Exp. 09	Kindle display	Bonafide	197	35
		PA	424	51
Exp. 10	Funny glasses	Bonafide	197	35
		PA	309	166
Exp. 11	Mannequin	Bonafide	197	35
		PA	447	28
Exp. 12	N/A	Bonafide	99	89
		PA	280	136

Table 3.5: ACER (%) of proposed methods across all experiments (Exp. 01-12) on the IPV dataset.

Experiments (unseen)	Intra-session					Cross-session	Cross-attack					Baseline
	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5	Exp. 6 (IPV01-02)	Exp. 7 (Print)	Exp. 8 (Artificial)	Exp. 9 (Kindle)	Exp. 10 (Funny Glasses)	Exp. 11 (Mannequin)	Exp. 12
MLP	0.61	0.61	0.61	0.20	**0.20**	3.66	**0.0**	**3.01**	2.85	20.00	18.57	**0.0**
LSTM	0.20	0.61	1.47	0.20	3.67	6.35	**0.0**	5.12	6.77	30.73	10.00	1.29
LRCN	15.57	10.94	29.87	13.93	14.68	40.78	4.10	3.52	8.82	**0.30**	20.71	11.10
C3D	16.66	23.68	5.81	25.49	6.93	37.90	14.67	46.31	10.70	13.20	24.28	10.50
3D ResNeXt-101	3.50	0.81	4.41	03.09	3.75	7.93	**0.0**	18.56	5.79	35.22	23.57	13.63
Spatial ConvNet	**0.0**	**0.20**	**0.0**	**0.0**	**0.20**	**0.96**	**0.0**	3.20	**0.0**	10.54	12.50	0.55
Temporal ConvNet	10.49	10.94	5.47	5.67	4.97	21.62	**0.0**	12.76	22.12	18.07	11.78	26.31
Two-stream CNN	3.25	6.08	1.26	3.14	1.62	2.12	**0.0**	4.31	0.0	10.54	**6.07**	6.41
Cross-Modality	5.88	3.75	4.16	9.63	10.90	11.78	6.15	19.84	3.38	26.95	24.64	-

complexity incurred in estimating optical flows for such large datasets, we did not analyze the Temporal ConvNet method for detecting face PAs.

3.5.2.1 Results on SiW dataset

The SiW [165] dataset provides three evaluation protocols along with a baseline method. Training and testing performed on the disjoint set of subjects in all three protocols. Training performs on 90 subjects and testing on rest (75 subjects). Protocol 1 evaluates the generalizability of algorithms under different face poses and expressions by considering only the first 60 frames for training (frontal view) and rest for testing. Protocol 2 represents the scenario of cross-medium of the same spoof type (replay attack). Training is on three replay attack media and tested on the fourth

Table 3.6: ACER (%) for all methods on the SiW [165] dataset. The ACER values outperforms the baseline [165] are shown in bold.

Protocol	Subset	Subject#	Attack	Auxiliary [165]	MLP	LSTM	LRCN	C3D	3D ResNeXt-101	Spatial ConvNet
1	Train	90	First 60 Frames	3.58	**0.034**	**0.0835**	**0.569**	**0.725**	**1.412**	**0**
	Test	75	All							
2	Train	90	3 display	0.57 ± 0.69	**0.083 ± 0.118**	2.216 ± 3.620	2.031 ± 2.873	**0.523 ± 0.574**	**0.208 ± 0.259**	**0 ± 0**
	Test	75	1 display							
3	Train	90	print (display)	8.31 ± 3.81	11.38 ± 15.98	9.865 ± 13.951	**1.063 ± 0.363**	**0.1849 ± 0.081**	22.561 ± 0.241	**2.0399 ± 0.246**
	Test	75	display (print)							

medium. The protocol uses a leave-one-out strategy and reports the mean and standard deviation of four experiments. Protocol 3 represents the scenario of cross-attack, where training is on print attacks and testing on replay attacks and vice versa. Table 3.6 presents results (ACER) on all three protocols.

The proposed methods compared with the algorithm specified in the work [165]. For protocol 1, all scene-based methods outperform the baseline [165]. Scene information is invariant to the variations of face pose and expression when used for detecting PAs. For protocols 2 and 3 as well, cues from the entire scene are more crucial than cues from just facial region in detecting face PAs. The C3D and Spatial ConvNet methods perform the best on this dataset. The issue of limited training data gets resolved for the C3D method on this dataset.

3.5.2.2 Results on SiW-M dataset

To further analyze the role of scene information in detecting the face PAs in the unseen attack scenario, we perform experiments on the SiW-M dataset [166]. The dataset specified 13 experimental splits for evaluating the performance on each presentation attack following the leave-one-out strategy. For each experiment split, training performed on 12 types of spoof attacks and 80% of the bonafide videos and testing on one left attack type and 20% of bonafide videos. There is no overlapping of subjects between the training and testing sets of bonafide videos. Table 3.7 presents the results (ACER) of all 13 experimental splits.

The proposed scene-based methods compared with SVM-RBF + LBP [33], Auxiliary [165], and Deep Tree Learning [166] algorithms. All scene-based methods outperform methods proposed in [165] and [33] when looking at the average (last column). Except for the C3D and LRCN

Table 3.7: ACER (%) for all methods on the SiW-M [166] dataset.

Methods	Replay	Print	Mask Attacks					Makeup Attacks			Partial Attacks			Average
			Half	Silicone	Trans.	Paper	Manne	Obfusc.	Imperson.	Cosmetic	Funny Eye	Paper Glasses	Partial Paper	
SVM-RBF + LBP [33]	20.6	18.4	31.3	21.4	45.5	11.6	13.8	59.3	23.9	16.7	35.9	39.2	11.7	26.9 ± 14.5
Auxiliary [165]	16.8	6.9	19.3	**14.9**	52.1	8.0	12.8	55.8	13.7	11.7	49.0	40.5	5.3	23.6 ± 18.5
Deep Tree Learning [166]	9.8	6.0	15.0	18.7	36.0	4.5	7.7	48.1	11.4	14.2	19.3	19.8	8.5	16.8 ± 11.1
MLP	6.77	4.35	**8.24**	16.03	**11.66**	0.76	1.53	**26.75**	2.35	10.84	**2.23**	**5.43**	1.15	**7.54 ± 7.13**
LSTM	**5.24**	5.0	13.79	20.44	12.18	0.76	1.92	26.88	4.04	14.61	3.17	9.73	1.15	**9.14 ± 7.76**
LRCN	7.92	17.39	12.61	31.66	34.5	4.86	18.65	45.11	5.58	23.92	16.25	31.58	1.15	19.32 ± 2.81
C3D	9.85	16.7	11.47	37.74	27.73	4.86	13.46	41.27	6.45	22.15	15.62	30.1	0.38	18.29 ± 12.22
3D ResNeXt-101	13.38	10.71	10.58	21.57	29.11	2.94	8.15	45.95	5.89	27.61	23.59	27.24	9.55	**13.38 ± 1.75**
Spatial ConvNet	5.61	**1.61**	11.59	18.51	24.58	**0.00**	**0.00**	28.89	**0.81**	**6.76**	17.45	14.32	**0.00**	**10.01 ± 9.62**
Cross-Modality	24.32	10.94	28.70	19.72	36.26	11.83	3.05	46.56	24.94	34.32	10.30	16.62	1.14	20.66 ± 12.96

methods, all other scene-based methods also outperform [166]. Considering individual unknown attacks, MLP and Spatial ConvNet methods show promising results under a cross-attack or unknown attack scenario. Results on SiW [165] and SiW-M [166] datasets show that cues from the entire image are more effective in detecting face PAs as it contains cues from the facial region as well as background region.

3.5.2.3 Results on OULU-NPU dataset

The OULU-NPU dataset specified four evaluation protocols. Data is divided into three subject-disjoint subsets named training, development, and testing. Protocol 1 assesses the face PA detection algorithms under unseen illumination and location. The training uses data of Sessions 1 and 2, and testing is on data of session 3. Protocol 2 evaluates the effect of using different presentation attack instruments (PAI) in print and replay attacks. Training is on one type of print and replay attack and testing on another type of print and replay attacks. Protocol 3 analyses the effect of the input sensor variations on PA detection algorithms using the leave-one-out strategy. There are six sensors used to capture the data. The training performs on videos of five sensors and testing on the remaining one. Protocol 4 combines all three challenges and evaluates the generalizability of face PA detection methods under unseen environmental conditions, PAIs, and input sensors. Table 3.8 presents results (ACER) on all four protocols.

The proposed scene-based methods are compared with SVM-RBF + LBP [33], Auxiliary [165], and De-spoofing [133]. For protocols 2 and 3, the Spatial ConvNet method performs the best. For all four protocols, 3D ResNeXt-101 and Spatial ConvNet methods are more effective than the

Table 3.8: ACER (%) for all methods on the OULU-NPU [33] dataset.

Methods	Protocol 1	Protocol 2	Protocol 3	Protocol 4
SVM-RBF + LBP [33]	13.5	14.2	12.1 ± 3.7	27.2 ± 14.3
Auxiliary [166]	1.6	2.7	2.9 ± 1.5	9.5 ± 6.0
De-spoofing [133]	**1.5**	4.3	3.6 ± 1.6	**5.6 ± 5.7**
MLP	22.29	16.25	18.68 ± 4.83	23.33 ± 8.97
LSTM	26.45	16.52	20.83 ± 7.49	23.75 ±5.72
LRCN	46.45	30.55	30.34 ± 7.01	48.75 ± 1.90
C3D	48.75	26.52	27.77 ± 7.24	49.16 ± 1.86
3D ResNeXt-101	5.83	5.13	4.72 ± 1.03	23.33 ± 8.97
Spatial ConvNet	3.54	**2.5**	**2.84 ± 1.77**	24.16 ± 16.62

baseline methods [33] even though the contextual information is deliberately kept out of the videos. Other scene-based methods perform poorly on this dataset. We anticipate these results on this dataset as the proposed approaches focus on cues from the entire frame (spatial) or along the temporal dimension. The dataset suppresses scene cues, which result in poor performance. Hence, the capture contextual information is advantageous for PA detection.

3.5.3 Cross-modality

We also perform two more experiments to evaluate the usefulness of scene cues in PA detection across modalities. In the first experiment, training performs on face PAs taken from SiW-M [166] dataset, and evaluation is on all test splits of the iris IPV dataset. Table 3.5 (last row) shows all results (ACER). Though the results show the inferior performance of the cross-modality model over the intra-modality model, the cues learned from the face PAs are worthwhile in distinguishing bonafide and PA samples in iris modality.

In the second cross-modality experiment, training performs on iris PAs (IPV dataset) and testing on the experimental splits of SiW-M [166] dataset. The last row of Table 3.7 presents its results. Surprisingly, it performs better than the SVM-RBF + LBP [33] and Auxiliary [165] techniques when examining the overall average (last column of table 3.7). Reasonable results of scene-based techniques under the cross-modality scenario validate the presence of common scene cues across PAs of different modalities.

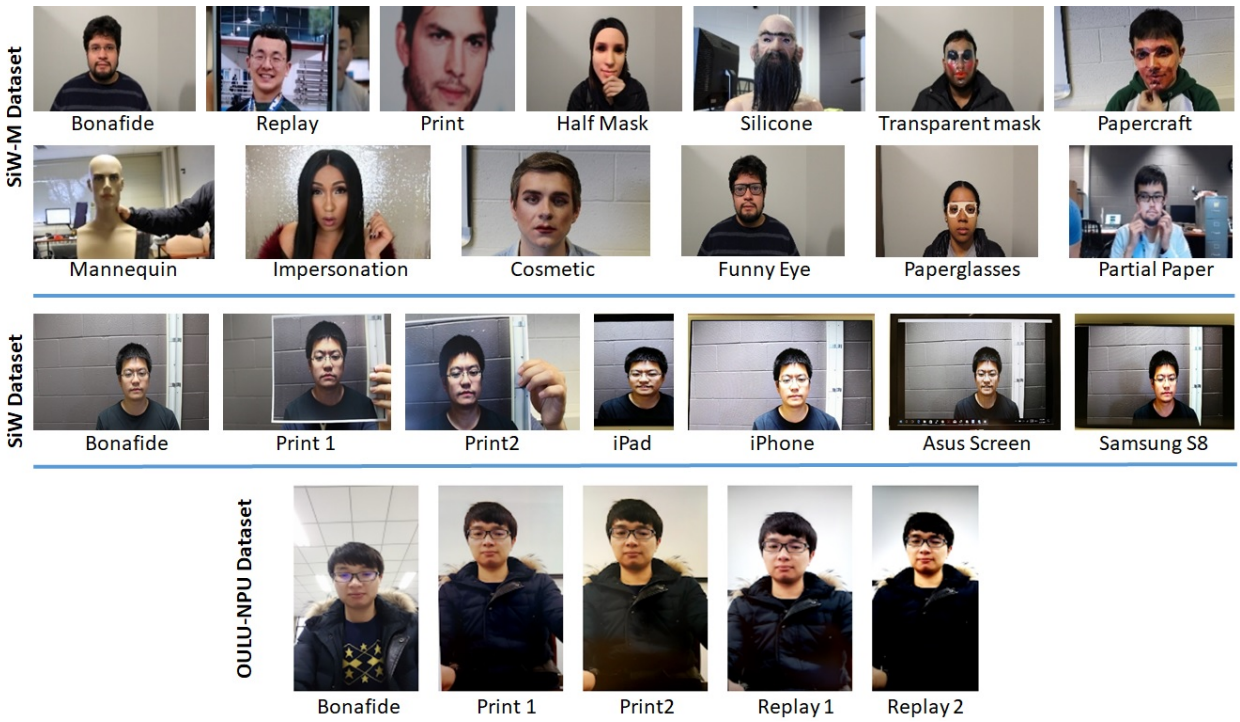


Figure 3.7: Sample video frames from various face PAD datasets: the first block shows frames from the SiW-M [166] dataset, the second block represents examples from the SiW [165] dataset and the third block shows samples from the OULU-NPU [33] dataset.

The key findings observed from all the experiments conducted in this work are as follows:

1. The scene provides useful information for detecting iris PAs under different (intra-session, cross-session, and cross-attack) scenarios (refer Iris Modality results).

2. The cues obtained from the scene video are complementary to the one obtained from the NIR iris region (refer to the last column of Table 3.5).

3. Scene analysis could be extended to other modalities. Outperforming results on the face modality (refer Face Modality results) validate the hypothesis.

4. Scene-based techniques utilize the common cues of presentation attacks across biometric modalities (refer to Cross-modality results).

5. Spatial ConvNet performs best in the majority of the experiments.

3.6 Analysis Using Heatmaps

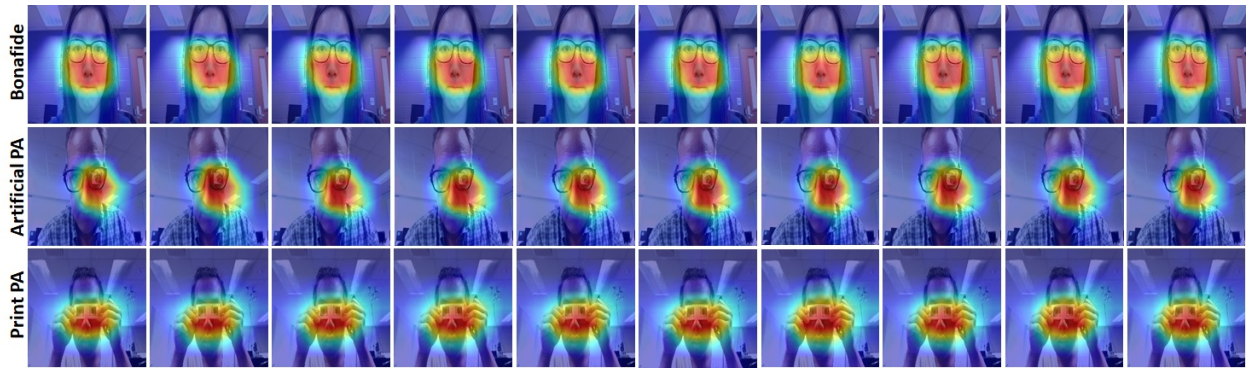


Figure 3.8: Frames of bonafide (first row), artificial eye (second row), and paper print (third row) videos overlaid with their corresponding Grad-CAM heatmaps. The columns correspond to the different frames of a video. Heatmap represents the focused region of a frame by the trained model (Spatial ConvNet). Red gradient regions in the heatmaps represent high focused regions considered by the trained model, whereas the blue-colored regions represent low focused regions. On the bonafide frames, the focus is mainly over the center of a face. On artificial eye frames, the focus is on the artificial eye mounted over the glasses. In the case of paper print video, the focus is on the print of the eyes. Different regions of focus in different categories help in differentiating bonafide videos from spoof one.

We further visually analyze the result by generating “heatmaps" using Gradient-weighted Class Activation Mapping (Grad-CAM) [245]. Grad-CAM produces a coarse localization map highlighting the salient regions in an image that were used by the network to generate its inference. It is generated by estimating the gradient of the loss function and backpropagates it through the hidden layers to the input frame. We use the Spatial ConvNet model trained using experiment 11 (Iris Modality cross-attack) setup for generating heatmaps of bonafide, artificial eye, and paper print videos (most commonly used PAs). Figure 3.8 exhibits sample frames of bonafide, artificial eye, and paper print videos along with their heatmaps. The first row of Figure 3.8 shows the heatmaps of bonafide frames, where the high activation regions are at the center of a face. The other two rows of Figure 3.8 correspond to artificial and print PA frames, where high activation regions are around the artificial eye and print paper respectively. The presence of spoof artifacts in the video frames shifted the salient region towards the artifacts. Distinct region of focus aids the models to discriminate bonafide from PAs videos.

3.7 Conclusion and Future work

We proposed an approach that utilizes multiple frames of a scene video for detecting the presentation attacks in iris and face biometric modalities. Experimental results validated the presence of significant cues in the scene video for detecting the PAs. In the case of iris modality, scene video cues are also complementary to the cues obtained from the NIR iris image. It has the generalizable capability as it produces reasonably good results with unseen attacks and modalities. We extended the approach for the face modality, but it could also be extended for other modalities such as fingerprint, where holding a fake fingerprint can be evidence for detecting the presentation attack.

CHAPTER 4

IRIS PRESENTATION ATTACK DETECTION USING A OCT IMAGE

Parts of this chapter appeared in the following publication:

R. Sharma and A. Ross, "Viability of Optical Coherence Tomography for Iris Presentation Attack Detection," International Conference on Pattern Recognition (ICPR), Milan, Italy, January 2021.

4.1 Introduction

In this chapter, we present another iris presentation detection (PAD) method utilizing Optical Coherence Tomography (OCT) imaging. Existing PAD methods utilize NIR or VIS imaging which captures the stromal textural patterns of the iris, whereas OCT[1] images capture the internal structure of the eye and the iris (Figure 4.1). The OCT imaging has been utilized for fingerprint PA detection [180]. But the unavailability of an OCT iris dataset and the high hardware costs associated with OCT has traditionally prevented its exploration for iris PA detection. However, the development of cost-effective OCT hardware [256] motivates us to consider it for iris PA detection.

The main contributions of the work are as follows:

1. We propose a hardware-based iris PA detection technique based on OCT imaging technology. We also assess its viability by comparing its performance against traditional NIR and VIS imaging modalities.

2. We implement OCT-based iris PA detection using three state-of-the-art deep CNN models which significantly differ in their architectures: VGG19 [255], ResNet50 [106] and DenseNet121 [121].

[1]OCT also employs NIR illumination but obtains cross-sectional views, not textural details.

3. We evaluate PA detection performance on a dataset of 2,169 bonafide, 177 Van Dyke eyes and 360 cosmetic contact lens images under intra-attack and cross-attack scenarios. Each input sample is captured in all three imaging modalities.

4. We also generate CNN visualizations (heatmaps [245] and t-SNE plots [280]) to further analyze the results on OCT, NIR and VIS images. Heatmaps are used to identify salient image regions that the deep architectures utilize to detect PAs. t-SNE plots aid in visualization of features extracted by the CNN architectures.

The rest of the chapter is organized as follows. Section 4.2 discusses the existing work for detecting hardware-bsed iris PAs. Section 4.3 discusses background of the imaging modalities. Section 4.4 describes the proposed approach. Section 4.5 provides a description of the dataset. Section 4.6 describes the experimental setup and reports the results. Section 4.7 provides a detailed analysis of the results obtained from the proposed approach. Finally, Section 4.8 concludes the chapter.

4.2 Related Work

Various presentation attack detection techniques in iris modality utilize different imaging techniques. Commonly used imaging technique is near-infrared (NIR) imaging [46, 114, 176, 249, 315]. Zhang *et al.* [315] utilized texture-based features, whereas works in [46, 114, 176, 249] used deep features to detect iris PAs. The authors in [98, 211] operated on visible spectrum (VIS) imaging utilizing LBP [98] and BSIF [211] features. Menotti *et al.* [176] showed results on both NIR and VIS iris images. Raghavendra and Busch [219] exploited characteristics of the Light Field Camera (LFC) for iris PA detection in the VIS spectrum. Sequeira *et al.* [246] suggested the use of a one-class classifier on VIS images for generalization across unseen attacks, i.e., attacks that were not used in the training phase. In [214], the authors utilized Eulerian Video Magnification (EVM) to detect PAs in VIS videos. Park and Kang [198] utilized a specialized tunable filter to capture iris images at different spectral bands ranging from 650nm to 1100nm. These multi-spectral images are then fused at the image level to detect PAs. Lee *et al.* [152] analyzed the reflectance properties of the

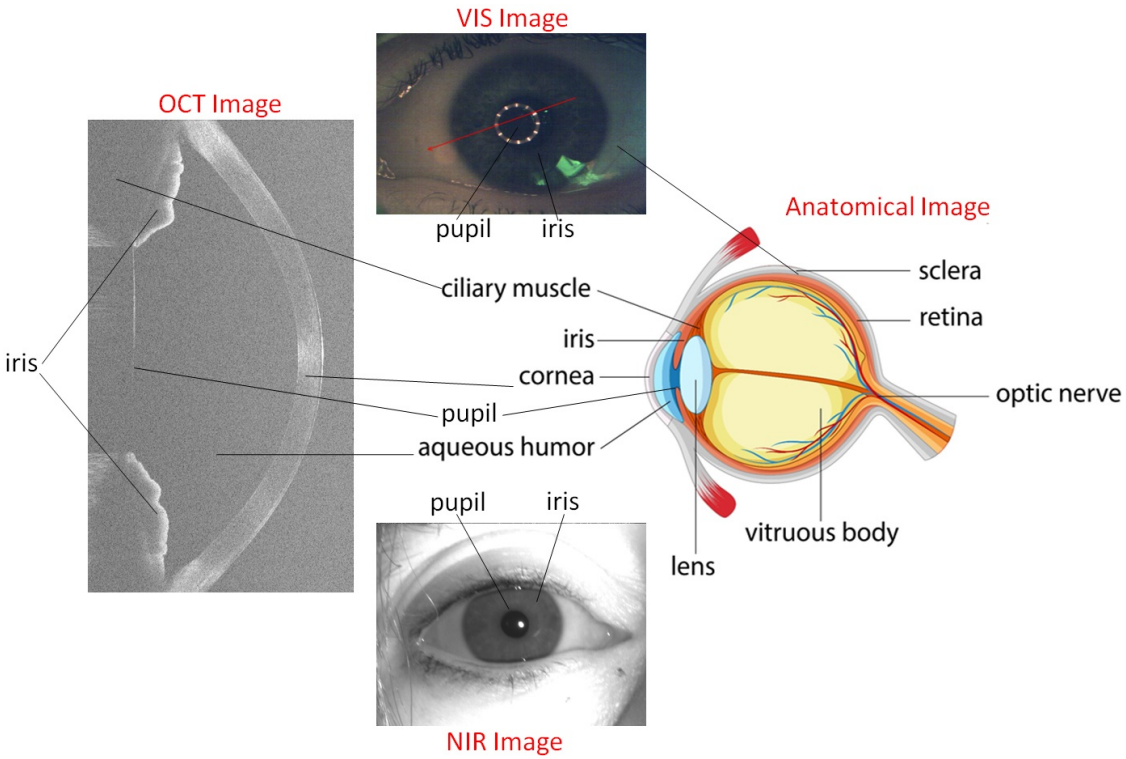


Figure 4.1: Components of the eye and iris sensed using OCT, NIR and VIS imaging. The anatomical image (*https://www.vecteezy.com/vector-art/431288-parts-of-human-eye-with-name*) is also shown. The red line in the VIS image shows the traverse scanning direction of the OCT scanner.

iris and sclera in multi-spectral illumination. Chen *et al.* [49] captured images at the near-infrared (860nm) and blue (480nm) wavelengths, and then analyzed the conjunctival vasculature patterns and the iris textural patterns for liveness detection.[2] Connell *et al.* [54] exploited the anatomy and geometry of the human eye using structured light to detect cosmetic contact lens. Thavalengal *et al.* [266] used both VIS and NIR images for iris liveness detection in smartphones. Hsieh *et al.* [117] utilized dual-band imaging hardware (VIS and NIR) to distinguish between the textured pattern of contact lens from real iris patterns using independent component analysis.

4.3 Background of Iris Imaging Modalities

The complex texture of the iris is characterized by its components, including, pigments (chromophore), blood vessels, muscles, crypts, contractile furrows, freckles, collarette and pupillary

[2]Early literature used the term “liveness detection" to refer to the problem of PA detection.

frills. Different spectral bands can potentially be used to capture different components of the iris. NIR illumination, which operates in the 700-900nm range, predominantly captures the stromal features (fibrovascular layer) of the iris, whereas VIS (400-700nm) captures information about the pigment melanin. Optical Coherence Tomography (OCT) [120] is a non-invasive, micrometer-resolution imaging modality, that can be used to capture 2-D cross-sectional or 3-D volumetric images of an eye. It is mainly used for biomedical and clinical purposes, such as ophthalmology, optometry, cardiology and dermatology. It works with a low-coherence near-infrared (800nm-1325nm) light source. OCT imaging captures cornea (circular arc), iris tissue structure, anterior humor (the space between iris and cornea) and the ciliary muscles (next to the iris tissues) of the eye as shown in Figure 4.1. OCT images are captured by shining the light source over a beam splitter, which splits the light into two beams, one directed to the sample arm (human eye) and another to the reference arm (mirror). The time delay and intensity of the back-reflected light from both the arms are estimated to create an axial back-scattering profile called A-Scan. Combination of A-Scans along transverse axis forms a 2-D cross-sectional image called B-Scan. The imaging setup of an OCT sensor is shown in Figure 4.2. OCT imaging primarily captures the structure and morphology of the eye as opposed to texture information that is typically observed in NIR and VIS images.

A majority of commercial iris recognition systems and iris PA detection algorithms utilize NIR images for the following reasons. Firstly, NIR illumination penetrates deeper into the iris and elicits the textural pattern of both light and dark irides; in contrast, majority of VIS illumination is absorbed by higher levels of melanin in dark-colored irides resulting in poorly discernible iris texture. Secondly, background illumination variations and corneal reflections do not affect NIR imaging as much as RGB imagers. However, some iris recognition and PA detection algorithms have started using VIS imaging due to inexpensive hardware and a wide range of applications (mobile devices, surveillance, etc.) [98, 286]. Due to expensive hardware, OCT imaging has not been traditionally discussed in the literature for either iris recognition or PA detection.

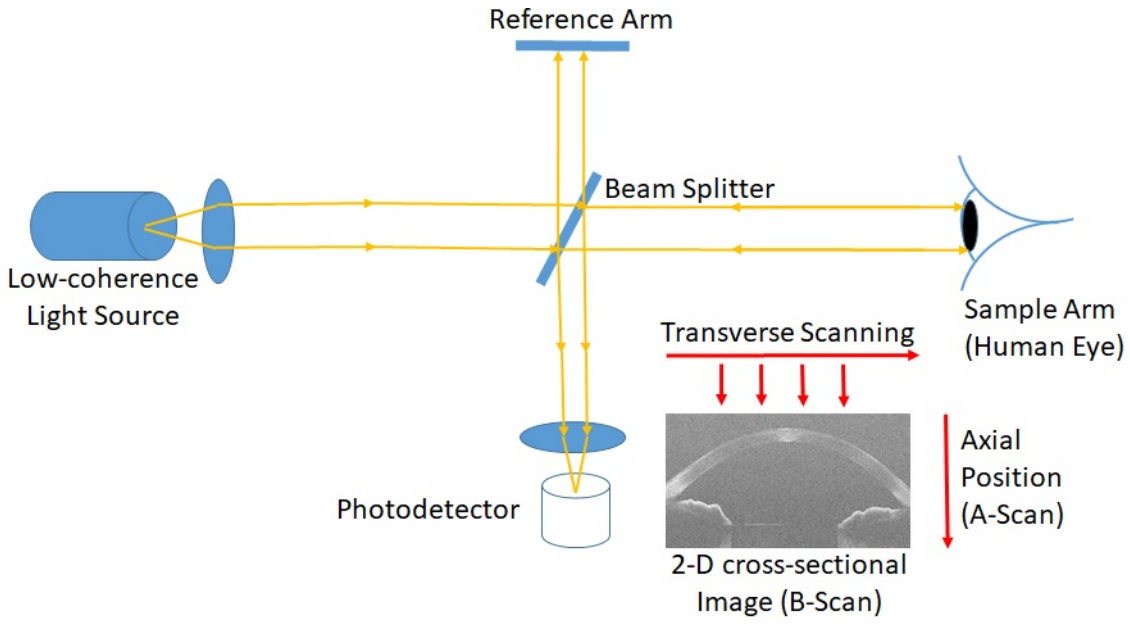


Figure 4.2: Typical optical setup of an OCT scanner. Low-coherence light is incident over the beam splitter, which splits the light into sample and reference arms. Back-reflected light from sample and reference arms are then collected by the photodetector. Cross-sectional OCT image (B-scan) is formed by combining a number of A-scans along the transverse direction.

4.4 Proposed Approach

In this work, we discuss the use of OCT imaging for iris PA detection. For classification of iris OCT images as bonafide or PA, we used three state-of-the-art deep CNN architectures: VGG19 [255], ResNet50 [106] and DenseNet121 [121]. These architectures output a single PA score in the range [0, 1], with a '1' indicating a PA and '0' indicating a bonafide. Using the same CNN architectures, we compare the PA detection capability of OCT images against NIR and VIS images. Overview of the approach is depicted in Figure 4.3. In the subsequent sub-section, we provide implementation details of all three network architectures.

To classify bonafide and PA iris images acquired from all three imaging modalities, we used three state-of-the-art deep architectures: VGG19 [255], ResNet50 [106] and DenseNet121 [121]. These three networks differ by the number of the convolutional layers, the number of trainable parameters and the connection type. VGG19 [255] has 19 convolutional layers with kernels of fixed size 3 $\times$ 3 throughout the network. It has 143,667,240 trainable parameters. ResNet50 [106] has 50

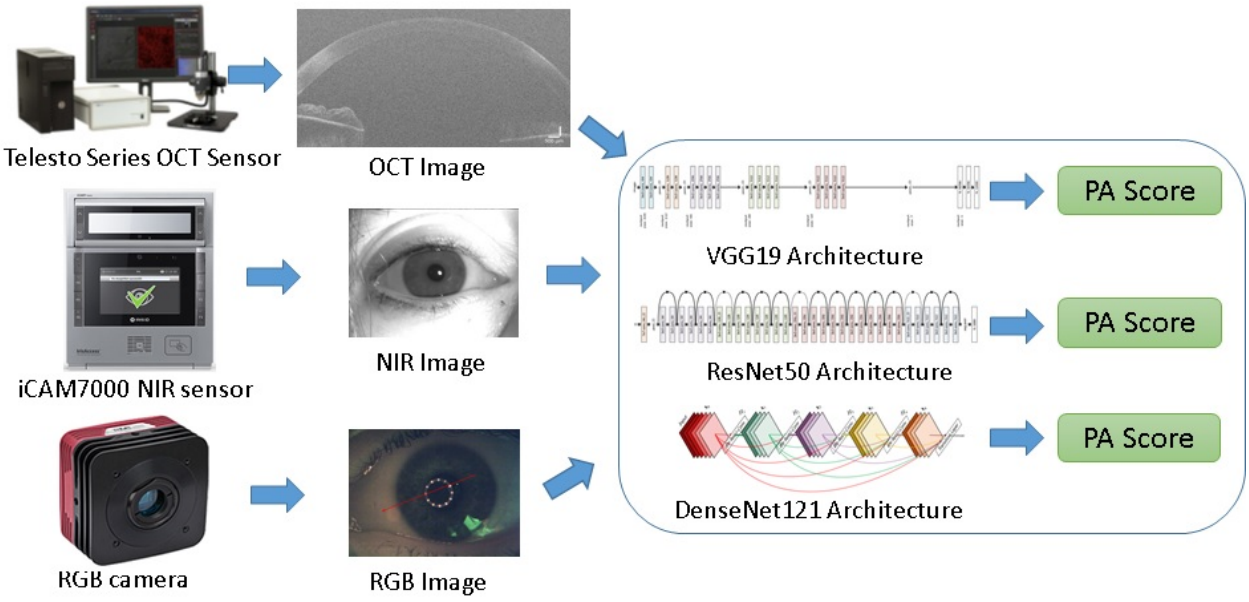


Figure 4.3: Comparative analysis of OCT, NIR and VIS imaging in detecting iris PAs. Three architectures, viz., VGG19, ResNet50, DenseNet121, are used for distinguishing between bonafides and PAs by emitting a PA score. A higher PA score indicates the input is a “PA" and a lower score indicates the input is a “bonafide" image.

convolutional layers with residual connections (skip connections) to moderate gradient flow and allow the training of a large network. It has 35,610,216 trainable parameters. DenseNet121 [121] consists of 121 convolutional layers, where each layer is connected to every other layer resulting in a much reduced set of trainable parameters (7,978,856). Three different sized architectures are utilized in the study to eliminate the bias created due to the network architecture (under-fitting or over-fitting) in the comparison results. As the dataset used in the study is insufficient to train these deep architectures, we utilize pre-trained models on ImageNet dataset. Pre-trained models also help in faster convergence during the training process. ImageNet is a large dataset used for object classification containing 1.2 million images of 1000 classes. The images in ImageNet dataset are visible spectrum images, i.e., RGB. To preserve the usefulness of pre-trained weights for the OCT and NIR spectrum images, we normalize OCT, NIR and VIS images using the mean and the standard deviation calculated from the ImageNet dataset images. The photometrically normalized images are then re-sized to 224×224 and input to the aforementioned architectures. All three models are then fine-tuned using OCT, NIR and VIS iris images resulting in nine trained models.

Table 4.1: Number of bonafide and PA samples corresponding to each imaging modality.

Classes	**Sub-Classes**	**Imaging Modality**		
		OCT	**RGB**	**NIR**
Bonafide		844	844	1371
Artificial Eyes	Van Dyke Eye (Brown)	30	30	51
	Van Dyke Eye (Blue)	29	29	56
	Face Mask	2	2	4
Cosmetic Contacts	Acuvue Accent Vivid	37	37	43
	Air Optix Sterling Grey	41	41	43
	Extreme FXS Halloween Blackout	42	42	34

The learning rate used in the training is 0.005, the batch size is 20, the optimization algorithm is stochastic gradient descent with momentum of 0.9, the number of epochs is 50, and the loss function is cross-entropy. During test and evaluation, each of these networks produce a single PA score which is used along with a threshold to determine if the input image is a PA or a bonafide.

4.5 Dataset

The dataset is collected under the Odin program of IARPA [2] from 740 eyes (370 subjects). Figure 4.4 provides age distribution of subjects. The number of male and female subjects are 136 and 243, respectively. OCT, NIR and VIS images are collected sequentially for a subject using an RGB camera, iCAM7000 NIR sensor and THORLabs Telesto series (TEL1325LV2) OCT sensor [5], respectively. The OCT images are acquired at 1325nm wavelength having 7mm imaging depth and 12μm axial imaging resolution. For a single sample, 50 cross-sectional frames are captured by the OCT sensor. However, temporal information is not significant among frames, so we use only the first frame. Iris PAs considered in this study are artificial eyes (Van Dyke eyes) and cosmetic contact lenses. For OCT and VIS, the dataset contains 844 bonafide images, 61 artificial eyes and 120 cosmetic contact lens images, whereas, for NIR, there are 1,371 bonafide images, 111 artificial eyes and 120 cosmetic contact lens images. Further sub-categorization of PA images is provided in Table 4.1. Figure 4.5 shows examples of bonafide and PA images acquired in all three spectra (OCT, NIR and VIS).

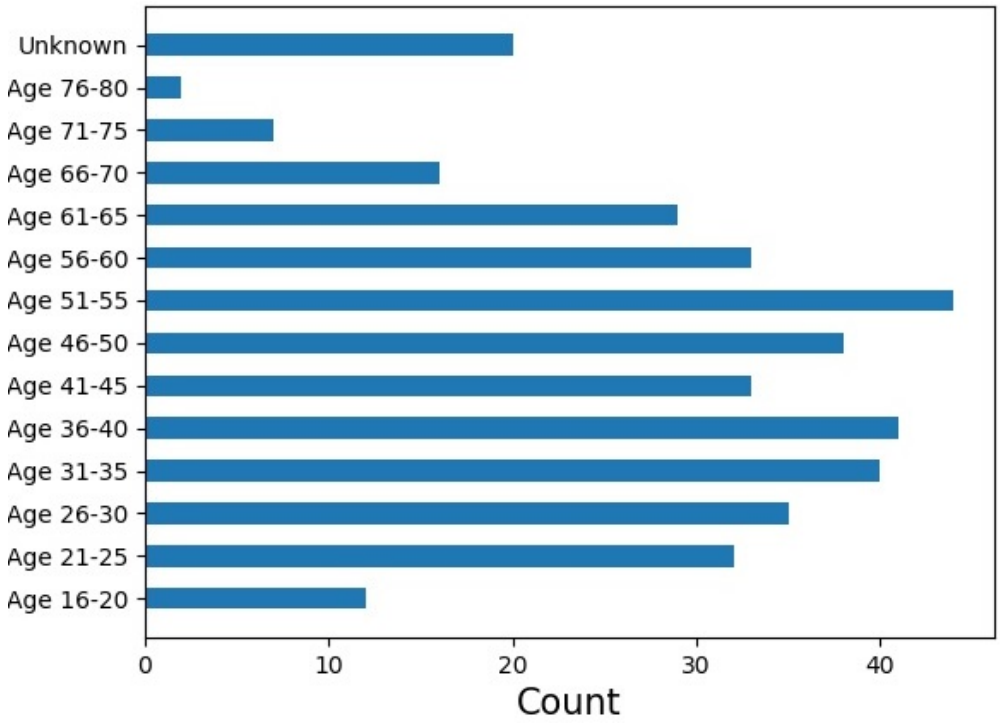


Figure 4.4: Age distribution of subjects in the dataset.

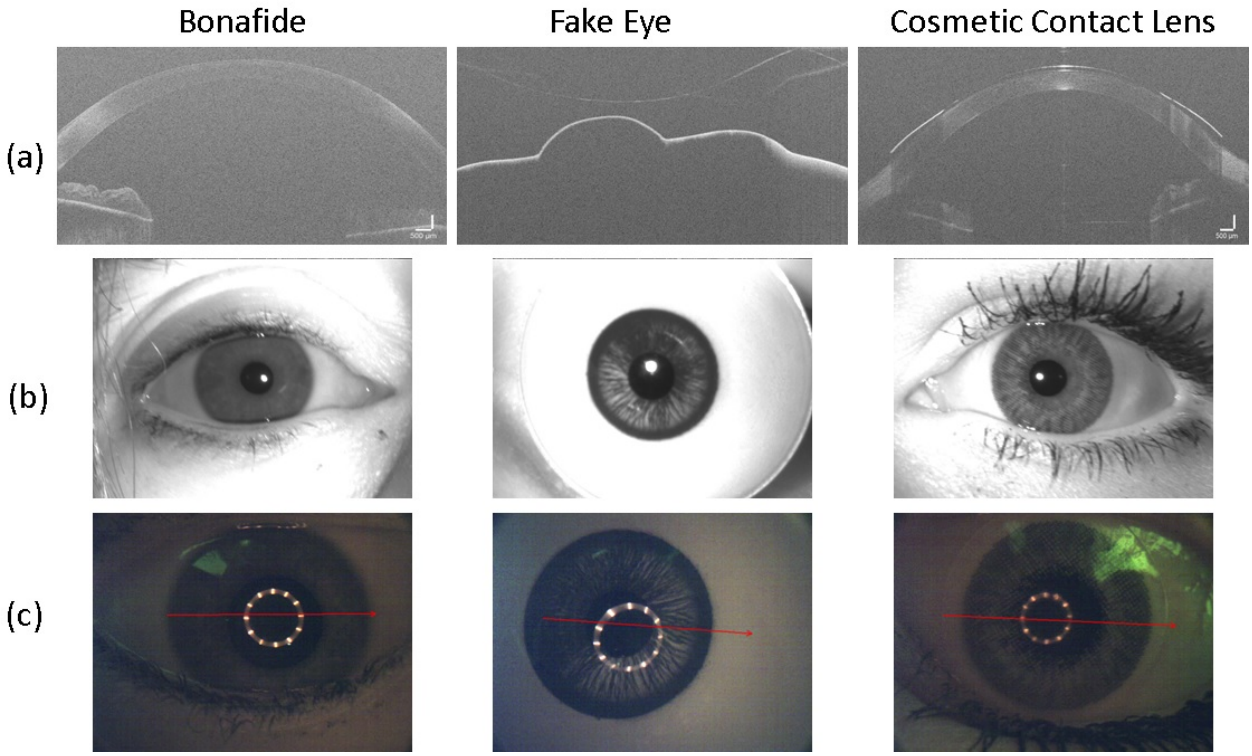


Figure 4.5: Samples of bonafide, artificial eyes and cosmetic contact lens images captured using (a) OCT, (b) NIR and (c) VIS imaging modalities.

Table 4.2: APCER (%) and BPCER (%) of all algorithms on LivDet-Iris 2017 Dataset [304]. Results are presented by averaging APCER and BPCER of all test sets in the dataset.

Algo.	**CASIA [304]**	**Anon1 [304]**	**UNINA [304]**	**VGG19**	**ResNet50**	**DenseNet121**
APCER	11.88	14.71	15.52	15.80	11.71	**6.25**
BPCER	9.48	3.36	12.92	**1.20**	3.24	10.39

4.6 Experimental Setup and Results

Before evaluating the three imaging modalities (OCT, NIR and VIS), we assess the performance of three fine-tuned architectures (VGG19, ResNet50 and DenseNet121) on the LivDet-iris 2017 [304] dataset for iris PA detection. The dataset is an amalgamation of Clarkson, Warsaw, Notre Dame and IIITD-WVU datasets. Print and cosmetic contact lens PAs are included in the dataset. The experimental setup is kept the same as specified in the competition [304]. Evaluation measures are Attack Presentation Classification Error Rate (APCER) and Bonafide Presentation Classification Error Rate (BPCER), where APCER is the proportion of PA samples misclassified as bonafide and BPCER is the proportion of bonafide samples misclassified as PAs. All three architectures either outperform or are comparable to the state-of-the-art algorithms (CASIA, Anon1 and UNINA) on the LivDet-iris 2017 competition as shown in Table 4.2.

Utilizing the three architectures, we perform comparative evaluation of OCT, NIR and VIS images in detecting iris PAs. Experiments are performed under intra- and cross-attack scenarios. Samples that were successfully captured in all three imaging modalities are selected for experiments. The dataset used for evaluation eventually has 723 bonafide samples, 59 artificial eyes and 120 cosmetic contact lens images captured in all three imaging modalities. The train, validation and test sets are *eye-disjoint*, i.e., they have data from different eyes and samples in the three sets are *mutually exclusive*. Intra-attack experiments examine which imaging modality performs best with known PAs (used during training), whereas cross-attack experiments analyze the generalizability across unknown PAs (not used in training). The evaluation measures used are True Detection Rate (TDR) at 0.2% False Detection Rate (FDR), and Average Classification Error Rate (ACER). TDR is the percentage of PA samples that were correctly detected, whereas FDR is a percentage

of bonafide samples that were misclassified as PA. ACER is the average of APCER and BPCER. Receiver operating characteristic (ROC) curves are also provided for a comprehensive overview. For successful detection, TDR should be comparatively higher and ACER should be comparatively lower.

4.6.1 Intra-attack Setup and Results

In the intra-attack setup, three experiments are performed: Intra-EXP 1, Intra-EXP 2 and Intra-EXP 3. Intra-EXP 1 includes both the PAs (artificial eyes and cosmetic contact lens) and bonafide images in the training and test sets, whereas Intra-EXP 2 and Intra-EXP 3 include images from only one PA along with bonafide images for training and testing. Intra-EXP 2 and Intra-EXP 3 experiments are performed to test the difficulty level of differentiating a specific PA from bonafide samples. Details about the train, validation and test sets of all three experimental setups are provided in Table 4.3. In the first experiment (Intra-EXP 1), the data are split in a 70:30 ratio, where 70% of eyes is used for training and the remaining for testing (30%). Thereafter, five-fold cross-validation is employed on the training set, where 4 folds are used for training and one for validation. The validation set is used to estimate the threshold to be used on the test set for calculating ACER. The TDR at 0.2% FDR and the ACER for VGG19, ResNet50 and DenseNet121 architectures are provided in Table 4.4. ROC curves of Intra-EXP 1 for all three architectures are shown in Figures 4.6(a), 4.7(a) and 4.8(a).

In the Intra-EXP 1 experiment, the best results are observed on OCT images, second-best on NIR images, and then on VIS images. All trained models (five) obtained from cross-validation show low standard deviation in the results when tested on OCT images (Figures 4.7(a) and 4.8(a)) compared to NIR and VIS images. Similar results are observed across all three network architectures (VGG19, ResNet50 and DenseNet121). This validates the robustness of PA detection when using OCT images. Considering individual PAs in Intra-EXP 2 and Intra-EXP 3 experiments, it is found that both types of PAs are perfectly classified (100% TDR) by the OCT and NIR modalities. There are a few errors when detecting cosmetic contact PAs using the VIS modality (98.63% TDR). So,

Table 4.3: Data distribution among train, validation and test sets for all experiments (intra-attack and cross-attack scenarios). Here, CC is Cosmetic Contacts.

Experiments	**Train Set**		**Validation Set**		**Test Set**	
	Bonafide	**PAs**	**Bonafide**	**PAs**	**Bonafide**	**PAs**
Intra-EXP 1 (Both Artificial Eyes & CC)	404	100	101	25	218	54
Intra-EXP 2 (Only Artificial Eyes)	435	35	145	12	146	12
Intra-EXP 3 (Only CC)	435	72	145	24	146	24
Cross-EXP 1 (CC are unknown)	435	41	145	18	146	120
Cross-EXP 2 (Artificial eyes are unknown)	435	84	145	36	146	59

in the intra-attack scenario, where attacks are known and used during training, the OCT modality perfectly separates (100% TDR at 0.2% FDR) bonafide and PA iris images by a higher margin compared to the NIR and VIS modalities.

4.6.2 Cross-attack Setup and Results

To perform the cross-attack (generalization to unknown attacks) analysis, two experiments are conducted: Cross-EXP 1 and Cross-EXP 2. In the first experiment (Cross-EXP 1), training is performed on bonafide and artificial eye images, and testing is done on bonafide and cosmetic contact lens images. Bonafide images are split in a 60:20:20 ratio for the training, validation and test sets, respectively. Artificial eye images are split in a 70:30 ratio for the training and validation sets, respectively. All cosmetic contact images constitute the test set. In the second experiment (Cross-EXP 2), training is performed on bonafide and cosmetic contact lens images, and testing is done on bonafide and artificial eye images. Bonafide images are split in the same way as Cross-EXP 1. Cosmetic contact lens images are split in a 70:30 proportion for the training and validation sets, respectively. All artificial eye images are used in the test set. Further details of both the experimental setups are given in Table 4.3. The TDR at 0.2% FDR and the ACER for VGG19, ResNet50 and DenseNet121 architectures are provided in Table 4.4. ROC curves of all three architectures for the

Table 4.4: TDR (%) at 0.2% FDR and ACER of all experiments (intra-attack and cross-attack) when using VGG19, ResNet50 and DenseNet121 architectures.

Experiments	Evaluation Measure	VGG19			ResNet50			DenseNet121		
		OCT	NIR	RGB	OCT	NIR	RGB	OCT	NIR	RGB
Intra-EXP 1	ACER	0.08 ± 0.15	**0.02 ± 0.01**	0.09 ± 0.03	**0.00 ± 0.00**	0.00 ± 0.01	0.08 ± 0.00	0.02 ± 0.03	**0.02 ± 0.02**	0.07 ± 0.02
(Both Artificial & CC)	TDR	**100 ± 0.00**	97.99 ± 2.66	82.58 ± 6.88	**100 ± 0.00**	97.33 ± 3.88	89.62 ± 3.62	**100 ± 0.00**	97.66 ± 3.26	86.66 ± 3.59
Intra-EXP 2	ACER	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.04	**0.00**	**0.00**	**0.00**
(Only Artificial Eyes)	TDR	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
Intra-EXP 3	ACER	**0.00**	**0.00**	0.03	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.03
(Only CC)	TDR	**100**	**100**	95.83	**100**	**100**	**100**	**100**	**100**	**100**
Cross-EXP 1	ACER	0.39	**0.01**	0.19	0.20	**0.01**	0.27	0.16	**0.01**	0.30
(CC are unknown)	TDR	21.66	**97.58**	26.66	92.50	**98.38**	15.00	84.16	**98.38**	11.66
Cross-EXP 2	ACER	0.06	**0.03**	0.04	**0.01**	0.02	0.07	0.05	**0.01**	0.04
(Artificial eyes are unknown)	TDR	86.44	**98.38**	93.22	94.91	**96.77**	81.35	94.91	**96.77**	91.52

two experiments are shown in Figures 4.6(b) and 4.6(c), 4.7(b) and 4.7(c), and 4.8(b) and 4.8(c), respectively.

In the cross-attack scenario, the best results are observed on NIR images, followed by OCT images and then VIS images. Basically, the OCT and VIS modalities failed in detecting cosmetic contact images when training is performed using artificial eye PAs (see Figures 4.6(b), 4.7(b) and 4.8(b)). The feature sub-spaces of bonafide samples and cosmetic contact lens seem to overlap (middle column of Figure 4.10). However, when classifiers are trained on cosmetic contact images (Figure 4.6(c), 4.7(c) and 4.8(c)), they can detect artificial eye PAs as feature sub-space of artificial eyes seems to be well separated from that of bonafide samples (last column of Figure 4.10). Difficulty in detecting cosmetic contact PAs is also reflected in the Intra-EXP 2 and Intra-EXP 3 experiments. ResNet50 and DenseNet121 architectures are better suited for the cross-attack scenario than the VGG19 network, as a higher number of trainable parameters are present in VGG19 and the training data is insufficient. As the networks are pre-trained on the ImageNet dataset (containing VIS images), trainable parameters converge in the case of VIS and NIR images, but fail to converge for OCT images due to the fundamentally different image modality (Figure 5(a)).

The main findings of the comparative analysis are:

1. In the intra-attack scenario, when PAs are known and used during training, OCT images provide more discriminative information for distinguishing between bonafide and PA samples. However, NIR imaging provides better generalizability across unknown iris PA attacks.

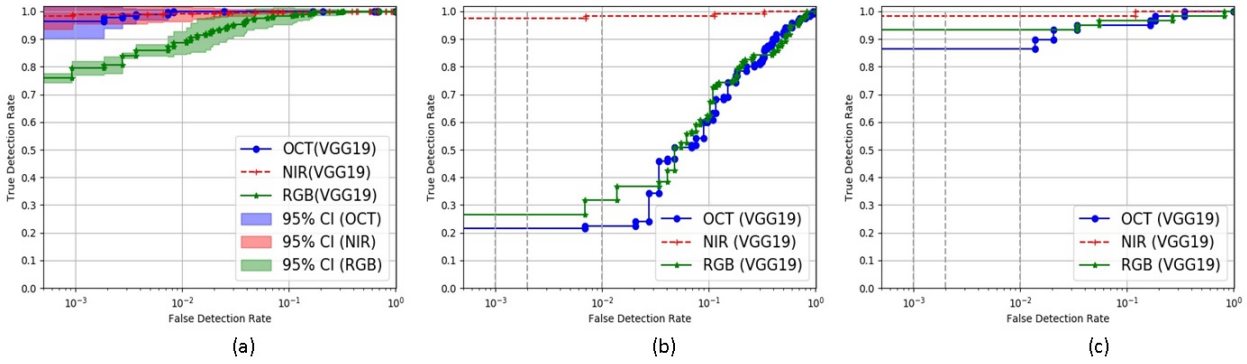


Figure 4.6: ROC curves of (a) Intra-EXP 1, (b) Cross-EXP 1 and (c) Cross-EXP 2 experiments using **VGG19** architecture. The first ROC plot (a) also shows the confidence interval of 95%. NIR imaging is more efficient in discriminating bonafide and PA samples on this network.

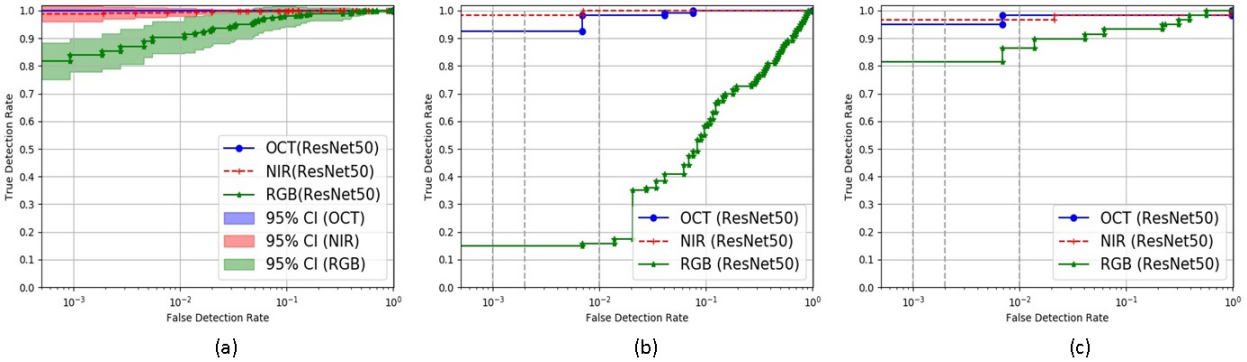


Figure 4.7: ROC curves of (a) Intra-EXP 1, (b) Cross-EXP 1 and (c) Cross-EXP 2 experiments using **ResNet50** architecture. OCT imaging results in better performance in distinguishing bonafide and PA images in the intra-attack scenario (a), whereas NIR imaging performs the best in the cross-attack scenario (b and c).

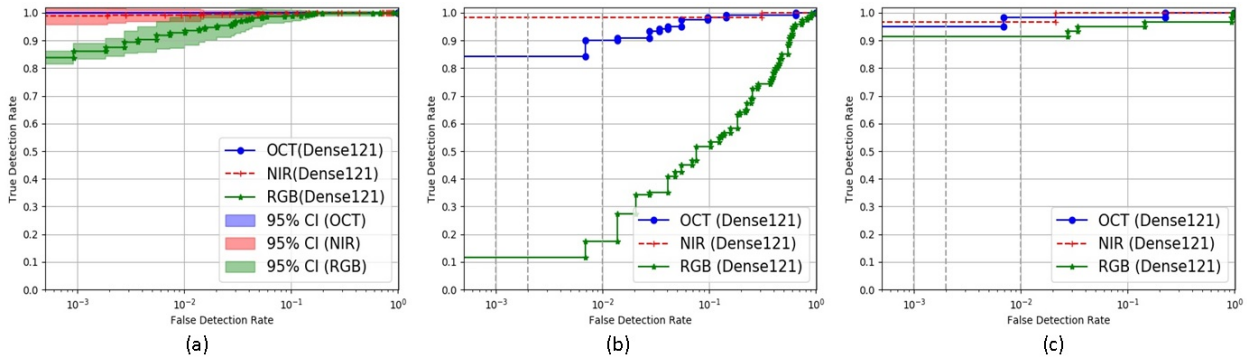


Figure 4.8: ROC curves of (a) Intra-EXP 1, (b) Cross-EXP 1 and (c) Cross-EXP 2 experiments using **DenseNet121** architecture. OCT imaging results in better performance in distinguishing bonafide and PA images in the intra-attack scenario (a), whereas NIR imaging performs the best in the cross-attack scenario (b and c).

2. Cosmetic contact PAs are difficult to detect compared to artificial eyes, especially on VIS images.

3. ResNet50 and DenseNet121 architectures are well-suited for iris PA detection in the OCT imaging modality possibly due to the smaller number of trainable parameters compared to VGG-19.

4.7 CNN Visualization

The performance of all three architectures is nearly perfect on OCT and NIR images. To further analyze the results, we generate heatmaps [245] and t-SNE plots [280]. Heatmaps provide the salient regions in OCT, NIR and VIS images where the classifier (ResNet50) focused on, in order to discriminate PAs from bonafide samples. Heatmaps are generated using Grad-CAM [245]. Grad-CAM uses a gradient of the loss function and backpropagates it through the convolutional layers to generate activations on the input image. OCT, NIR and VIS images of a bonafide, artificial eye and cosmetic contact lens are shown along with their heatmaps in Figure 4.9. In the case of OCT images (Figure 4.9(a)), the heatmap of the bonafide image highlights the iris regions, which is the most discriminative region compared to OCT PA images. The heatmap of an artificial eye image focuses over the outer structure. Cosmetic contact lens conceals the underlying iris pattern (partially or fully), which causes the focus to shift over to the corneal region corresponding to the pupil. In the case of NIR and VIS imaging (Figure 4.9(a) and 4.9(b)), heatmaps of bonafide sample focus over the iris pattern. For an artificial eye image, the heatmap is activated all over the image, whereas for a textured contact lens more emphasis is given to the circumference of the iris. Different regions of focus for different categories (bonafide and PA) aid the CNN architecture to discriminate between them.

After visualizing activations on the input image, we also visualize the CNN features using a t-SNE plot [280]. The CNN features are extracted from the average pooling layer (penultimate layer, a layer before the last fully connected layer) of the ResNet50 architecture. The dimensionality of the features is 2048, which is reduced to two dimensions using t-Distributed Stochastic Neighbor

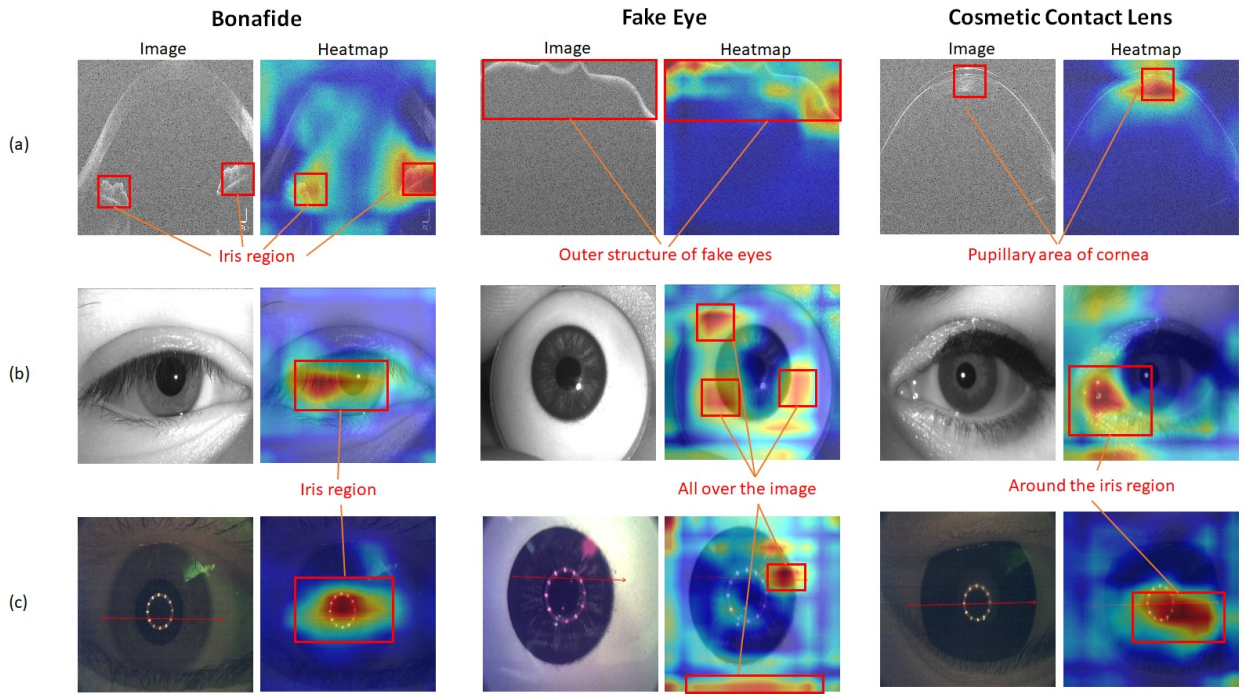


Figure 4.9: (a) OCT, (b) NIR and (c) VIS images and their corresponding fixation regions for bonafide, artificial eyes and cosmetic contact lens samples. Red in the heatmaps represents high priority (high CNN activations) regions considered by the CNN architecture. Blue represents low priority regions. Red boxes mark the high priority regions. Different regions of focus help the CNN architecture to differentiate between bonafide and PA iris images.

Embedding (t-SNE). The t-SNE plots are shown in Figure 4.10. These t-SNE plots correspond to Intra-EXP 1 (first column), Cross-EXP 1 (second column) and Cross-EXP 2 (third column) test data. Distribution of bonafide, artificial eyes and cosmetic contact images are observed to be well separated in OCT imaging in the case of Intra-EXP 1 and Cross-EXP 2 experiments. Separation of these features is also prominent in NIR imaging under the cross-attack scenario (Cross-EXP 1 and Cross-EXP 2). Features in the case of Cross-EXP 1 experiment overlap for VIS images. These plots substantiate our observations that OCT imaging works efficiently in the intra-attack scenario and moderately in the cross-attack scenario, while NIR imaging generalizes well in the cross-attack scenario.

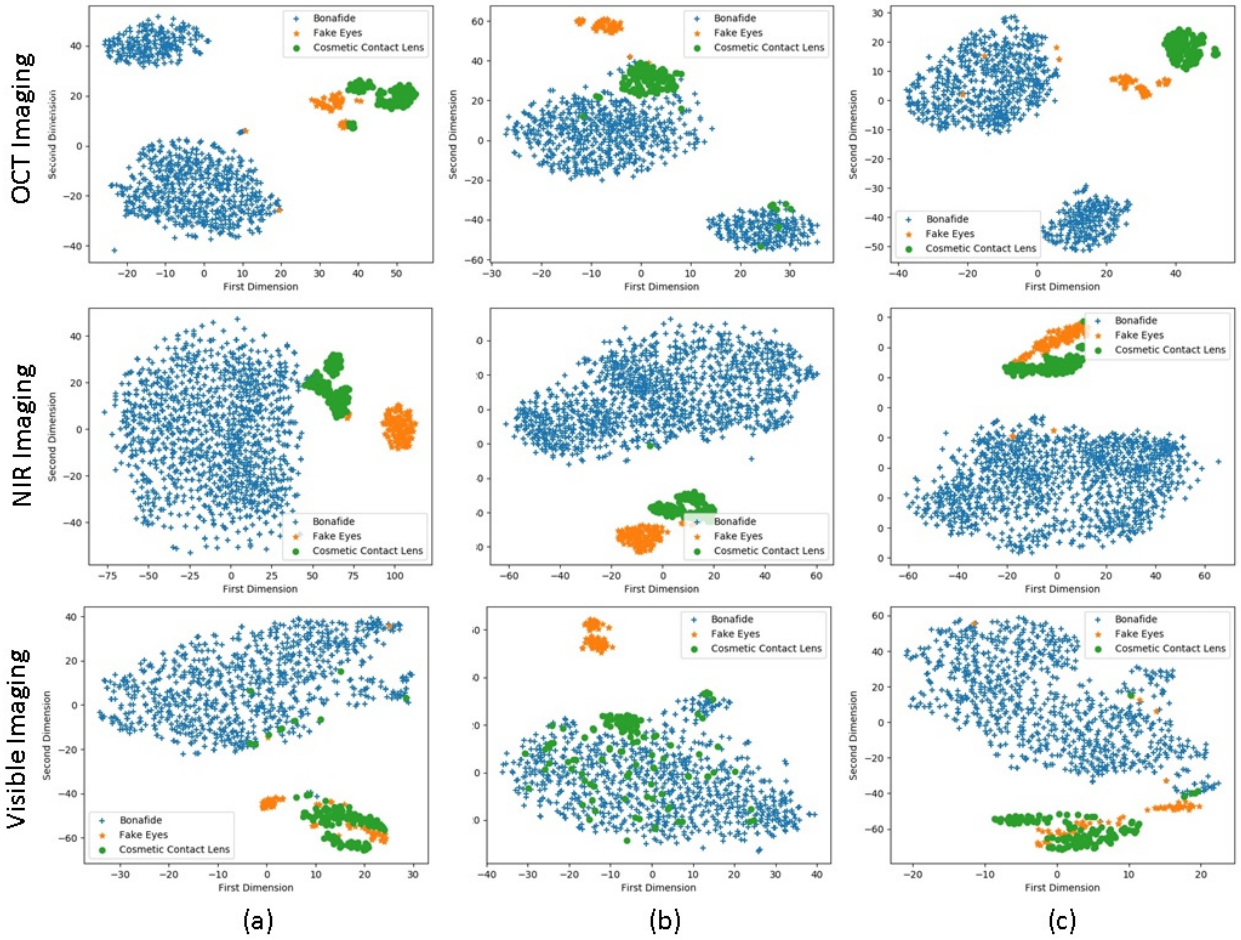


Figure 4.10: t-SNE plots of Intra-EXP 1, Cross-EXP 1 and Cross-EXP 2 test data pertaining to OCT, NIR and VIS imaging. 2048 dimensions of features from the average pooling layer (penultimate layer) of ResNet50 network are reduced to two dimensions for visualization. Features of bonafide and PAs from OCT images are well separated in Intra-EXP 1 and Cross-EXP 2 experiments. NIR images show good separation in all three experiments. Features from VIS images are overlapping between the bonafide and PA categories (especially in the Cross-EXP 2 experiment). More the separation of features, better the classification.

4.8 Conclusion and Future Work

In this chapter, we described the use of the OCT imaging modality for iris PA detection. By comparative analysis against other imaging modalities (traditional NIR and VIS), we determined that OCT is a viable solution for iris PA detection. Extensive experiments were conducted both in the intra-attack and cross-attack scenarios using three state-of-the-art deep architectures, and results were analyzed using CNN visualizations (heatmaps and t-SNE plots). Future work will involve collecting OCT data from more subjects and other types of PAs. Hardware cost continues to be a barrier for the use of OCT in iris recognition applications. However, as sophisticated presentation attacks are launched in the future, the OCT modality is likely to be of great benefit.

CHAPTER 5

ROBUSTNESS OF DEEP NEURAL NETWORKS

5.1 Introduction

In this chapter, we empirically analyze the robustness of iris presentation attack detection (PAD) models by manipulating their architectural parameters. Here, we consider three state-of-the-art architectures (VGG [255], ResNet [105], and DenseNet [122]) under three types of parameter perturbations (Gaussian noise, weight zeroing and weight scaling). We apply the perturbations in two settings: over all the layers of a network simultaneously and over each layer at a time. Our main contributions are as follows:

1. We perform robustness analysis of three state-of-the-architectures (VGG [255], ResNet [105] and DenseNet [122]) against parameter perturbations.

2. We apply a large number of parameter perturbations (three types of perturbations and its variant in two settings) to analyze the robustness of deep neural networks in the context of iris presentation attack detection.

3. We leverage the robustness analysis to propose better performing ensemble models.

4. We perform experiments using five datasets. Three of the datasets (IARPA, NDCLD-2015, Warsaw Postmortem v3) are used for training, whereas the others (LivDet-Iris-2017 and LivDet-Iris-2020) are used for testing. This represents a cross-dataset scenario, where training and testing are performed on different datasets.

The rest of the chapter is organized as follows: Section 5.2 discusses the existing work related to the robustness analysis of DNNs, Section 5.3 provides the details of various parameter perturbations used for the robustness analysis, Section 5.4 describes the application scenario considered in this work, Section 5.5 explains the dataset and experimental setup, Section 5.6 provides the robustness

analysis of the three architectures against considered parameter perturbations, and Section 5.7 describes how we leverage the robustness analysis to generate an ensemble of perturbed models for improving performance. Finally, Section 5.8 summarizes the chapter and provides future directions.

5.2 Related Work

Deep Neural Networks (DNNs) have revolutionized the machine learning field through their superior performance in various tasks especially in the field of computer vision [105, 122, 255], natural language processing [75], and speech technology [74]. In essence, a DNN comprises a sequence of layers containing trainable parameters (weights and bias) to learn a complex mapping between input signals and output labels. For deploying DNNs in real-world applications, it is crucial to analyze their robustness or sensitivity to hardware/sensor noise introduction [51], environment changes [276] and adversarial attacks [94]. Robustness analysis also helps in building a quantized-weights model with commensurate performance [102, 293].

In the literature, robustness analysis of DNNs has been performed by perturbing either the input signal or the architectural parameters. The work in [83, 96, 137, 181, 191, 259] analyze DNN robustness by manipulating the input signals, whereas the work in [102, 253, 276, 293, 297] perturb architectural parameters to analyze robustness. Yeung *et al.* [309] provide a detailed sensitivity analysis of neural networks over input and parameter perturbations. In this work, we focus on the robustness analysis of DNNs when architectural parameters are perturbed.

The authors in [253, 276, 293, 297] provide a theoretical robustness analysis based on parameter perturbations. Shu and Zhu [253] propose an influence measure motivated by information geometry to quantify the effects of various perturbations to input signals and network parameters on DNN classifiers. Xiang *et al.* [297] design an iterative algorithm to compute the sensitivity of a DNN layer by layer, where sensitivity is defined as "the mathematical expectation of absolute output variation due to weight perturbation with respect to all possible inputs" [297]. Tsai *et al.* [276] study the robustness of the pairwise class margin function against weight perturbations. Weng *et al.* [293] compute a certified robustness bound for weight perturbations, within which a neural

network will not make erroneous outputs. In addition, they also identify a useful connection between the developed certification and the problem of weight quantization.

Our work is motivated from [51], where they also empirically analyze the robustness of the pre-trained AlexNet and VGG16 networks to internal architecture and weight perturbations. However, our work is vastly different. First, we extend the work by evaluating the robustness of more recent DNN architectures: VGG, ResNet, and DenseNet. Second, we perform additional weight manipulations (weight scaling and perturbations over the entire network) in the robustness analysis. Third, we leverage the findings from the robustness analysis and propose an ensemble of perturbed models for improved performance without any further training.

5.3 Parameter Perturbations

We explore the stability of neural networks by perturbing their architectural parameters (weights and bias). From now on, we use the terms 'architectural parameters', 'parameters', and 'weights' interchangeably. To measure the stability, we consider the change in the performance of the DNN when weights are perturbed. Let n input samples be $\{x_1, x_2, ..., x_n\}$ and their output as $\{y_1, y_2, ..., y_n\}$. Here, we labeled the positive class as '1' and the negative as '0'. The predicted output values from a DNN approximator are $\{f(x_1, W_{org}), f(x_2,$
$W_{org}), ..., f(x_n, W_{org})\}$, where W_{org} are the learned parameters. We measure the performance of the DNN in terms of True Detection Rate (TDR). TDR is a percentage of positive samples correctly classified as

$$TDR_{org} = \frac{\sum_i^n (f(x_i, W_{org}) > T)}{\sum_i^n y_i} * 100 \tag{5.3.1}$$

where, T is the threshold. The input sample with a predicted value above the threshold is considered a positive class. On weight perturbation, we estimate the output as $\{f(x_1, W_{mod}), f(x_2, W_{mod}),$
$..., f(x_n, W_{mod})\}$, where W_{mod} are the perturbed parameters. We then use these predicted values to measure the performance of DNN (TDR_{mod}). The higher the change in the performance, the lower the robustness of the neural network to the particular perturbation.

We perturb the parameters in two settings: manipulating parameters of all layers simultaneously

and manipulating parameters one layer at a time. The first setting aims to understand the overall robustness of DNNs, whereas the second setting examines which layer has more impact on the stability of the model. We consider three types of perturbations: Gaussian noise manipulation, weight zeroing, and weight scaling. These perturbations resemble noise introduction due to (a) defects in hardware implementations of neural networks [174], and (b) adversarial attacks [94]. Details of these perturbations are as follows:

1. Gaussian Noise Manipulation: Here, we manipulate the original parameters of the layers by adding Gaussian noise sampled from a normal distribution of zero mean and scaled standard deviation. We control the scaling of the standard deviation by the scalar factor α. The modified parameters are defined as

$$W_{mod} = W_{org} + N(0, \boldsymbol{\alpha} * \sigma(W_{org})) \tag{5.3.2}$$

where, W_{org} are the original parameters, W_{mod} are the modified parameters, and $N(\mu, \sigma)$ is the normal distribution. We calculate $\sigma(W_{org})$ for a particular layer by first flattening the parameter tensor to a 1-D array and then computing the standard deviation. So, the standard deviation and the Gaussian noise distribution will differ for each layer. Consequently, the absolute perturbations applied to the different layers also vary. However, relative perturbations are the same across layers.

1. Weight Zeroing: In the second manipulation, we randomly select a certain proportion of parameters and set them to zero. The portion of parameters is determined by a scalar factor β. The modified parameters are represented as

$$W_{org}[random(\boldsymbol{\beta}, W_{org})] = 0 \tag{5.3.3}$$

where, $random(.,.)$ is the function that returns the index of β proportion of randomly selected parameters from the original set of parameters. We also define another version of weight zeroing, where weights are first sorted, and then the β proportion of low-magnitude weights is set to zero.

3. Weight Scaling: The third perturbation scales the original parameters by a scalar factor γ as

$$W_{mod} = \boldsymbol{\gamma} * W_{org}. \tag{5.3.4}$$

5.4 Application Scenario

We perform our robustness analysis in the context of iris presentation attack detection (PAD). A presentation attack (PA) occurs when an adversary presents a fake or altered biometric sample such as printed eyes, plastic eyes, or cosmetic contact lenses to circumvent the iris recognition system [3]. Our application is to detect these PAs launched against an iris system. We formulate the detection problem as a two-class problem based on DNNs, where the input is a near-infrared iris image and the output is a PA score that is assigned one of two labels: "bonafide" or "PA".

5.5 Datasets and Experimental Setup

Table 5.1: Summary of training and test datasets along with the number of bonafide and PA images present in the datasets. The information about the sensors used to capture images is also provided. Here, "K. Test" means a known test set of the dataset, and "U. Test" means an unknown test set (see text for explanation).

Train/Test	**Train**			**Test**						
Datasets	**IARPA**	**NDCLD -2015**	**Warsaw PostMortem v3**	**LivDet-Iris-2017**						**LivDet-Iris-2020**
Dataset Subsets				**Clarkson (Cross-PA)**	**Warsaw (Cross-sensor)**		**Notre Dame (Cross-PA)**		**IIITD-WVU (Cross-Dataset)**	
Splits				**Test**	**K. Test**	**U. Test**	**K. Test**	**U. Test**	**Test**	
Bonafide	19,453	-	-	1,485	974	2,350	900	900	702	5,331
Print	1,005	-	-	908	2,016	2,160	-	-	2,806	1,049
Cosmetic Contacts	1,187	2,236	-	765	-	-	900	900	701	4,336
Artificial Eyes	1,804	-	-	-	-	-	-	-	-	541
Electronic Display	51	-	-	-	-	-	-	-	-	81
Cadaver Eyes	-	-	1,200	-	-	-	-	-	-	1,094
Sensor	COTS Iris Sensors x3[1]	IrisGuard AD100, IrisAccess LG4000	IriShield MK2120U	IrisAccess EOU2200	IrisGuard AD100	Aritech ARX-3M3C, Fujinon DV10X7.5A, DV10X7.5A-SA2 lens B+W 092 NIR filter	IrisGuard AD100, IrisAccess LG4000		IriShield MK2120U	Iris ID iCAM7000, IrisGuardAD100, IrisAccess LG4000, IriTech IriShield

The training data we use to build our iris PAD models are IARPA [2], NDCLD-2015 [267] and Warsaw PostMortem v3 [275] datasets. The IARPA dataset is a proprietary dataset collected under the IARPA Odin program [2]. It consists of 19,453 bonafide irides and 4,047 presentation attack (PA) samples. From the NDCLD-2015 dataset, we use 2,236 cosmetic contact lens images for training. From the Warsaw PostMortem v3 dataset, 1,200 cadaver iris images from the first 37 cadavers are used for training. Testing is performed on the LivDet-Iris-2017 [304] and LivDet-Iris-2020 [61] datasets. Both of these are publicly available competition datasets for evaluating iris presentation attack detection. The LivDet-Iris-2017 dataset [304] consists of four subsets:

Clarkson, Warsaw, Notre Dame, and IIITD-WVU. All subsets contain train and test partitions, and we use only the test partition. Warsaw and Notre Dame subsets further contains two splits in the test partition: 'Known' and 'Unknown'. The 'Known' split corresponds to the scenario, where PAs of the same type or images from similar sensors are present in both train and test partitions, while the 'Unknown' split contains different types of PAs or images from different types of sensors in the train and test partitions. Our experimental setup corresponds to a cross-dataset scenario as we use different datasets for training and testing. In the case of LivDet-Iris-2020 [61], we use the entire dataset for testing, and this scenario also corresponds to the cross-dataset. Table 5.1 describes all training and test sets along with the types of PAs and images present in them.

We use three iris PA detectors for stability analysis. Two of the detectors utilize VGG19 [255] and ResNet101 [105] networks as their backbone architecture. The third detector is D-NetPAD [249], where the backbone architecture is DenseNet161 [122]. The D-NetPAD shows state-of-the-art performance on both LivDet-Iris-2017 and LivDet-Iris-2020 iris PAD competitions [61, 249]. The input given to these models is a cropped iris region resized to 224×224. For training, we initialize the model with the weights from the ImageNet dataset [72] and then fine-tuned the models using the training datasets described above. The learning rate was set to 0.005, the batch size was 20, the number of epochs was 50, the optimization algorithm was stochastic gradient descent with a momentum of 0.9, and the loss function used was cross-entropy.

We measure the robustness of these DNNs by evaluating their performance as a function of the weight perturbations. The performance is estimated in terms of TDR (%) at 0.2% False Detection Rate (FDR). FDR is the percentage of bonafide samples incorrectly classified as PAs. In Table 5.3, the row corresponding to the 'Original' method reports the performance of these models on the LivDet-Iris-2017 and LivDet-Iris-2020 datasets *before* weights were perturbed. On the LivDet-Iris-2017 dataset, ResNet101 performs the best (average 74.55% TDR), whereas on the LivDet-Iris-2020 dataset, D-NetPAD performs the best (90.22% TDR). We also provide information about the number of weights and bias parameters present in all three models (Table 5.2). The VGG19 architecture has the highest number of parameters, followed by the ResNet101

Table 5.2: The number of parameters (weights and bias) present in all convolutional layers of the VGG19, ResNet101, and D-NetPAD architectures.

Architecture	**VGG19**	**ResNet101**	**D-NetPAD**
Weights	139,570,240	42,451,584	26,366,448
Bias	19,202	52,674	109,970
Total	139,589,442	42,504,258	26,476,418

architecture.

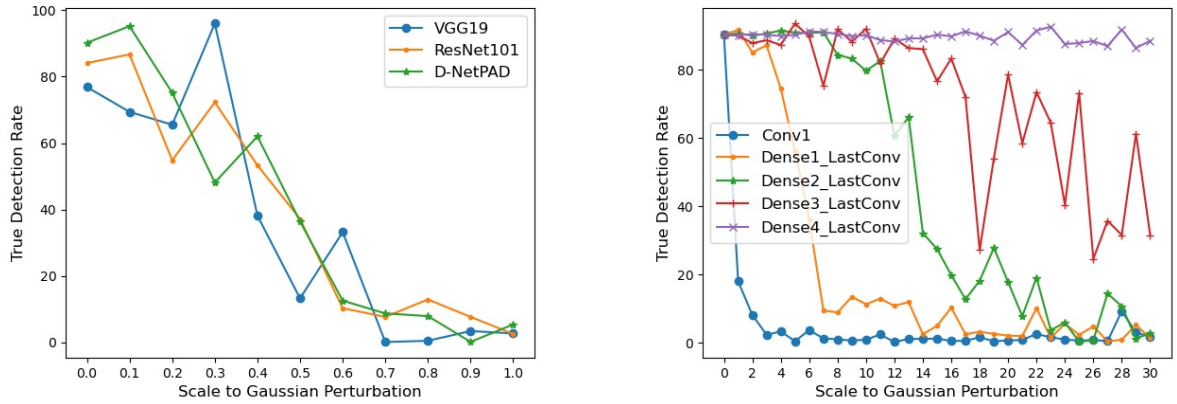
5.6 Robustness Analysis

5.6.1 Gaussian Noise Addition

The Gaussian noise manipulation involves the addition of Gaussian noise to the original parameters. Figure 5.1 (a) shows the performance of all the networks when we perturb parameters of all layers with the Gaussian noise. The scale factor (α) used to modify the standard deviation is shown on the x-axis. From a trend standpoint, the performance of all networks decreases with an increase in the standard deviation. However, this decrease is not linear. In fact, there are some performance gains at certain scales. These scales are different for different networks. For instance, the VGG19 network shows improvement for α = 0.3, 0.6, and 0.9, ResNet101 for α = 0.1, 0.3, and 0.9, and D-NetPAD for α = 0.1, 0.4 and 1.0. Surprisingly, certain scales give higher performance than the original model, such as 0.1 scale for the ResNet101 and D-NetPAD models, and 0.3 scale for the VGG19 model. It should be noted that all three networks are not robust to Gaussian noise perturbations, and we cannot conclude which network is comparatively robust under these weight perturbations.

We further analyze the impact of perturbation at different layers on the performance of the models. We manipulate the parameters one layer at a time and observe the performance change. For the layer-wise analysis, we show the results of only the D-NetPAD model since the other two models also show similar performance trends. In the case of D-NetPAD, we select the first convolution layer and the last convolution layers of four dense blocks for perturbation. Figure 5.1

[1]Specific sensor names withheld at sponsor's request



(a) (b)

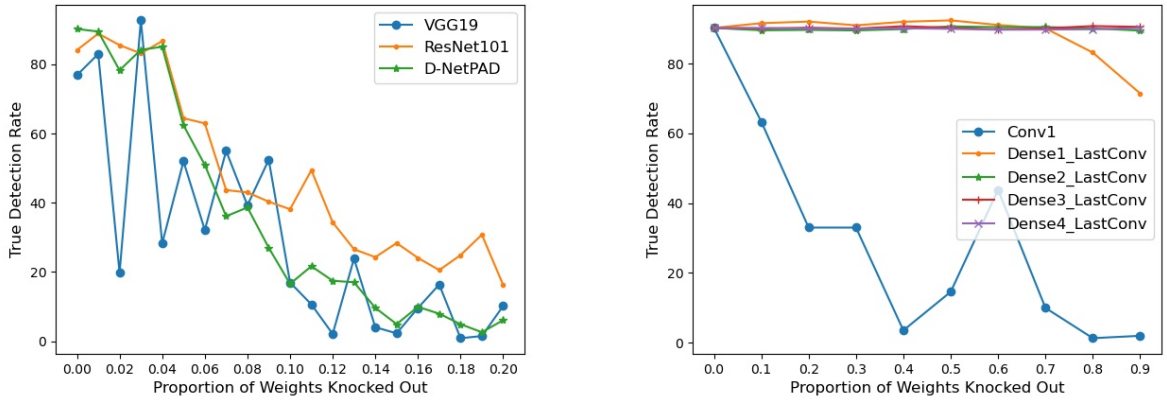
Figure 5.1: Gaussian noise manipulation: (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when weights and bias parameters of the entire network are perturbed. (b) Performance of D-NetPAD when the individual layer's parameters (weights and bias) are perturbed. Here, Conv1 means the first convolution layer of the D-NetPAD, Dense1_LastConv means the last convolution layer of the first dense block, and so on.

(b) shows the performance of D-NetPAD when the individual layer's parameters are perturbed. We observe that the initial layers have more influence on the performance of the D-NetPAD compared to the later layers. The model is highly robust to the perturbations in the last convolution layer of the fourth dense block, even at a scale factor of 30. Cheney *et al.* [51] also observe the higher impact of perturbations in the initial layers on the performance. This is because the perturbations in the initial layers impact all the subsequent layers, resulting in a substantial decrease in performance. Change in middle layers exhibit large fluctuations in performance compared to the initial and later layers.

5.6.2 Weight Zeroing

The weight zeroing manipulation involves random selection of a particular fraction of weight parameters and setting them to zero. Figure 5.2 (a) shows the performance of all three architectures when we manipulate the entire set of network parameters, while Figure 5.2 (b) shows the performance of D-NetPAD when we perturb individual layers. Similar conclusions can be drawn from Figure 5.2 (a) as drawn from Figure 5.1 (a) that the overall performance of all three architectures de-

creases with an increase in the proportion of weights set to zero. **However, certain perturbations give improved performance.** For example zeroing 3% of weights improves the VGG19 network performance from 76.87% TDR (original) to 92.70% TDR, in the case of ResNet101 zeroing 3% of weights improves performance from 84.11% TDR (original) to 88.88% TDR. Again, all three networks are not robust to the zeroing out of randomly selected weights from the entire network.



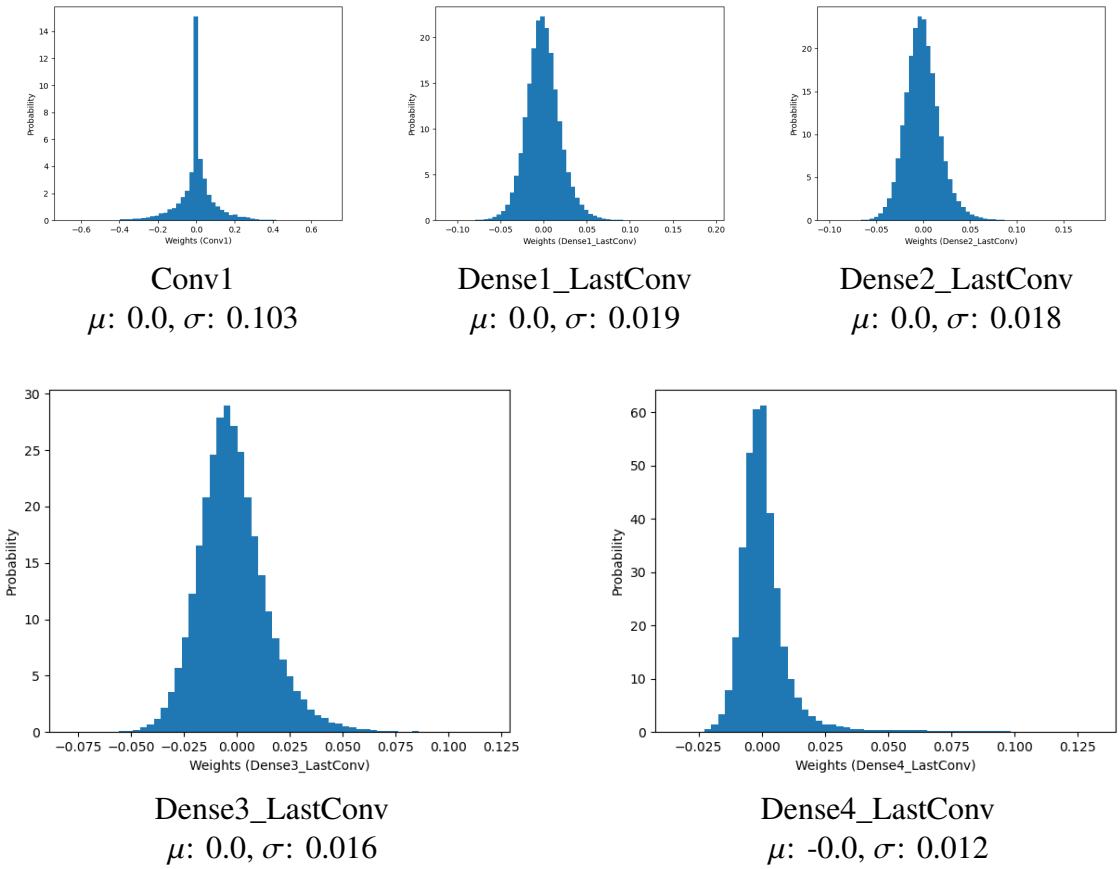
(a) (b)

Figure 5.2: Weight zeroing manipulation: (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when parameters of the entire network are perturbed. (b) Performance of D-NetPAD when the individual layer's parameters are perturbed.

In the layer-wise setup (Figure 5.2 (b)), the performance of D-NetPAD is stable except for the first convolution layer. This is due to the fact that the original weights of the convolution layers have zero mean and small standard deviation ranging from 0.10 (first convolution layer) to 0.01 (last convolution layer) as shown in Figure 5.3. A similar performance trend is observed in the VGG19 and ResNet101 networks as well.

To further analyze the effect of weight zeroing, we assess its three variants - first is to set low-magnitude weights to zero, second sets high-magnitude weights to zero and in the third randomly selected weights to make them zero and non-zero weights are scale to factor 5. The details of these variants are as follows:

1. Since most of the original weights are already close to 0, we set low-magnitude weights to zero. Figure 5.4 (a) shows the performance of all architectures when we manipulate the



Conv1
μ: 0.0, σ: 0.103

Dense1_LastConv
μ: 0.0, σ: 0.019

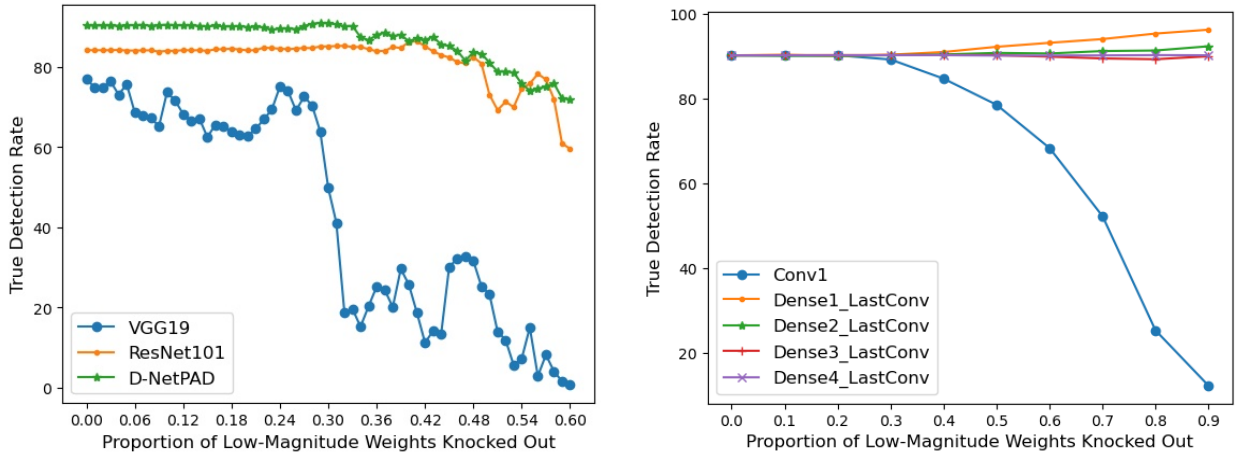
Dense2_LastConv
μ: 0.0, σ: 0.018

Dense3_LastConv
μ: 0.0, σ: 0.016

Dense4_LastConv
μ: -0.0, σ: 0.012

Figure 5.3: Weight distribution of different layers of the trained D-NetPAD architecture. Mean (μ) and standard deviation (σ) are provided below each distribution.

entire network in this fashion, while Figure 5.4 (b) shows the performance of D-NetPAD on layer-wise manipulation. ResNet101 and D-NetPAD networks are robust to this manipulation as zeroing out even 33% of all weights does not affect their performance. VGG19 also shows robustness with only a 10% drop in performance, though its performance is not as stable as the ResNet101 and D-NetPAD networks. Figure 5.4 (b) shows the stability of the D-NetPAD on layer-wise perturbations. Zeroing out even 30% of the first convolution layer weights does not impact its performance. Remarkably, the manipulation in the last convolution layer of the first and second dense blocks shows a linear increase in performance. The performance of D-NetPAD increases from 90.22% TDR to 96.28% TDR upon manipulating the last convolution layer of the first dense block. This suggests that we could zero out low-magnitude weights and reduce the size of the model without affecting its performance.

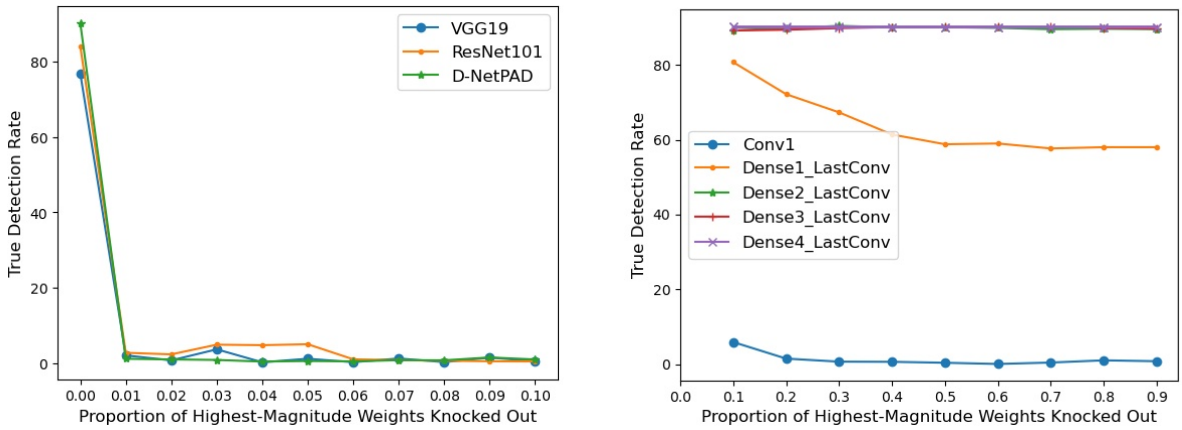


(a) (b)

Figure 5.4: Variant of the weight zeroing manipulation (low-magnitude weights are set to zero): (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when parameters of the entire network are perturbed. (b) Performance of D-NetPAD when individual layer's parameters are perturbed.

2. The second variant make high-magnitude weights to zero. Figure 5.5 (a) shows the performance of all architectures when high-magnitude weights are set to zero on the entire network, while Figure 5.5 (b) shows the performance of D-NetPAD on layer-wise manipulation. There is a drastic drop in the performance of all three architectures when high-magnitude weights are set to zero. It shows that high-magnitude weights are high priority parameters. In the layer-wise analysis, manipulation in the first convolution layer and layers of DenseBlock1 show a drop in the performance, whereas manipulation in other higher layers does not impact the performance.

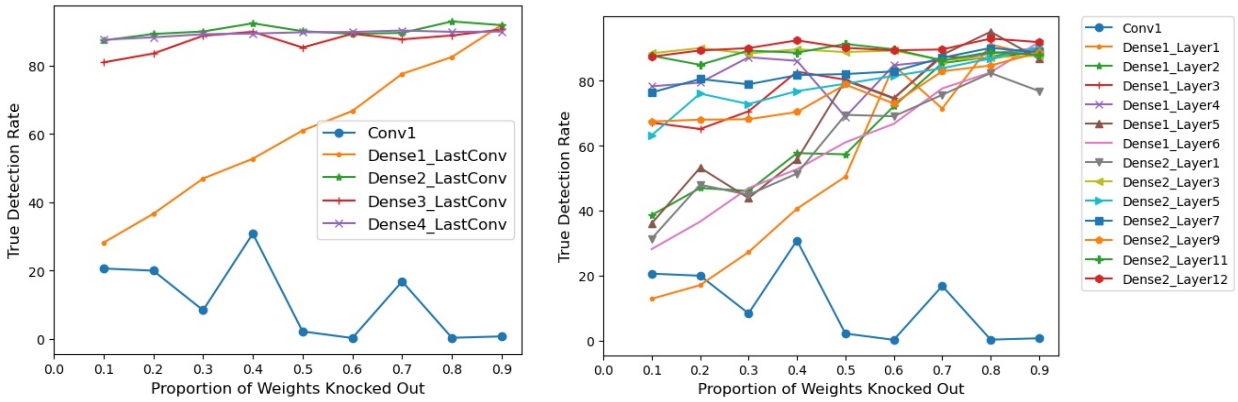
3. The third variant mimics the operation of Dropout layer, where randomly selected weights are set to zero and non-zero weights by scaled to the factor five. Figure 5.6 (a) shows the performance of D-NetPAD when layer-wise manipulation is applied. The last layer of the DenseBlock1 layer shows a different trend of increasing performance with an increase in the manipulation magnitude. To explore it further, we plot the performance against layers manipulation between the first convolution layer and the last convolution layer of DenseBlock2 (Figure 5.6 (b)). We found that the layers in DenseBlock1 and DenseBlock2 show a similar



(a) (b)

Figure 5.5: Variant of the weight zeroing manipulation (high-magnitude weights are set to zero): (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when parameters of the entire network are perturbed. (b) Performance of D-NetPAD when individual layer's parameters are perturbed.

pattern. This is due to the scaling of non-zero weights whose impact decreases with an increase in weight proportion set to zero. The impact of scaling weights is more than the setting weights to zero.

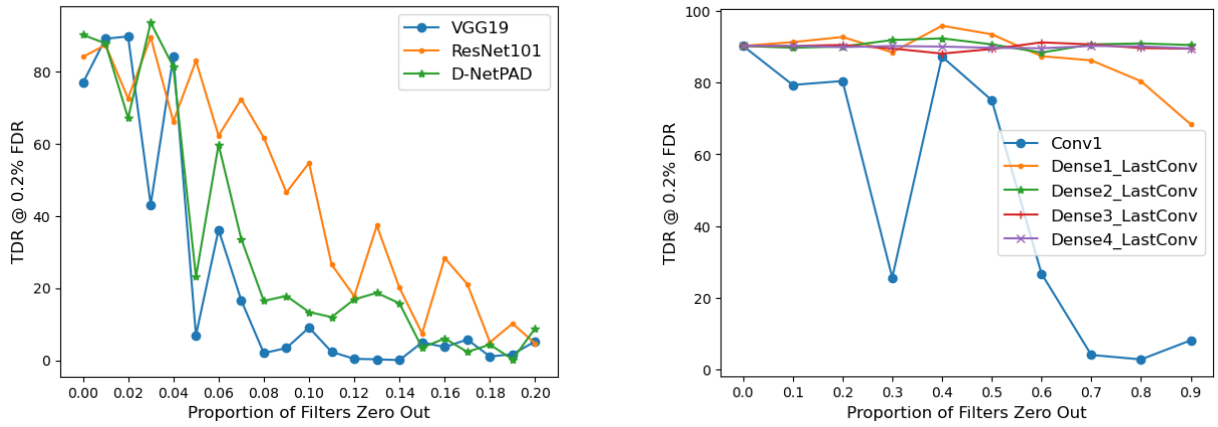


(a) (b)

Figure 5.6: Variant of the weight zeroing manipulation (randomly selected weights are set to zero and non-zero weights are scaled by factor 5): (a) Performance of D-NetPAD when individual layer's parameters are perturbed. (b) Closer look at the performance of D-NetPAD when convolution layers of DenseBlock1 and DenseBlock2 are perturbed.

4. The fourth variant randomly selected filters from the layers and set their all weights to zero.

Figure 5.7 (a) shows the performance of all architectures when randomly selected filters are set to zero on the entire network, while Figure 5.7 (b) shows the performance of D-NetPAD on layer-wise manipulation. Similar conclusions can be drawn from Figure 5.7 (a) as drawn from Figure 5.2 (a) that the overall performance of all three architectures decreases with an increase in the proportion of filters set to zero. However, certain perturbations give improved performance. Overall, all three networks are not robust to zero out randomly selected filters from the entire network. Again from Figure 5.7 (b) performance is robust to the manipulations in later layers compared to initial layers.

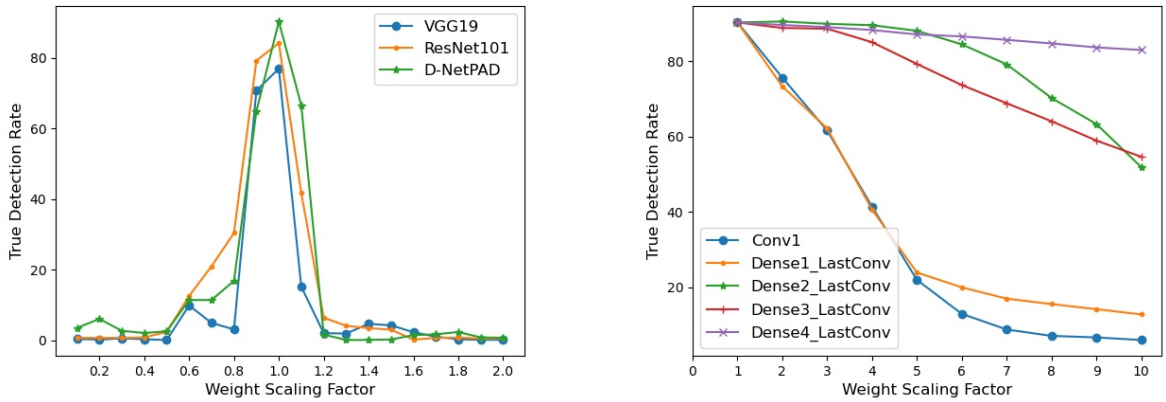


(a) (b)

Figure 5.7: Variant of the weight zeroing manipulation (randomly selected filters are set to zero): (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when filters of the entire network are perturbed. (b) Performance of D-NetPAD when individual layer's parameters are perturbed.

5.6.3 Weight Scaling

This manipulation scales the original parameters with a scalar value. Figure 5.8 (a) shows the performance of all three architectures when we manipulate the entire set of network parameters, while Figure 5.8 (b) presents the performance of D-NetPAD when we perturb specific layers. The performance at scale 1 indicates the original performance without weight perturbations. Weight perturbations across the entire network resulted in a radical drop in performance even with a



(a) (b)

Figure 5.8: Weight scaling manipulation: (a) Performance (TDR at 0.2% FDR) of VGG19, ResNet101, and D-NetPAD when parameters of the entire network are perturbed simultaneously. (b) Performance of D-NetPAD when the individual layer's parameters are perturbed.

small scalar factor (0.8 or 1.1). In the layer-wise manipulation, the initial layers show a higher impact on the performance of D-NetPAD compared to the later layers. The manipulation in the last convolution layer does not impact the performance even at a scaling factor of 10. A similar performance trend is observed on the VGG19 and ResNet101 networks as well.

5.6.4 Findings

Here are the main findings from the aforementioned analysis:

1. All three networks decrease in performance when perturbations are applied over the entire network. However, the networks show stability when low-magnitude weights are set to zero. The scaling of weights has a major negative impact on the performance of networks.

2. Layer-wise robustness analysis shows that perturbations in initial layers impacted the performance to a greater extent compared to the later layers. Initial layer perturbation impacts all the subsequent layers, resulting in drop in performance. Gaussian noise perturbations negatively impacted the performance when applied layer-wise.

3. Certain perturbations improve the performance of network models over the original one.

This observation indicates that the parameters learned by the models during training are not optimum. Hence, there is a further scope for optimizing weights.

4. Zeroing out low-magnitude weights results in better performance as well as reduces the size of the model.

5.7 Performance Improvement

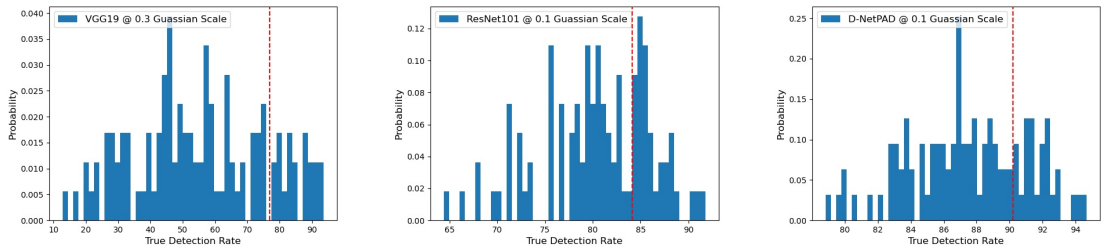
We observe that certain perturbations result in better performance, even higher than that of the original model. We leverage this observation and obtain better performing models using these perturbations without any additional training. In this regard, we explore two directions: the first is to find a single perturbed model which achieves good performance consistently, and the second is to create an ensemble of perturbed models to obtain high performance.

5.7.1 Single Perturbed Model

Certain perturbations result in higher performance than the original one, for instance, 0.1 scale factor of Gaussian manipulation for D-NetPAD and ResNet101 architectures, 0.3 scale factor of Gaussian manipulation for VGG19 architecture, 0.01 proportion of weight zeroing for ResNet101, and 0.03 proportion of weight zeroing for VGG19 architecture. These manipulations involves random selection, so we repeat these manipulations 100 times and plot the performance distributions. Figures 5.9 and 5.10 show the performance distributions corresponding to these manipulations. Approximately 20-40 times, these manipulations result in higher performance than the original one.

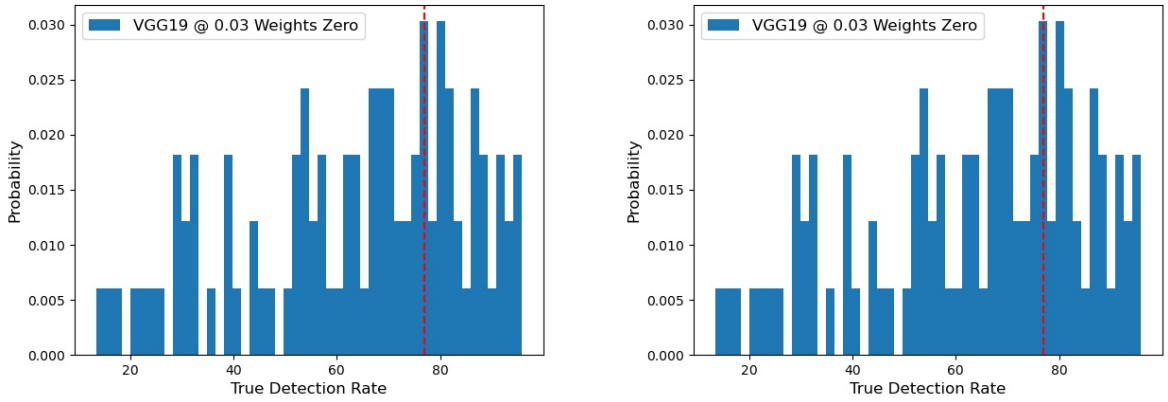
5.7.2 Ensemble of models

The second direction to improve the performance using weight manipulations without further training is an ensemble of models. Ensemble of models better spans the decision space and generalize well on the test data [203]. For initial analysis, we ensemble three perturbed models.



(a) (b) (c)

Figure 5.9: The performance distributions when Gaussian perturbation is applied over the entire architecture at the specified scale on (a) D-NetPAD, (b) ResNet101, and (c) VGG19 architectures.



(a) (b)

Figure 5.10: The performance distributions when weights are set to zero over the entire architecture of (a) ResNet101 and (b) VGG19 for the specified proportion. The red vertical line represents the original performance of the architectures when weights are unperturbed.

Figure 5.11 shows the process of assembling perturbed models. We consider three settings for ensemble - models having the same manipulation and scalar factor, same manipulation but different scalar factors, and different manipulations. In each setting, we also consider manipulations applied on the entire network and to the last convolution layer only.

1. **Same parameter manipulations and scalar factor:** In the first setting, we use the same manipulation and scalar factor to manipulate the parameters of component models. The manipulation is the addition of Gaussian noise sampled from Gaussian distribution having zero mean and 0.1 scaling factor. All three models have the same manipulation, but their weights differ as there involve random sampling from the Gaussian distribution. We repeat

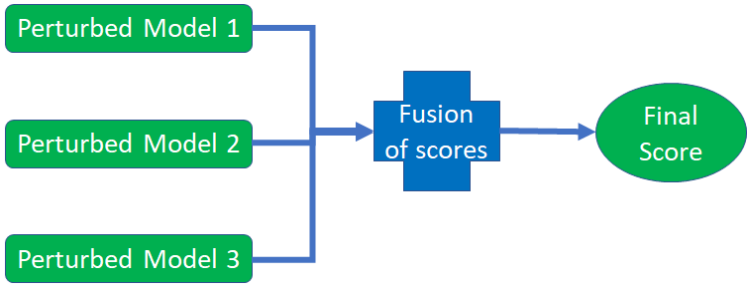
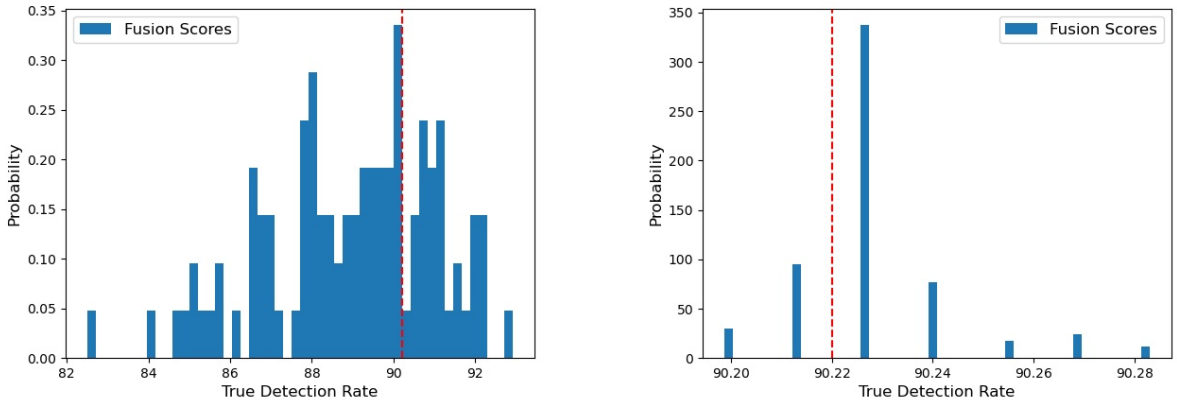


Figure 5.11: Ensemble process of perturbed models to improve the performance of DNN model without undergoing further training.

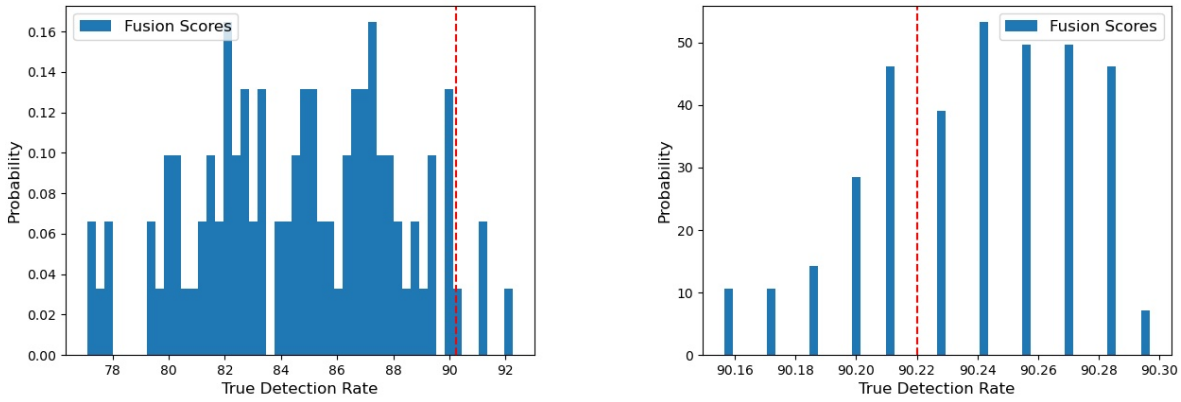
the ensemble models 100 times and plot the performance distribution. Figure 5.12 (a) shows the performance distribution when we manipulate the entire network, whereas Figure 5.12 (b) shows the performance distribution when only last convolution layer manipulated. There are 29 times when TDR is higher than the original performance on the entire network manipulation, whereas there are 79 times when TDR improves over the last convolution layer manipulation. So, ensemble models show a higher chance of improved performance when we manipulate only the last convolution layer.



(a) (b)

Figure 5.12: Performance distributions when three Gaussian noise manipulated D-NetPAD models are ensembled. The Gaussian distribution scaling parameter used in all three models is 0.1. The red vertical line corresponds to the original performance (without weight perturbations). (a) Performance distribution when the entire network is manipulated. In this case, 29 times TDR is higher than the original performance. (b) Performance distribution when only the last convolution layer of DenseBlock4 is manipulated. In this case, 79 times TDR is higher than the original performance.

2. **Same parameter manipulations, but different scalar factor:**In the second setting, we use the same manipulation but different scalar factors to manipulate the parameters of component models. The manipulation used is the addition of Gaussian noise sampled from Gaussian distribution having zero mean and 0.1, 0.2, and 0.3 scaling factors. Figure 5.13 (a) shows the performance distribution when we manipulate the entire network, whereas Figure 5.13 (b) presents the performance distribution on only last convolution layer manipulation. Four times TDR is higher than the original performance when the entire network is manipulated, whereas TDR improves 69 times when we only manipulate the last layer. We draw similar conclusion that the ensemble models show a higher chance of improved performance when we manipulate only the last convolution layer.

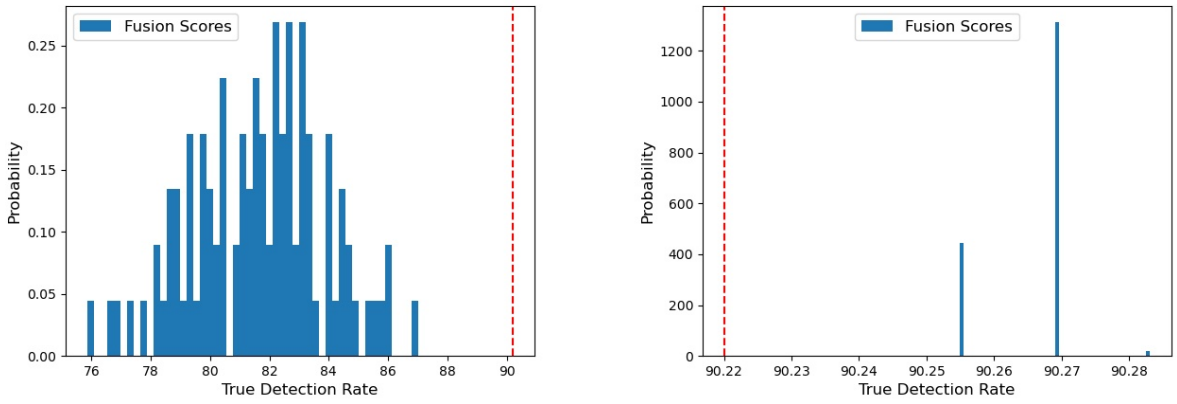


(a) (b)

Figure 5.13: Performance distributions when three Gaussian manipulated D-NetPAD models are ensembled. The Gaussian distribution scaling parameters for the three models are 0.1, 0.2, and 0.3, respectively. The red vertical line corresponds to the original performance (without weight perturbations). (a) Performance distribution when the entire network is manipulated. In this case, four times TDR is higher than the original performance. (b) Performance distribution when only the last convolution layer of DenseBlock4 is manipulated. In this case, 69 times TDR is higher than the original performance.

3. Different manipulations: In the third setting, we utilize three models undergoing different parameter manipulations. The manipulations applied to the component models are: (a) Gaussian noise with a scale factor of 0.1, (b) weight zeroing with 0.01 proportion, and (c) weight scaling with a scalar factor of 1.1. Figure 5.14 (a) shows the performance distribution

when we manipulate the entire network, whereas Figure 5.14 (b) presents the performance distribution we manipulate only the last convolution layer. There are zero times when TDR is higher than the original performance in the case of entire network manipulation, and there are 100 times when TDR is higher than the original in the case of last convolution layer manipulation. Again, ensemble models with only last layer manipulation show a higher chance of improved performance.



(a) (b)

Figure 5.14: Performance distributions when three parameter-manipulated D-NetPAD models are fused undergoing three different types of manipulations. The manipulations in the three models are additive Gaussian Noise (scale factor is 0.1), weight zeroing (proportion is 0.01), and weight scaling (scale factor is 1.1), respectively. The red vertical line corresponds to the original performance (without weight perturbations). (a) Performance distribution when the entire network is manipulated. In this case, zero-times TDR is higher than the original performance. (b) Performance distribution when only the last convolution layer of DenseBlock4 is manipulated. In this case, 100 times TDR is higher than the original performance.

5.7.3 Performance validation on other dataset

Until now, we observed the performance increment on the LivDet-Iris-2020 dataset. Here, we select high-performing models (single and ensemble of models) based on their performance on the LivDet-Iris-2020 dataset and validate their performance on another dataset, i.e., the LivDet-Iris-2017 dataset. In the case of ensemble models, we consider two high-performing perturbed models and fuse their scores using the sum rule. We explore both directions of improved performance

(single and ensemble of models) for each architecture and compare their performance with the original models. Details of these models are given below:

1. Original Model: The model utilizes originally trained parameters without any perturbation of the parameters.

2. Perturbed Model: In the case of VGG19, we create a perturbed model by setting 80% of the low-magnitude weights of the seventh convolution layer to zero. For the ResNet101 model, a perturbed model is formed by setting 40% of low-magnitude weights of the first convolution layer to zero, while for the D-NetPAD, 90% of the low-magnitude weights of the last convolution layer of the first dense block are set to zero. The selection of these perturbed models are based on their performance on the LivDet-Iris-2020 dataset (5.4 (b)).

3. Ensemble Models: To create an ensemble model, we select two perturbed models and fuse their PA scores by the sum rule. In the case of VGG19, we fuse the perturbed model defined above and the model created by adding Gaussian noise with $\alpha = 0.3$ ($N(0, 0.3 * \sigma(W_{org}))$ to the entire network. In the case of ResNet101, we again use the perturbed model defined above, and the second model is created by adding Gaussian noise with $\alpha = 0.1$ to the entire network. Similarly, for D-NetPAD, we fuse the above specified perturbed model and the model formed by adding Gaussian noise with $\alpha = 0.1$ to all layers.

Table 5.3 provides the performance of these three models corresponding to the three architectures (VGG19, ResNet101, and D-NetPAD). **The performance of perturbed and ensemble models is better than the original model on both datasets.** The observation is consistent for all three architectures. The perturbed models show an average of 30.90% improvement on the LivDet-Iris-2017 and 3.86% on the LivDet-Iris-2020 dataset, whereas the ensemble models show an average of 47.59% improvement on the LivDet-Iris-2017 and 5.44% on the LivDet-Iris-2020 dataset. One major advantage of these perturbed models is that these models are created without any further training. Another advantage is that these high-performing perturbed models have reduced model size.

Table 5.3: The performance of VGG19, ResNet101, and D-NetPAD models in terms of True Detection Rate (%, higher the better) at 0.2% False Detection Rate on the LivDet-Iris-2017 and LivDet-Iris-2020 datasets. The performance is shown on original model (no parameter perturbations), perturbed model and an ensemble of model.

Datasets	**LivDet-Iris-2017**						**LivDet-Iris-2020**
Subsets	**Clarkson**	**Warsaw**		**Notre Dame**		**IIITD-WVU**	
Splits	**Test**	**K. Test**	**U. Test**	**K. Test**	**U. Test**	**Test**	
VGG19 Model							
Original	51.32	86.25	10.12	**100**	99.00	1.44	76.87
Perturbed	51.81	73.90	7.71	**100**	99.00	6.67	**78.55**
Ensemble	**67.64**	**88.14**	**21.71**	**100**	**99.11**	**8.49**	**82.93**
ResNet101 Model							
Original	**15.82**	89.93	91.67	**100**	99.44	50.47	84.11
Perturbed	14.50	**95.33**	93.18	**100**	**99.55**	55.55	86.39
Ensemble	14.71	95.18	**94.51**	**100**	99.33	**56.26**	**87.00**
D-NetPAD Model							
Original	60.04	76.68	35.76	**100**	**99.33**	32.01	90.22
Perturbed	**69.24**	**90.72**	**40.96**	**100**	97.33	**48.35**	**96.28**
Ensemble	68.89	89.53	36.94	**100**	96.88	41.68	94.76

5.8 Summary and Future Work

We analyze the robustness of three DNN architectures (VGG19, ResNet101, and D-NetPAD) under three types of parameter perturbations (Gaussian noise manipulation, weight zeroing, and weight scaling). We apply the perturbations in two settings: modifying the weights across all layers and modifying weights layer-by-layer. We found that DNNs are generally robust to a variant of weight zeroing, where low-magnitude weights are set to zero. From the layer-wise analysis, we observe that the DNNs are more stable to perturbations in later layers compared to the initial layers. Certain manipulations improve the performance over the original one. Based on these observations, we propose the use of an ensemble of models that consistently perform well on both LivDet-Iris-2017 and LivDet-Iris-2020 datasets. As future work, we will focus on finding the theoretical optimum direction for weight perturbations.

CHAPTER 6

RETRAINING OF DEEP NEURAL NETWORKS

6.1 Introduction

While a great deal of research in machine learning involves achieving higher performance at various classification/regression tasks, maintenance of that performance in a non-stationary environment [182] is a less explored area. The non-stationary environment (change in data capturing device, change in the deployment location, or change in a population group) degrades the performance of machine learning models. The performance degradation happens due to the dataset shift [182]. The dataset shift involves a shift in the input or output distributions or a shift in the relationship between input and output. The focus of this work is a shift in the input distribution. To maintain the performance of machine learning models under dataset shift, one needs to update the models with new incoming data. One solution is to fine-tune the model with new data; however, it results in catastrophic forgetting of the previously learned information [70, 71, 161]. Another solution is to retrain the model using entire data (old and new), but in the real-world scenario, old training data is generally unavailable due to security or privacy issues. So, the research problem arises how we should update the existing model that maintains the previous performance while improving the performance on new data, given that old training data is unavailable.

Mathematically, we define the problem as – let there is an expert model M trained on old training data TR_{old} and tested on old test data TS_{old}. It works satisfactorily on TS_{old}. Now, comes the new training data TR_{new} and new test data TS_{new}. So, given the existing trained model M and new training data TR_{new}, how we should retain the model M such that it maintains the performance on TS_{old} and improves the performance on TS_{new}. Here, we consider old training TR_{old} and test data TS_{old} belong to one domain, and new training TR_{new} and test data TS_{new} belong to another domain. We further assume following constraints to define the real-world retraining scenario:

1. **Sequential learning:** Data of different domains are supposed to learn in a sequence.

2. **Unavailability of old training data TR_{old}:** Training samples from the previous domain might not be available due to privacy or security concerns.

3. **Limited availability of new training data TR_{new}:** Training samples from new domain are generally small in number compared to the old training data. There might be an absence of training samples for certain classes.

4. **Memory constraints:** Information transfer from one domain to another is also restrained due to memory limitation.

5. **Architectural capacity constraints:** Finite capacity of an architecture limits its ability to learn new domain over time.

6. **Knowledge constraints:** Generally, a third party performs retraining of the deployed model. There could be a lack of expertise compared to the original architecture designer or developer.

In this work, we propose a dynamic weight-based fusion retraining strategy, where we train a new expert model with new incoming training data and make a final decision for a probe sample by a weighted sum of the predicted scores from the old and new trained models. We assign the weights individually for each probe sample using in-domain models at the run-time. The in-domain models provide information about the membership of the probe sample to the old and new training data. Our main contributions of the work are as follows:

1. We propose a novel retraining methodology which involves dynamic weight-based fusion of expert models. We allocate dynamic weights at the run-time for each probe sample.

2. We propose an in-domain model to assign dynamic weights to the scores of the expert models. The in-domain model works on the principle of outlier detection.

3. We perform experiments on three setups: LivDet-Iris-2017, LivDet-Iris-2020, and Split MNIST. These setups illustrate two levels of dataset shift. The first shift is between TR_{old} and TR_{new} and the second shift is between TR_{new} and TS_{new}.

The rest of the chapter is organized as follows: Section 6.2 discusses the related work on retraining strategies, Section 6.3 explains the proposed method. Section 6.4 describes the experimental setup and results. Finally, Section 6.5 concludes the chapter.

6.2 Related Work

Retraining is a process of including new samples in the old prediction pipeline. The objective is to improve the performance of the deployed model on new test samples TS_{new} and maintain the performance of old test samples TS_{old}. The closest terminology to the retraining paradigm is continual learning, which involves sequential learning of the number of tasks without forgetting knowledge obtained from the previous tasks. Continual learning considers three scenarios [118, 278]: Task-Incremental Learning (Task-IL), Domain-Incremental Learning (Domain-IL), and Class-Incremental Learning (Class-IL). Task-IL incrementally learns several independent tasks, explicitly knowing the task identity. Domain-IL learns tasks of the same output label space but differs input distribution. Here, task identity is not known. Class-IL incrementally learns new classes in each task without being given any information on the task identity. The retraining scenario is close to Domain-IL continual learning scenario.The difference is that Domain-IL assumes no shift in train and test distributions within a task, whereas the retraining scenario considers no assumption in this regard (includes both cases shift or no shift). Other synonyms of Domain-IL are lifelong learning [50], never-ending learning [155].

Other terms that could be confused with the retraining paradigm are multi-domain incremental learning [93], domain adaptation [289], and transfer learning [197, 292]. Multi-domain incremental learning [93] concerns with sequentially learning a task, say image classification, on multiple visual domains with possibly different label spaces, whereas we consider the same label spaces. Domain adaptation involves learning a new task of a different domain without retaining old domain knowledge. Transfer learning also does not retain the previous task knowledge either.

Two elementary retraining methodologies are: fine-tune the model using new training data (Fine-Tuned) or retrain the model using entire (old and new) training data (Full-Retrain). The

Fine-Tuned methodology results in catastrophic forgetting of the previous knowledge [70, 71, 161], whereas Full-Retrain is not a practical solution due to the unavailability of the old training data. Continual learning methodologies could also directly apply to the retraining paradigm, and in the literature, those are categorized as

1. **Regularization-based Approaches:** These approaches add regularization terms in the learning process to penalize the drastic change in parameters of the mapping function. The regularization helps to prevent catastrophic forgetting of previously learned tasks. Regularisation-based methods have a limited capacity to learn a large number of the tasks. Some seminal works based on regularization approaches are elastic weight consolidation (EWC) [141], online EWC [244], Kronecker factored EWC (KFAC) [229], Synaptic Intelligence (SI) [313], Memory Aware Synapses (MAS) [13], Learning without Forgetting (LwF) [161], Orthogonal Weight Modification (OWM) [312], and Natural Continual Learning (NCL) [136]).

2. **Dynamic Neural Networks/Parameter-Isolation Approaches:** These approaches begin with a simplified architecture and, when needed, augment the network incrementally with new components to attain satisfactory performance on subsequent tasks [42,81,156,228,233]. In a practical scenario, a finite capacity of models limits their ability to learn a large number of tasks over time.

3. **Replay-based/Memory/Rehearsal-based Approaches:** Replay-based approaches complement the existing expert models with memory to accommodate information about previously learned tasks. These approaches involve the usage of a subset of training samples from the previous tasks [14, 22, 44, 167, 204, 296] or the learning of generative or probabilistic models to simulate pseudo-samples from previously learned tasks [230, 252, 277]. However, as the number of tasks increases, it becomes difficult to maintain additional memory to store previous tasks information.

We propose a fusion-based methodology that learns a separate expert model using new training data and makes a final decision by a weighted sum of old and new prediction scores. The work

Table 6.1: Different methodologies of retraining along with the information about the knowledge needs to transfer to the next task and the special requirements for the training of the current task.

Approaches	**Methods**	**Knowledge transferred to next task**	**Special requirements for the training of new model**
Baselines	Fine-Tuned	None	None
	Full-Retrain	Old training data	Training includes old training data
Regularization-based Approaches	[13, 136, 141, 161, 229, 244, 312, 313]	None	Learning constraints
Dynamic Neural Networks Approaches	[42, 81, 156, 228, 233]	None	Increment architecture
Replay-based Approaches	[14, 22, 44, 167, 204, 296]	Subset of old training data	Training includes subset of old training data
	[230, 252, 277]	Generative model	Generation of synthetic old training data
Fusion-based Approaches	Proposed Method	In-domain Model	None

in [153] is close to our work, where they also learned separate expert models with incoming new training data and measured the marginal likelihood of the expert model using a density estimator, whereas we assign dynamic weights to expert models using in-domain model.

We also explicitly discuss the information required to transfer to the subsequent tasks (apart from the expert model) along with special requirements during the training of the current expert model in all retraining methodologies in Table 6.1. The Fine-Tuned method does not require anything from the previous task, whereas the Full-Retrain method requires the entire old training dataset. Regularization-based approaches do not require information from the previous tasks but apply additional constraints in the learning process. Dynamic Neural Networks approaches require a change in the architecture with subsequent tasks. Replay-based methods require additional memory to transfer a generative model or subset of old training data to the subsequent tasks. The proposed fusion-based approach does not require a change in the learning process or architecture. The approach also consumes less memory compared to the relay-based approaches. However, there is an increase in the inference time compared to the other methodologies.

6.3 Proposed Algorithm

In this work, we propose a dynamic weight-based fusion method to update existing expert model such that it maintains its performance on both TS_{old} and TS_{new} data. Figure 6.1 illustrates the proposed idea. Let's consider that we have old training data TR_{old} and its corresponding expert model. We also need to build an in-domain model on TR_{old}, which provides weight information. When a new training data TR_{new} come, we build two separate models (expert and in-domain models). During testing, we input a probe sample into all four models. Two expert models output prediction scores ($S1$ and $S2$), whereas in-domain models output weights ($W1$ and $W2$) assign to the prediction scores. We estimate the final prediction score as

$$S = W1 \times S1 + W2 \times S2. \tag{6.3.1}$$

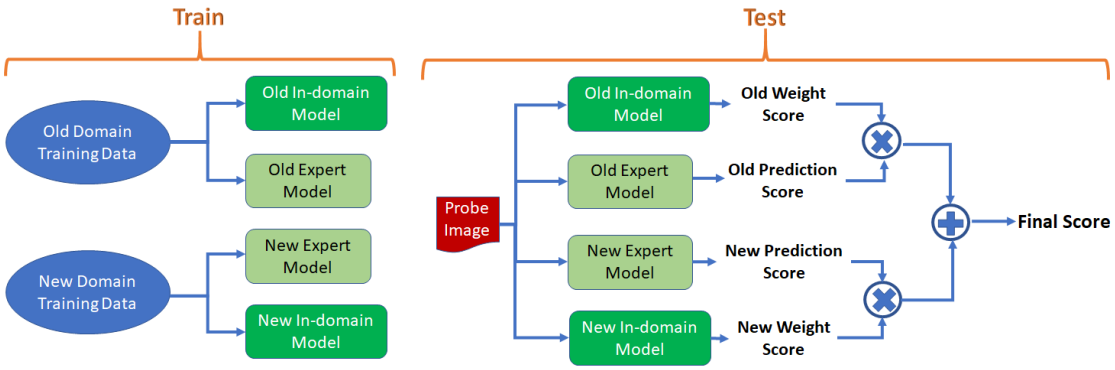


Figure 6.1: The overall idea of the dynamic weight-based fusion strategy for retraining. We train two models (expert and in-domain models) on incoming training data, and a final decision is made based on the weighted sum of their prediction scores. The expert model provides the prediction score, and the in-domain model assigns weight to the prediction score.

Our main contribution lies in the introduction of the in-domain model to estimate the dynamic weights. The in-domain model works on the principle of outlier detection, where training data of one expert model is considered as inliers and for each probe sample, we determine the degree of being an outlier with respect to the training data of the expert model. To accomplish this, an in-domain model contains two components: (i) a feature extractor that represents training distribution in feature space and (ii) a distance measure that provides the outlier score of a probe sample to the obtained training feature space. Details of these components are as follows

Feature Extractor (FE): The base architecture we use for feature extraction is Vision Transformer (ViT) [78]. The great success of transformers in natural language processing [116, 281] and computer vision [78] inspired us to use it for representing training data. The input to the ViT architecture is a sequence of flattened 2D image patches $x_p \in \mathbb{R}^{N\times(P^2.C)}$ and 1D positioning vector (providing position information of image patches), where N is the number of patches, P is the patch size, and C is the number of channels. We remove the MLP head used for classification from the ViT original architecture to make it a feature extractor. We use the "Base" version of the ViT, where there are 12 layers, 768-dimensional hidden latent vector, and 16×16 input patch. The total number of learnable parameters in the architecture is 86M. We train the ViT feature extractor using two losses: center and mean-shifted intra-class loss. The details of these losses are as follows:

1. **Center Loss:** The objective of the center loss is to extract features from the training samples such that feature embeddings are close to the center of the embeddings. The center of the training data embeddings is calculated as

$$c = \mathbb{E}_{x\in\chi_{train}}[\phi(x)] \tag{6.3.2}$$

   where, x is the input image, $\phi(x)$ is the features embedding from the ViT model and χ_{train} is the train set. We update the center position in every epoch. The center loss is then calculated as

$$\ell_{center}(x) = \|\phi(x) - c\|^2. \tag{6.3.3}$$

   The loss reduces the intra-train set variations among training samples and forms a closer feature representation of the samples. This helps in detecting outlier samples from other training set.

2. **Mean-Shifted Intra-Class Loss:** The objective of this loss is to form a cluster of samples belonging to the same class. To accomplish the objective, we first mean-shifted the embeddings of the training samples as

$$\theta(x) = \frac{\phi(x) - c}{\|\phi(x) - c\|^2} \tag{6.3.4}$$

where, $\phi(x)$ is the features embedding from the ViT model with x input sample and c is the center of the training samples in the ViT feature space. We then estimate contrastive loss over the two mean-shifted representations x' and x'' belong to the same class C_i as:

$$\ell_{msic}(x', x'')_{\{x', x''\} \in C_i} = \ell_{con}(\theta(x'), \theta(x'')) \\ = -\log \frac{\exp((\theta(x').\theta(x''))/\tau)}{\sum_{i=1}^{2N} \mathbb{E}[x_i \neq x'].\exp((\theta(x').\theta(x_i))/\tau)} \quad (6.3.5)$$

where, $\theta(.)$ is the mean-shifted representation, N is the batch size and τ is the temperature hyperparameter. Both losses together form cluster of samples belonging to the same classes around the center of the training samples. The class cluster formation helps in the detection of local outliers. By local outlier, we mean those samples whose inter-train set distance is lower but are outliers to their class distribution. Let consider Figure 6.2, where blue-colored data points belong to one training set, C is the center of the training set, and red-colored data point P is a probe sample. There are two classes (Class 1 and Class 2) of different densities. So, if we consider the global outlier measure, the probe sample would be an inlier to the blue-colored train set as its distance from Class 1 is less compared to the distance between data points of Class 1 and Class 2, but according to the local outlier measure, it is an outlier to the Class 1 as distance among data points of Class 1 are significantly lower the that of the probe sample and the data points of Class 1. Subsequently, the probe sample is also considered as an outlier to the blue-colored training set.

The total loss is the sum of center loss and mean-shifted intra-class loss as

$$\ell_{total}(x', x'') = \ell_{center}(x') + \ell_{center}(x'') + \ell_{msic}(x', x''). \quad (6.3.6)$$

Based on these losses, we train the feature extractor and used their features to represent the training data.

Distance Measure: After the representation of the training data and a probe sample, we estimate the distance of the probe sample with respect to the distribution of the training data using

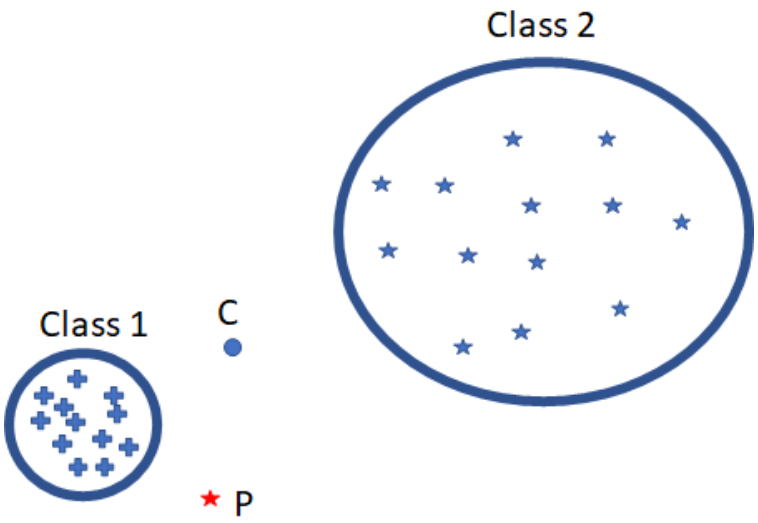


Figure 6.2: Illustration of a local outlier concept used in the mean-shifted intra-class loss. Blue-colored data points belong to one training set, C is the center of the training set, and red-colored data point P is a probe sample. There are two classes (Class 1 and Class 2) in the blue-colored train set. If we consider the global outlier concept, the red-colored probe sample would be inlier. However, if the local outlier concept is considered, the probe sample is an outlier to the Class 1 as well as to the blue-colored training set. The figure is better viewed in color.

Local Outlier Factor (LOF) [39]. The LOF is an unsupervised outlier detection algorithm that provides a score to each sample using local density deviation to its neighbors. It considers those samples as outliers whose density is substantially lower than their neighbors. If the LOF score has approximately a value of one, it suggests that the sample has a similar density as its neighbors, a value less than one indicates the sample has a higher local density than its neighbors, whereas a value greater than one presents the sample has a lower density than its neighbors. To assign weights to the individual expert models, we first inverse LOF scores and then perform SoftMax normalization to ensure weights lie in the range of [0-1] and weights sum to one.

6.4 Experimental Setup and Results

To evaluate the proposed methodology, we perform experiments on three setups: LivDet-Iris-2017, LivDet-Iris-2020, and Split MNIST. These setups consider dataset shift at two levels: 1) shift between training data of old TR_{old} and new TR_{new} domain, and 2) shift between training TR_{new} and test TS_{new} data within a domain (new domain). The LivDet-Iris-2017 setup represents the

scenario where dataset shift occurs between TR_{old} and TR_{new}, but no shift between TR_{new} and TS_{new} (except in one case, explained in Section 6.4.1). The LivDet-Iris-2020 setup illustrates the scenario where dataset shift occurs both in between TR_{old} and TR_{new} and between TR_{new} and TS_{new}. The Split MNIST setup depicts the condition where distribution of TR_{old} is disjoint of TR_{new}, however no shift between TR_{new} and TS_{new}. The LivDet-Iris-2017 and LivDet-Iris-2020 setups are in the application of detecting presentation attacks (PA) in iris biometric modality. We formulate the PA detection as a binary classification between bonafide and fake (print, cosmetic contacts, artificial eyes, and electronic display) iris images. The Split MNIST setup used to compare the proposed method with existing state-of-the-art continual learning methods.

6.4.1 LivDet-Iris-2017 Setup and Results

In this setup, we utilize two datasets: IARPA dataset [2] and LivDet-Iris-2017 dataset [304]. The IARPA dataset is a proprietary dataset collected under the IARPA Odin program [2]. We divide the dataset into two splits which in this setup considered as TR_{old} and TS_{old} data. The LivDet-Iris-2017 dataset [304] is a publicly available dataset for iris presentation attack detection. It consists of four subsets: Clarkson, Warsaw, Notre Dame, and IIITD-WVU. All these subsets consist of their corresponding train and test sets which in this setup considered as TR_{new} and TS_{new}, respectively. Details of these subsets are as follows:

1. Clarkson subset: It consists of print and cosmetic contacts PAs. The subset represents the cross-PA testing scenario, where five additional cosmetic contacts and prints of visible spectrum iris images captured using an iPhone 5 are included in the test set.

2. Warsaw subset: It consists of only print iris PA. The subset consists of two test sets: “known" and “unknown". The “unknown" test represents a cross-sensor scenario, where different sensors are used to capture images of the training and test sets.

3. Notre-Dame subset: It consists of only cosmetic contact iris PA. This subset also contains two test sets (“known" and “unknown"). The “unknown" test set represents the cross-PA

Table 6.2: Description of the old and new training/test sets in the LivDet-Iris-2017 setup along with the number of bonafide and fake iris images present in the datasets. The information about the sensors used to capture images is also provided. Each test set represents different testing scenarios. The Clarkson and Notre Dame test sets correspond to the cross-PA scenario, whereas the Warsaw data corresponds to the cross-sensor scenario. The IIITD-WVU represents a cross-dataset scenario. Here, "K. Test" means a known test set of the dataset, and "U. Test" means an unknown test set.

Domains	**Old Train and Test Domains (IARPA Dataset)**		**New Train and Test Domains (LivDet-Iris-2017 Dataset)**									
Datasets	**IARPA Split I**	**IARPA Split II**	**Clarkson (Cross-PA)**		**Warsaw (Cross-Sensor)**			**Notre Dame (Cross-PA)**			**IIITD-WVU (Cross-Dataset)**	
Train/Test	**Train**	**Test**	**Train**	**Test**	**Train**	**K. Test**	**U. Test**	**Train**	**K. Test**	**U. Test**	**Train**	**Test**
Bonafide	9,660	2,963	2,469	1,485	1,844	974	2,350	600	900	900	2,250	702
Print	2,634	-	1,346	908	2,669	2,016	2,160	-	-	-	3,000	2,806
Cosmetic Contacts	2,757	177	1,122	765	-	-	-	600	900	900	1,000	701
Artificial Eyes	554	175	-	-	-	-	-	-	-	-	-	-
Electronic Display	130	-	-	-	-	-	-	-	-	-	-	-
Sensor	Iris ID iCAM7000, IrisGuard AD100, IrisAccess LG4000	Iris ID iCAM7000	IrisAccess EOU2200	IrisAccess EOU2200	IrisGuard AD100		Aritech ARX-3M3C, Fujinon DV10X7.5A, DV10X7.5A-SA2 lens B+W 092 NIR filter	IrisGuard AD100, IrisAccess LG4000			Cogent CIS 202, VistaFA2E	IriShield MK2120U

scenario, where different cosmetic contacts introduce in the test set.

4. IIITD-WVU subset: It includes the PA images from both print and cosmetic contacts. The subset is a combination of data from IIITD and WVU collections. The subset corresponds to the cross-dataset scenario where training performs on the IIITD collection and testing on the WVU collection. The training set is captured in a controlled environment using two iris sensors: Cogent dual iris sensor (CIS 202) and VistaFA2E single iris sensor. The test set is captured using the IriShield MK2120U mobile iris sensor at two different locations: indoors (controlled illumination) and outdoors (varying environmental conditions).

The setup represents the scenario where there is a dataset shift between TR_{old} and TR_{new}, but no shift between TR_{new} and TS_{new} except in the case of the IITD-WVU dataset. In the IITD-WVU dataset, a shift also exists between TR_{new} and TS_{new}. Table 6.2 describes all old and new training/test sets along with types of PAs and images present in all training and test sets.

For evaluation, we compare it with following models: (i) **Old Expert Model:** trained only on TR_{old}, (ii) **New Expert Model:** trained only on TR_{new}, (iii) **Fine-Tuned:** trained on TR_{old} and then fine-tuned using TR_{new}, (iv) **Full-Retrain:** trained on both TR_{old} and TR_{new}, (v) **Fusion-Equal Weights:** fusion of scores from two expert models (Old and New Expert Models) with

equal weights, (vi) **Fusion-Pre-trained ViT Features-Dynamic Weights:** fusion of scores from two expert models with dynamic weights where in-domain model uses pre-trained ViT model for the feature representation, and (vii) **Fusion-Dynamic Weights (proposed method):** fusion of scores from two expert models with dynamic weights where proposed feature extractor (FE) model is used to represent the training data. To train FE model, we initialize weights with pre-trained model trained on the ImageNet-21k and JFT-300M datasets, the number of epochs used is 100, the batch size is 15, τ is 0.25, and the optimizer is stochastic gradient descent with a learning rate of 1e-5. For the implementation of LOF distance measure, we use default values provided in [39], the number of neighbors is 20, and the distance metric to estimate neighbors is euclidean distance. As an expert model, we use the model proposed in [249] for iris presentation attack detection. Table 6.3 presents the performance of all the models in terms of True Detection Rate (TDR (%)) at 0.2% False Detection Rate (FDR). TDR is the percentage of fake samples correctly detected, whereas FDR is a percentage of bonafide samples incorrectly detected as fake. The performance scores are reported individually on both test splits. The objective is to obtain high performance (higher TDR) on both test splits (TS_{old} and TS_{new}). The Full-Retrain model provides a benchmark for evaluating the performance of retraining methods.

The Old and New Expert models perform better in their respective test splits but fail on other test splits. The Fine-Tuned model performs better on TS_{new} but fails to retain knowledge about the old (poor performance on TS_{old}). The Full-Retrain model performs better on both test splits, but it is not a practical approach as old training data is generally unavailable. With respect to the fusion methodologies, we consider providing equal weights as our lower performance benchmark. We perform one ablation study where a pre-trained ViT model is used for the feature extraction. The proposed method outperforms both fusion-based methodologies, which validate that the proposed FE model better represents the training data and the weights are appropriately assigned to their respective models. We also visualize weight histograms of various test splits (Figure 6.3). The histograms are generated using weight values given to the New Expert Model for test data (both old and new). So, weight values toward '0' of the x-axis symbolize higher priority given to the

Table 6.3: The performance of all retraining methods in terms of True Detection Rate (%, higher the better) at 0.2% False Detection Rate on old (TS_{old}) and new (TS_{new}) test sets of the LivDet-Iris-2017 setup.

Test Domains	**Old** (TS_{old})	**New** (TS_{new})	**Old** (TS_{old})	**New** (TS_{new})		**Old** (TS_{old})	**New** (TS_{new})		**Old** (TS_{old})	**New** (TS_{new})
Datasets	**IARPA Split II**	**Clarkson (Cross-PA)**	**IARPA Split II**	**Warsaw (Cross-Sensor)**		**IARPA Split II**	**Notre-Dame (Cross-PA)**		**IARPA Test**	**IIITD-WVU (Cross-Dataset)**
	Test	Test	Test	K. Test	U. Test	Test	K. Test	U. Test	Test	Test
Old Expert Model	98.44	28.63	98.44	92.95	98.56	98.44	93.55	91.00	98.44	42.91
New Expert Model	25.54	92.05	0.31	100	100	29.90	100	66.55	0.31	29.30
Fine-Tuned	86.91	93.51	45.48	100	100	98.75	100	99.77	83.17	48.85
Full-Retrain	96.57	91.63	93.76	100	100	96.57	100	100	96.57	66.81
Fusion of Old and New Domain Expert Models										
Equal Weights	97.50	89.67	97.81	99.45	**100**	**99.37**	99.88	96.22	**98.44**	43.62
Pre-trained ViT Features-Dynamic Weights	98.13	72.80	91.27	**100**	99.38	99.37	**100**	80.44	88.16	29.27
Fine-tuned ViT Features-Dynamic Weights	**98.44**	**92.67**	**98.13**	**100**	**100**	**99.37**	**100**	**99.55**	98.13	**44.94**

Old Expert Model, whereas weight values towards '1' of the x-axis denote higher priority given to the New Expert Model. New test data of the IIIT-WVU subset produces weights around 0.5 as the distribution of IIIT-WVU subset test samples is independent of the training distribution of both expert models. So, assigning weights around 0.5 is an appropriate step. In all other cases, TS_{old} receives higher weights for the old expert model and TS_{new} receives higher weights for the new expert model. It is noteworthy that the proposed method outperforms even the Full-Retrain method except in the case of the IIIT-WVU test split. As specified earlier, the distribution of the IIIT-WVU test split is different from both training sets, which limits the performance of dynamic weight-based fusion in this particular case.

6.4.2 LivDet-Iris-2020 Setup and Results

In this setup we utilizes three datasets: IARPA dataset [2], Warsaw PostMortem v3 dataset [7] and LivDet-Iris-2020 dataset [61]. We divide the IARPA dataset into three splits and consider them as TR_{old} and two TR_{new} sets. Warsaw PostMortem v3 dataset is used as a TR_{new} and the LivDet-Iris-2020 dataset as TS_{new}. Description of these datasets are as follows:

1. Old training set (TR_{old}): One split of IARPA dataset collected using iCAM7000 iris sensor.

2. New training set (TR_{new}):

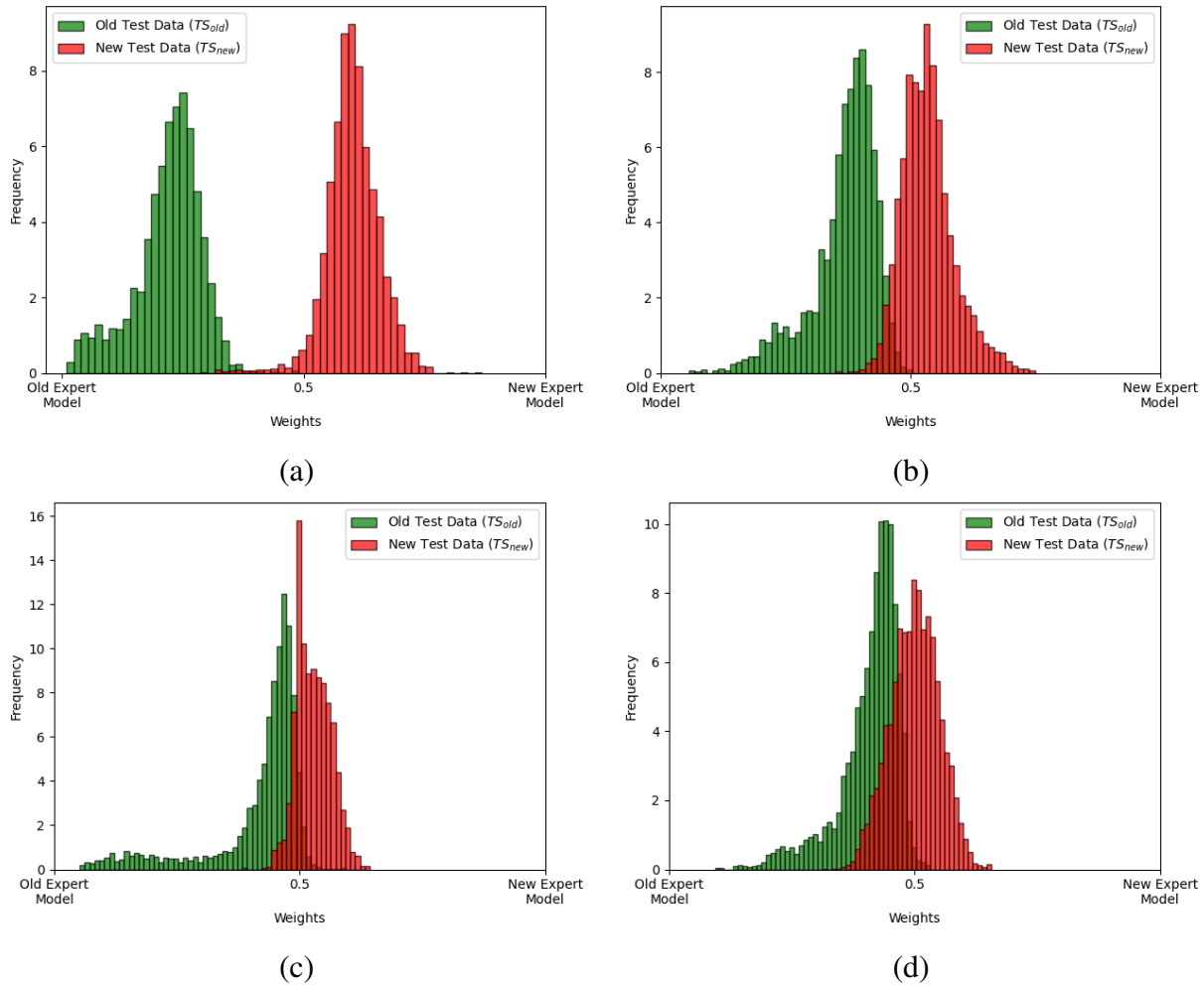


Figure 6.3: Histogram of weights dynamically allocated for all test samples (old and new) corresponds to (a) Clarkson,(b) Warsaw, (c) Notre-Dame, and (d) IIIT-WVU subsets of LivDet-Iris-2017 setup. In the case of Warsaw and Notre-Dame, 'Known' test splits are used for illustration. Weight values toward '0' of the x-axis symbolize higher priority given to the Old Expert Model, whereas weight values towards '1' of the x-axis denote higher priority given to the New Expert Model. New test data of the IIIT-WVU subset estimate weights around 0.5 as the distribution of the IIIT-WVU subset test set is independent of the training distribution of both expert models. The figure is better viewed in color.

a) IARPA split: It is the second split of the IARPA dataset. The type of fake images is the same as present in TR_{old}. The images are also collected from the same sensor. So, there is no additional information other than the more varieties of cosmetic contacts. The total number of images is also limited compared to TR_{old}.

b) Cross-sensor data: This is the third split of the IARPA dataset. The split represents the cross-sensor scenario, where images of this split are captured using LGIris and VistaEY2 iris sensors, whereas TR_{old} images are collected from the iCAM7000 sensor. The type of fake images is the same as TR_{old}. So, the data contains additional information about the sensors, and the number of images is higher than in TR_{old}.

c) Post-mortem data: It consists of images from the Warsaw PostMortem v3 dataset [7]. It represents the cross-PA scenario where a new type of fake images from cadaver eyes are present. It does not contain any bonafide images.

d) Combined data: It includes the data from all the above-stated data sets. So, it represents both cross-sensor and cross-PA scenarios. The total number of images is also higher than in TR_{old}.

3. New test set (TS_{old} and TS_{new}): The test dataset is the LivDet-Iris-2020 [61] competition data. It is independent of TR_{old} and TR_{new}.

The setup considers the scenario where dataset shift occurs both in TR_{old} and TR_{new}, and TR_{new} and TS_{new} except in the first case of IARPA split (no shift between TR_{old} and TR_{new}). Table 6.4 provides the number of bonafide and fake images used in these sets. Implementation details for the in-domain and expert models are the same as the previous experimental setup. Evaluation models are also the same: Old Expert Model, New Expert Model, Fine-Tuned, Full-Retrain, Fusion-Equal Weights, Fusion-Pre-trained ViT Features-Dynamic Weights, and Fusion-Dynamic Weights (proposed method). Table 6.5 presents the performance of all these models in terms of TDR (%) at 0.2% FDR on the LivDet-Iris-2020 test dataset.

Table 6.4: Description of the old and new train/test sets in the LivDet-Iris-2020 setup along with the number of bonafide and fake iris images present in the sets. The information about the sensors used to capture images is also provided.

Domains	**Old (TR_{old})**	**New (TR_{new})**				**Old and New (TS)**
Datasets	**IARPA Split I**	**IARPA Split II**	**Cross-sensor**	**Post-Mortem**	**Combined**	**LivDet-Iris 2020**
Train/Test	**Train**	**Train**	**Train**	**Train**	**Train**	**Test**
Bonafide	9,660	2,963	9,606	-	12,569	5,331
Print	2,634	-	-	-	-	1,049
Cosmetic Contacts	2,757	177	539	-	716	4,336
Artificial Eyes	554	175	383	-	558	541
Electronic Display	130	-	-	-	-	81
Cadaver Eyes	-	-	-	2,400	2,400	1,094
Sensor	Iris ID iCAM7000, IrisGuard AD100, IrisAccess LG4000	Iris ID iCAM7000	LGIris, VistaEY2	IriShield M2120U	Iris ID iCAM7000, LGIris, VistaEY2, IriShield M2120U	Iris ID iCAM7000, IrisGuard AD100, IrisAccess LG4000, IriTech IriShield

Table 6.5: The performance of all retraining methods in terms of True Detection Rate (%, higher the better) at 0.2% False Detection Rate on the LivDet-Iris-2020 test set.

Train Dataset	**IARPA Split I**	**IARPA Split II**	**Cross-Sensor**	**Post-Mortem**	**Combined**
Test Dataset	**LivDet-Iris 2020**				
Old and New Expert Models	61.86	58.25	75.55	0.94	85.56
Fine-Tuned	-	63.18	66.53	0	83.00
Full-Retrain	-	77.96	76.96	67.76	94.05
Fusion of Old and New Domain Expert Models					
Equal Weights	-	**72.42**	79.04	58.73	87.05
Pre-trained ViT Features-Dynamic Weights	-	69.91	79.03	58.73	89.38
Fine-tuned ViT Features-Dynamic Weights	-	69.27	**81.36**	**61.99**	**93.62**

The Old and New Expert models are not performing well on the test set as the distribution of the test set is different from TR_{old} and TR_{new} training sets. Similar is the case with the Fine-Tuned model. The Full-Retrain model outperforms the other models, but again it is not a practical approach due to the unavailability of the old training data. The proposed method outperforms both fusion-based methods (fusion with equal weights and Pre-trained ViT feature-based dynamic weight fusion). However, its performance is lower than the Full-Retrain method as the distribution of the test set does not match any of the training sets, and the proposed methodology depends on the training sets used by the expert models.

6.4.3 Split MNIST Setup and Results

We further perform experiments on the MNIST dataset for comparing the proposed retraining methodology with existing state-of-the-art continual learning strategies. The original dataset consists of 28×28 pixel grey-scale images of ten digits. We use standard train and test split, with 60,000 training images (~6,000 per digit) and 10,000 test images (~1,000 per digit). The main task is to classify even digit images from odd digit images. The main task is subdivided into five binary sub-tasks, where the first task is to classify '0' and '1' digits, the second task is to classify '2' and '3', and so on. The splitting of the dataset according to the five sub-tasks is referred as Split MNIST in the literature [118, 278]. The tasks are learned sequentially. The class labels are the same for all tasks, giving labels 0 to odd digit images and 1 to even digit images. Here, distributions of training data corresponding to different tasks are disjoint, but there is no shift between training and testing distribution within a task. Figure 6.4 depicts the experimental setup for the Split MNIST dataset.

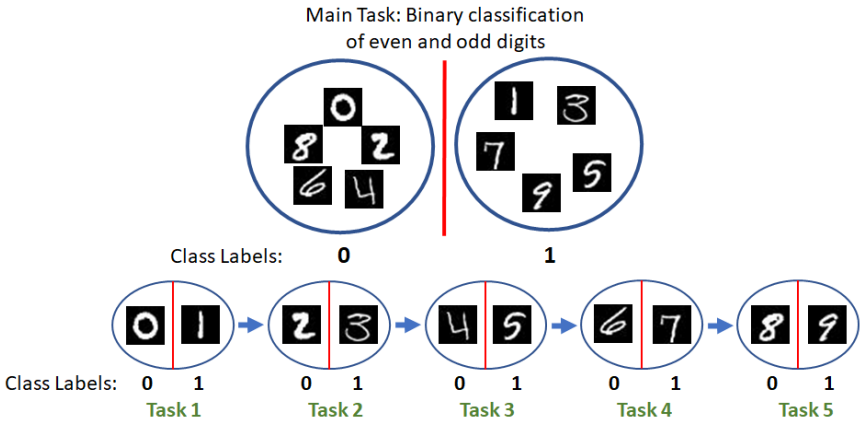


Figure 6.4: The experimental setup of the Split MNIST dataset for the retraining scenario. The main task is to classify old and even digit images. The task is divided into five sub-tasks, where the first task is to classify '0' and '1' digits, the second task is to classify '2' and '3', and so on. The class labels remain the same for all sub-tasks: 0 for odd digit images and 1 for even digit images.

For the expert model, we use multi-layer perceptron (MLP) architecture and learning parameters as used in the [118] for a fair comparison. The MLP architecture consists of two fully connected layers with 400 nodes each, followed by a softmax output layer. ReLU non-linearity is used in both fully connected layers. The loss function used is cross-entropy, the number of epochs is four and

batch size is 128, and the optimizer is stochastic gradient descent with a learning rate of 0.01. For each task, we build separate expert and in-domain models, which results in a total of ten models (five expert and five in-domain models). Implementation details of the in-domain model are the same as the previous experimental setup.

For comparative evaluation, we use the baseline models as provided in [118] which include Fine-tuned and Full-Retrain models. We also compare the proposed method with other continual learning methods: regularization-based (EWC [141], online EWC [244], SI [313], KFAC [229], MAS [13], LwF [161], OWM [312], NCL [136]) and replay-based (BiC [296], ER [44], GDumb [204], RM [22], DGR [252], GEM [167], RtF [277]). In the fusion-based approaches, we consider Fusion-Equal Weights, Fusion-Manual Weights, Fusion-Pre-trained ViT Features-Dynamic Weights, and Fusion-Dynamic Weights (proposed method). In the Fusion-Manual Weights, we manually assign one to the correct expert model and zero to other models. Fusion-Equal Weights is our lower performance limit, whereas Fusion-Manual Weights is the upper limit. As training data of all sub-tasks are disjoint, the manual weight assignment is a reasonable choice for the upper limit. All methods utilize the same experimental setup and expert models but few methods from regularization-based and replay-based approaches differ in hyperparameters (batch size, learning rate, and the number of epochs). Table 6.6 provides the results of all the methods in terms of the accuracy (%).

The proposed method outperforms Fine-Tuned baseline method and all regularization-based methods. Its performance is lower compared to three replay-based methods (DGR [252] and RtF [277] and GEM [167]). DGR [252] and RtF [277] are generative-based methods that involve separate training of a generative model along with an expert model. The generative model is then used to generate previous task samples and augment the training of the subsequent tasks. The process increases the training time of the subsequent tasks and makes the training of the expert model dependent on the generative model. However, the proposed method does not involve any generation of the samples, and the expert model is independent of additional models. GEM [167] method requires additional memory to store a subset of the previous task samples, which is a concern in terms of memory as well as privacy. The fusion-based methodology shows the maximum limit

Table 6.6: The average accuracy (%, higher the better) of the proposed retraining approach with different state-of-the-art continual learning approaches on the Split MNIST dataset. Methods with '+' superscript are reported from [118], 'o' from [136], '*' from [22] and '-' from [153]. All methods utilize the same experimental setup and expert models but differs in hyperparameters (batch size, learning rate, and the number of epochs). We use the same hyperparameters as used in [118]. Each value is an average of ten runs.

Approaches	**Method**	**Accuracy (%)**
Baselines	Fine-Tuned^{+}	63.20 ± 0.35
	Full-Retrain^{+}	98.59 ± 0.15
Regularization-based Approaches	EWC [141]$^{+}$	58.85 ± 2.59
	Online EWC [244]$^{+}$	57.33 ± 1.44
	SI [313]$^{+}$	64.76 ± 3.09
	KFAC [229]	67.86 ± 1.33
	MAS [13]$^{+}$	68.57 ± 6.85
	LwF [161]$^{+}$	71.02 ± 1.26
	OWM [312]^{o}	87.46 ± 0.74
	NCL [136]^{o}	91.48 ± 0.64
Replay-based Approaches	BiC [296]*	77.75 ± 1.27
	ER [44]$^{-}$	85.69
	GDumb [204]*	88.51 ± 0.52
	RM [22]*	92.65 ± 0.33
	DGR [252]$^{+}$	95.74 ± 0.23
	GEM [167]$^{+}$	96.16 ± 0.35
	RtF [277]$^{+}$	97.31 ± 0.11
Fusion-based Approaches	Equal Weights (Lower Limit)	84.20 ± 0.08
	Manual Weights (Upper Limit)	98.66 ± 0.008
	CN-DPM [153]$^{-}$	93.23
	Pre-trained ViT Features-Dynamic Weights	81.34 ± 0.005
	Fine-tuned ViT Features-Dynamic Weights (Proposed Method)	94.32 ± 0.01

as 98.66% (performance of Fusion-Manual Weights), which outperforms all methods even the Full-Retrain method. We also experiment with another distance measure (Mahalanobis distance), and it results in an accuracy of 97.03 ± 0.0001%, which is as par as the replay-based methods without any generation or storage of additional training data. In this setup, Mahalanobis distance performs the best as disjoint training distributions are effectively characterize by the mean and variance, whereas LOF outperforms in other setups.

To understand the importance of training the ViT-based feature extractor with the proposed losses, we visualize the features extracted from the pre-trained ViT model (Figure 6.5) and our trained FE model (Figure 6.6). The visualization involves training embeddings corresponding to all five sub-tasks (shown in different colors). The embeddings are reduced to three dimensions using t-SNE [279]. The pre-trained model features show significant overlap among different task embeddings compared to our trained FE model. The performance and visualization both validate the use of loss functions involved in the training of ViT-based feature extractor.

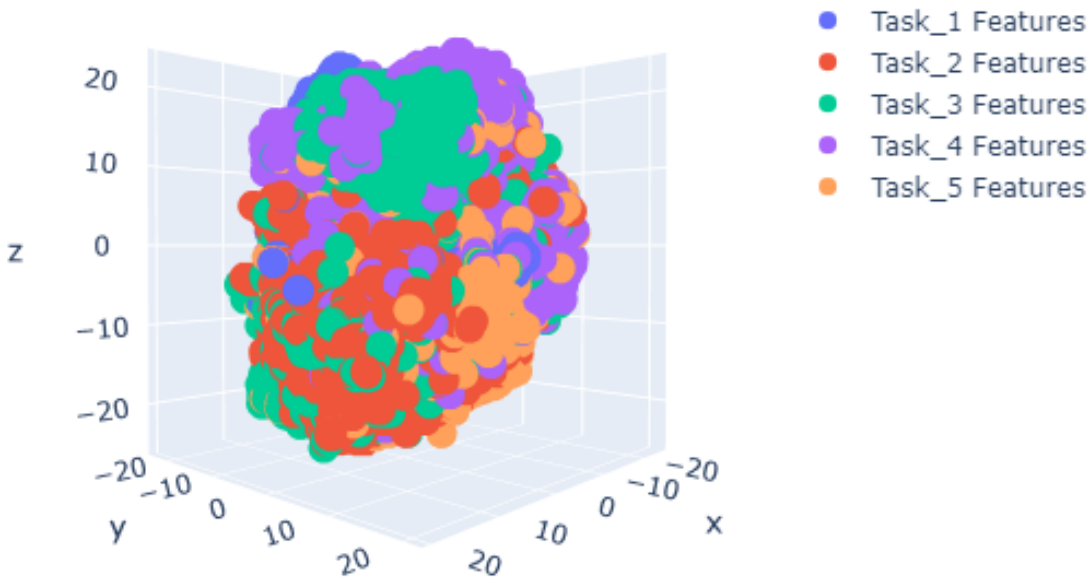


Figure 6.5: 3-D t-sne plot showing pre-trained ViT embeddings correspond to five sub-tasks of the Split MNIST dataset. The training samples of different classes are overlapping in the feature space. The figure is better viewed in color.

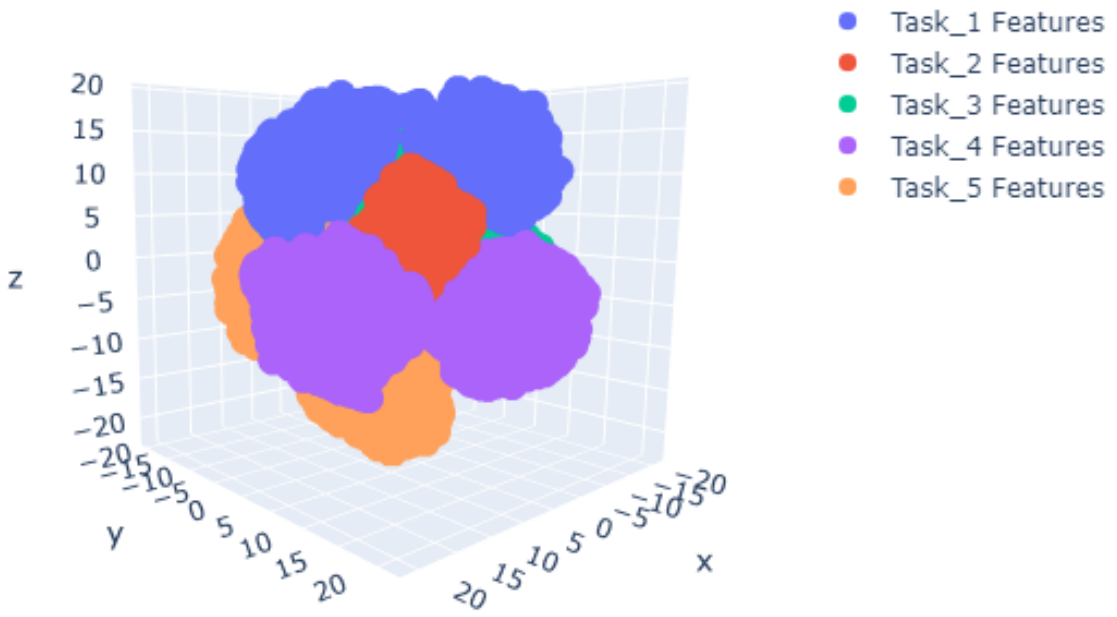


Figure 6.6: 3-D t-sne plot showing fine-tuned ViT embeddings correspond to five sub-tasks of the Split MNIST dataset. There is a formation of clusters of training samples belonging to the same class in the feature space. The figure is better viewed in color.

6.4.4 Findings

The main findings from the three experimental setups are as follows:

1. When TS_{new} has a similar distribution as TR_{old} or TR_{new} data, our proposed approach outperforms other approaches, even the Full-Retrain method as shown in the LivDet-Iris-2017 setup (Table 6.3).

2. When TS_{new} distribution is independent of TR_{old} and TR_{new} training data, the proposed approach outperforms other approaches, but not the Full-Retrain method as shown in the IIIT-WVU subset of LivDet-Iris-2017 setup (Table 6.3) and LivDet-Iris-2020 setup (Table 6.5).

3. In the case of disjoint training distribution between TR_{new} and TR_{old}, the proposed approach outperforms baselines, regularization-based, and other fusion-based approaches. Its performance is lower than some replay-based methods. However, the performance improves by Mahalanobis distance as an outlier distance measure in this particular scenario.

4. The proposed in-domain model for dynamic weights allocation is appropriately assigning weights to their respective expert models, as exhibited by its higher performance compared to the Fusion-Equal Weights method in Tables 6.3, 6.5 and 6.6. Weight histograms in Figure 6.3 also validate the accurate allocation of the weights.

5. The proposed FE model better represents the training data as shown by comparing its performance with pre-trained ViT model in Tables 6.3, 6.5 and 6.6. Figure 6.6 also visually validate the finding.

6.5 Summary and Future Work

We propose a dynamic weight-based fusion methodology to update the existing expert models such that it maintains the performance on old test data alongside improves the performance on new test data. The method asserts a new expert model on new training data and makes the final decision by the weighted sum of the prediction scores from all expert models. Evaluation of the proposed approach in three setups depicting two levels of dataset shift validates its effectiveness. In this work, by dataset shift, we mean shift in input distribution. As the method does not manipulate the existing expert models, it motivates the reuse of existing expert models without any manipulation in the training process or the architecture of the expert models. It also requires less memory as compared to the replay-based methods. However, there is an increase in the inference time as a probe image input to another model for estimating weights. Regarding scalability, there involves the addition of two models with every new incoming task, and hence the number of models linearly increases with an increase in tasks. The number of models could reduce by applying pre-condition (performance difference or data distribution difference) before building additional models. In future work, we will define the pre-condition for improving the scalability of the proposed method.

CHAPTER 7

IRIS MORPHING ATTACK: CREATION AND DETECTION

Parts of this chapter appeared in the following publication:

R. Sharma and A. Ross, "Image-Level Iris Morph Attack," IEEE International Conference on Image Processing (ICIP), 2021.

7.1 Introduction

In this chapter, we investigate the problem of morph attacks in the context of iris biometrics. We employ a landmark-based iris morphing scheme at the image level which generates morphed iris images. The potential of generated morphed iris images is then analyzed over the three iris recognition systems.

The main contributions of the work are as follows:

1. We propose a landmark-based method to perform iris morphing at the image-level.
2. We evaluate vulnerability of three iris recognition techniques (USITv3.0 [227], VeriEye[1], and CNN-Pairwise [206]) to morphed iris images using two publicly available datasets (IITD [150] and WVU multi-modal[2]). The attack success rate is over 90% at 0.01% false match rate.
3. We explore the similarity required between the component images to create a successful morphed iris image.
4. We provide preliminary results on the detection of image-level morphed iris images.

The rest of the chapter is organized as: Section 7.2 discusses the various morphing techniques in the context of biometrics, Section 7.3 provides the methodology used to create image-level

[1]https://www.neurotechnology.com/verieye.html
[2]https://biic.wvu.edu/data-sets/multimodal-dataset

morphed iris images, Section 7.4 describes the datasets, Section 7.5 provides the experimental setup and results on both the datasets, and Section 7.6 concludes the chapter.

7.2 Related Work

Morphing can be performed at the image-level or feature-level. Morphing at the image-level is relatively simple as it does not require knowledge of the internal working of a biometric system, whereas morphing at the feature-level requires knowledge of the feature extraction module. The image-level morphed samples can directly be presented to the sensors or digitally uploaded to the biometric system.

Commonly used morphing techniques at the image-level are landmark-based [26, 86, 171, 241]. The landmark-based techniques first detect corresponding landmark points in the two images then warp the images based on the detected landmarks, and finally blend the warped images. Shechtman *et. al* [251] posed the morphing as an optimization problem to achieve bidirectional similarity of each morphed image with its neighboring frames within the morph sequences as well as the input images. Recently, morphing techniques based on generative adversarial networks have been proposed [9, 60, 314]. However, their attack success rate at this time is still lower than the landmark-based techniques. Morphing techniques have also been proposed at the feature-level, e.g., minutiae-based [85], iris-codes [225]. Detailed surveys on morphing techniques in the context of morph attacks can be found in [242, 284]. Further, frameworks for evaluating the vulnerability of biometric systems to morphed samples are presented in [95, 240].

In the case of the iris modality, Rathgeb and Busch [225] proposed morphing at the *feature-level*, where iris-codes are morphed using stability-based bit substitution. Erdongan *et. al* [82] proposed morphing on the normalized iris images.[3] They created composite normalized iris images based on the selection of pixels from the two images considering their intensity and phase profiles. We propose a morphing scheme for generating morphed iris images using two *unnormalized* iris images.

[3]Here, normalization refers to the unwrapping of the iris wherein it is mapped from Cartesian coordinates to Pseudo-polar coordinates resulting in a fixed-size rectangular entity

7.3 Algorithmic Details

We generate a synthetic iris image (morphed image) from samples of two different identities such that the morphed image matches with both of its component identities. We utilize the landmark-based method to create morphed images. There are generally three steps to such an approach [242]: correspondence, warping, and blending. In the correspondence step, a set of correlated landmarks points from both images are detected. In the warping step, two images are non-linearly deformed to make them geometrically aligned with respect to the detected landmarks. Finally, the warped images are blended by linearly combining pixel values from both images at each location using a scalar value (blending factor). The scalar value controls the degree of contribution of each source image to the morphed image.

1. Correspondence: To establish the correspondence between two iris images, we first obtain iris segmentation parameters – iris center, iris radius, pupil center, and pupil radius. Using the segmentation parameters, we estimate equally spaced landmarks on both the inner and outer iris boundaries. The landmark points are 10 degrees apart with respect to the iris center resulting in 72 landmarks (36 on the inner iris boundary + 36 on the outer iris boundary). We select these 72 landmarks to minimize iris feature distortion during warping. A lower number of landmarks distorts the iris pattern during warping, and a higher number increases computational complexity. We also include four extreme corner points of an image (top left, top right, bottom left, and bottom right) in the landmarks set, creating a total of 76 landmark points. The corner points are required to align the iris regions of both images.

2. Warping: Given the landmark points, we compute the Delaunay triangulation using the convex hull method [23]. We average the corresponding triangle coordinates and compute their affine transformation matrix, T, as follows:

$$T_{3\times3} = A_{3\times3} X_{3\times3}^{-1}. \tag{7.3.1}$$

Here, A is the averaged triangle coordinates arranged column-wise, and X is one of the corresponding triangles coordinates. Using the transformation matrics, triangles from both images are warped

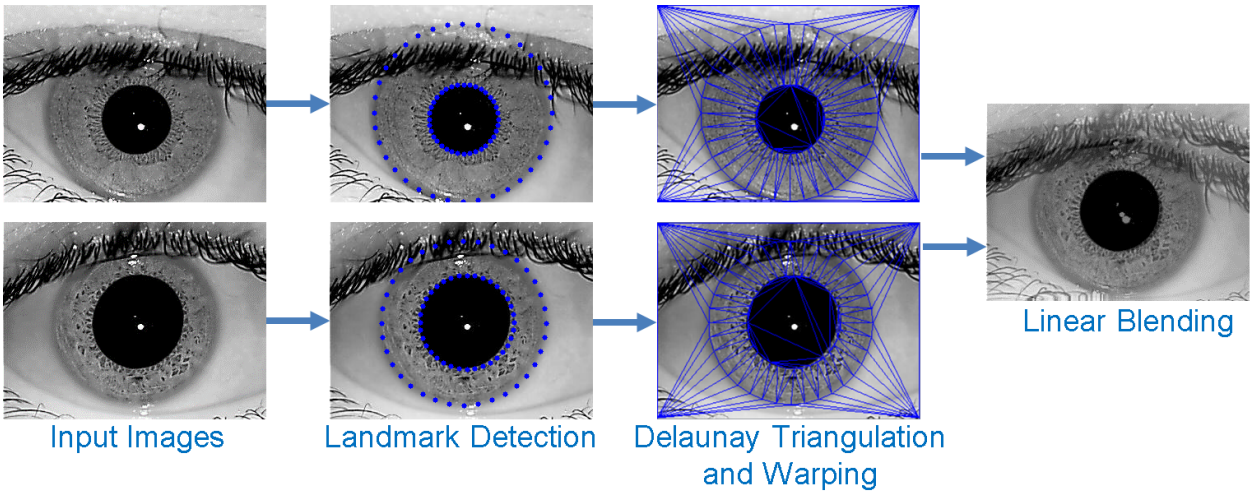


Figure 7.1: (a) Three categories of techniques applied to detect iris presentation attacks. (b) Illustration of the iris morphing at the image-level. It consists of registration of landmark points on both the images, alignment of images, and then blending into a single image.

to the averaged triangle coordinates. We further interpolate the missing values using bilinear interpolation.

3. Blending: Finally, we blend the pixels within the warped triangles using linear blending at each location (i, j) as follows:

$$M(i, j) = \alpha X_w(i, j) + (1 - \alpha) Y_w(i, j). \tag{7.3.2}$$

Here, M is the morphed triangle, X_w and Y_w are two corresponding warped triangles, and α is the blending factor. The blending factor is set to 0.5 to get an equal contribution of identity information from both the images. Figure 7.1 shows a pictorial representation of these steps.

7.4 Datasets

To demonstrate the vulnerability of iris recognition techniques to morph attacks, we conduct experiments on the following two publicly available iris datasets:

1. IITD Iris Dataset [150]: The IITD iris dataset consists of 2,240 iris images from 224 subjects. There are ten iris images per subject (5 left and 5 right). The images are acquired using JIRIS, JPC1000, and digital CMOS sensors. The subjects in the dataset are in the age range of

14-55 years. There are 176 males and 48 females in the dataset. These images have a resolution of 320×240 pixels.

2. WVU Multi-modal Release 1 Dataset: The WVU multi-modal dataset consists of the iris, face, fingerprint, voice, palmprint, and hand-geometry modalities. We only use the iris modality, which contains 3,099 iris images from 244 subjects. There is an average of 12 images per subject. The resolution of these images is 640×480 pixels.

7.5 Evaluation and Results

To evaluate the impact of image-level morphed images on iris recognition, we first compute the baseline performance of three iris recognition techniques (USITv3.0 [227], VeriEye, and CNN-Pairwise [206]) on the IITD and WVU multi-modal datasets. The two datasets are used to create morphed iris images. Subsequently, we assess the susceptibility of iris recognition techniques to the generated morphed iris images. Further, we also analyze the textural similarity of component images required to create a successful morphed iris image. Finally, we provide preliminary results on the detection of morphed iris images.

7.5.1 Baseline Recognition Performance

We utilize three iris recognition techniques to assess their vulnerability to morphed iris images. The first is a best performing technique within the open-source iris recognition software toolkit, University of Salzburg Iris Toolkit (USIT v3.0) [227]. It extracts iris-code using quadratic spline wavelet (QSW) [151] and uses hamming distance to measure the dissimilarity between the iris-codes. The second is a commercially available off-the-shelf technique called VeriEye. The third is a deep learning-based CNN-Pairwise [206] technique. We utilizes DenseNet121 [122] as the base architecture. The network inputs two cropped iris images and formats them as multiple channels and then outputs a similarity score between 0 (impostor) and 1 (genuine). The iris images are manually segmented for the USITv3.0 and CNN-Pairwise techniques, whereas VeriEye uses its own iris segmentation module. The segmentation failures occured in VeriEye are manually corrected.

Table 7.1: Performance of three iris recognition techniques in terms of TMR (%) at 0.01%, 0.1%, and 1% FMRs, on the IITD and WVU datasets. The USITv3.0 is an open-source iris recognition toolkit, VeriEye is a commercial iris recognition SDK, and CNN-Pairwise is a deep learning-based technique.

Algorithms	IITD Dataset (TMR(%))			WVU Dataset (TMR(%))		
	FMR 0.01%	FMR 0.1%	FMR 1%	FMR 0.01%	FMR 0.1%	FMR 1%
USITv3.0	99.33	99.55	99.72	94.73	96.40	97.62
VeriEye	**99.77**	**99.77**	**99.77**	**98.54**	**98.78**	**99.02**
CNN-Pairwise	98.16	98.72	99.38	85.70	90.54	93.69

The recognition performance of these three techniques is evaluated on both datasets. As the CNN-Pairwise method is a deep learning-based, training is performed using 60% of the subjects, and the rest is used for testing (subject-disjoint strategy). Table 7.1 provides the True Match Rate (TMR) at 0.01%, 0.1%, and 1% False Match Rate (FMR). The VeriEye algorithm performs the best followed by the USITv3.0 algorithm. CNN-Pairwise shows relatively lower performance (presumably) due to insufficient training data, whereas the other two techniques do not require training.

7.5.2 Morph Attack Setup and Results

We utilize both the datasets to create image-level morphed iris images. In the IITD dataset, there are 224 left eye classes and 224 right eye classes. We randomly select one image per class for generating the morphs, which should result in 49,952 ($^{224}C_2 + ^{224}C_2$) morphed images. However, landmarks could not be detected in some images with partial irides, so a total of 49,816 morphs were created. In the WVU dataset, there are 237 left eye classes and 233 right eye classes, which should result in a total of 54,994 ($^{237}C_2 + ^{233}C_2$) morphs. However, due to landmark detection problems in partial irides, not all pairs could be considered, resulting in a total of 50,573 morphs. Figure 7.2 presents few samples of morphed images generated from both datasets along with their component images. We input the morphed iris images to three iris recognition techniques (morph attack) and measure their vulnerability in terms of Mated Morph Presentation Match Rate (MMPMR) [240]. MMPMR is the ratio of successful morph attacks to total morph attacks. The

Table 7.2: Vulnerability assessment of three iris recognition techniques to iris morph attacks in terms of MMPMR (%) at different thresholds corresponding to 0.01%, 0.1%, and 1% FMRs on the IITD and WVU datasets.

Algorithms	IITD (MMPMR(%))			WVU (MMPMR(%))		
	FMR 0.01%	FMR 0.1%	FMR 1%	FMR 0.01%	FMR 0.1%	FMR 1%
USITv3.0	93.8	95.64	96.91	85.96	93.82	**97.07**
VeriEye	**95.77**	**97.07**	**97.85**	**90.22**	**94.48**	97.02
CNN-Pairwise	17.64	24.76	25.32	47.70	54.23	56.76

morph attack succeeds when the morphed image matches with all of its component subjects at a specified threshold. Table 7.2 provides the performance of morph attacks in terms of MMPMR at different thresholds corresponding to 0.01%, 0.1%, and 1% FMRs.

The VeriEye and USITv3.0 techniques are more susceptible to morph attacks (> 90% MMPMR). CNN-Pairwise shows a relatively lower morph attack success rate as it has been trained on a relatively small amount of data, and a slight perturbation in the images (due to morphing) results in non-matches. To evaluate further, we plot a histogram of the genuine, impostor, and morph match scores on both the datasets (top row of Figure 7.3). The distribution of morph match scores leans towards the genuine distribution, and a majority of the morph match scores are labeled as genuine when considering the threshold at 0.01% FMR. We also visualize the morph match scores using scatter plots (bottom row of Figure 7.3) with component identities along X and Y-axes. Most of the match scores are in a quadrant that corresponds to successful matches with both component subjects (top right corner). This shows how well the morphed images match with their component identities and substantiates the vulnerability of iris recognition techniques to morph attacks.

7.5.3 Analysis of Textural Similarity

Next, we analyze the similarity between the component iris images used to create morphed iris images. To calculate the similarity, we utilize the Root Mean Square Error (RMSE) and Structural Similarity Index Measure (SSIM) [290] measures. RMSE calculates the pixel-wise difference between the two images (higher the value, lower the similarity), while SSIM estimates the structural

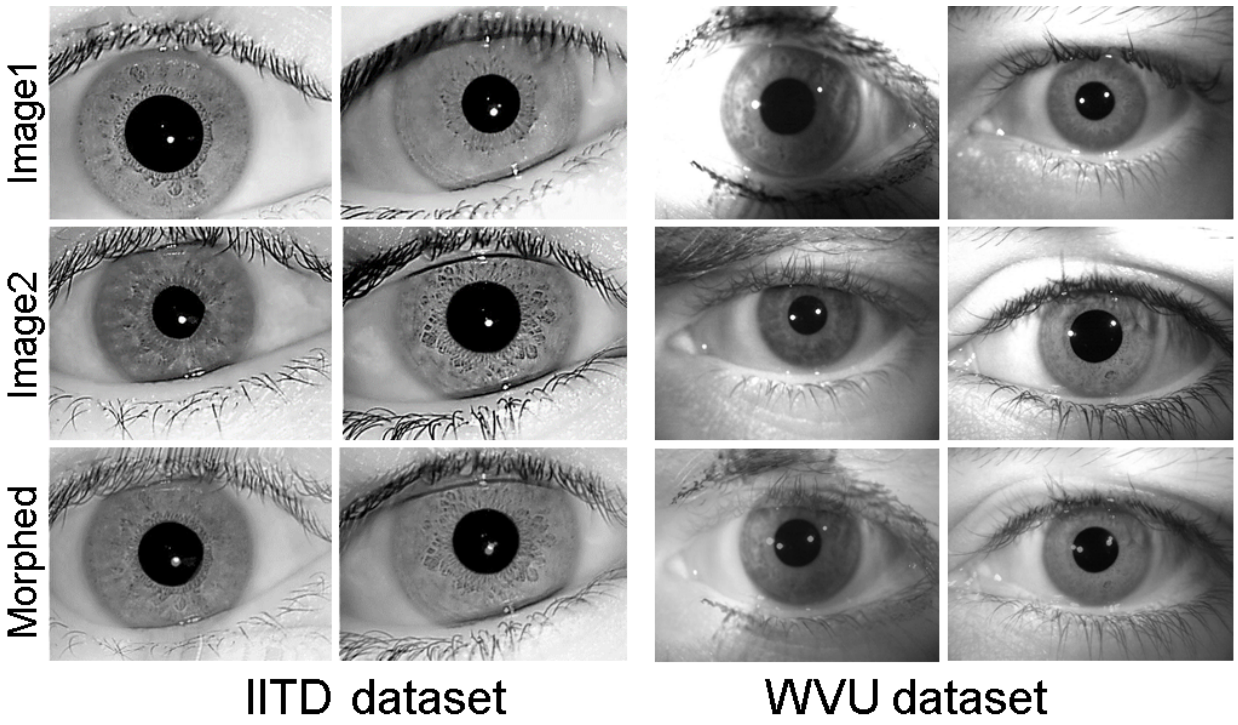
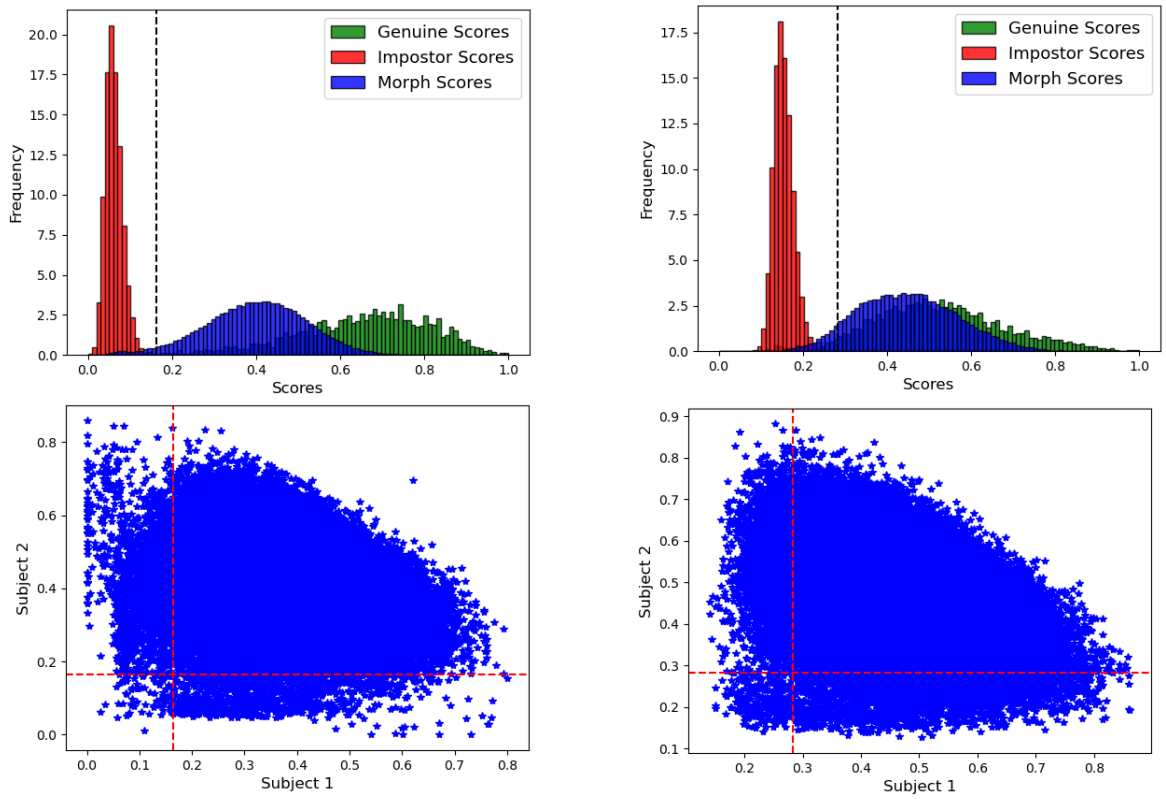


Figure 7.2: Samples of morphed images generated from the IITD and WVU datasets.

similarity between the two images (higher the value, higher the similarity). Figure 7.4 presents the distribution of similarity scores as calculated by RMSE and SSIM on both the datasets. Distribution in green corresponds to successful morphs, whereas distribution in red corresponds to unsuccessful morphs. Match scores and threshold are according to the USITv3.0 iris recognition technique.

The mean SSIM scores corresponding to successful and unsuccessful morphs are 0.49 and 0.45 (the mean difference is 0.029), respectively, on the IITD dataset. The mean difference increases to 0.032 when using RSME scores. Though distributions of successful and unsuccessful morphs significantly overlap, we can still conclude that there is a high chance of generating a successful morph if SSIM between the component images is more than 0.31 (or less than 0.20 in the case of RSME). A similar conclusion can be made on the WVU dataset that there is a high chance of obtaining a successful morph when SSIM between the component images is more than 0.49 (or less than 0.22 in the case of RSME).



IITD dataset WVU dataset

Figure 7.3: Top: Match score distribution of genuine (green), imposter (red), and morph attacks (blue) on the IITD and WVU datasets using the USITv3.0 iris recognition technique. Bottom: Scatter plots of match scores, where morphed images match with their component identities. The dotted line represents the threshold at 0.01% FMR.

7.5.4 Morph Attack Detection

The next natural step is to address morph attacks by detecting morphed images prior to inputting them into an iris recognition system. We present preliminary results on the detection of morphed iris images. Firstly, we perform detection using a pre-trained presentation attack detector [249][4] at a pre-defined threshold (corresponds to 0.2% False Detection Rate (FDR)). It results in 9.06% True Detection Rate (TDR) on morphed images from the IITD dataset and 16.82% on morphed images from the WVU dataset. Next, we fine-tune the detector with morphed iris images (60% from each dataset used for training and the rest for testing). It attains 86.83% TDR on the IITD

[4]An iris presentation attack detector trained to detect artifacts such as printed iris, cosmetic contacts, and artificial eyes.

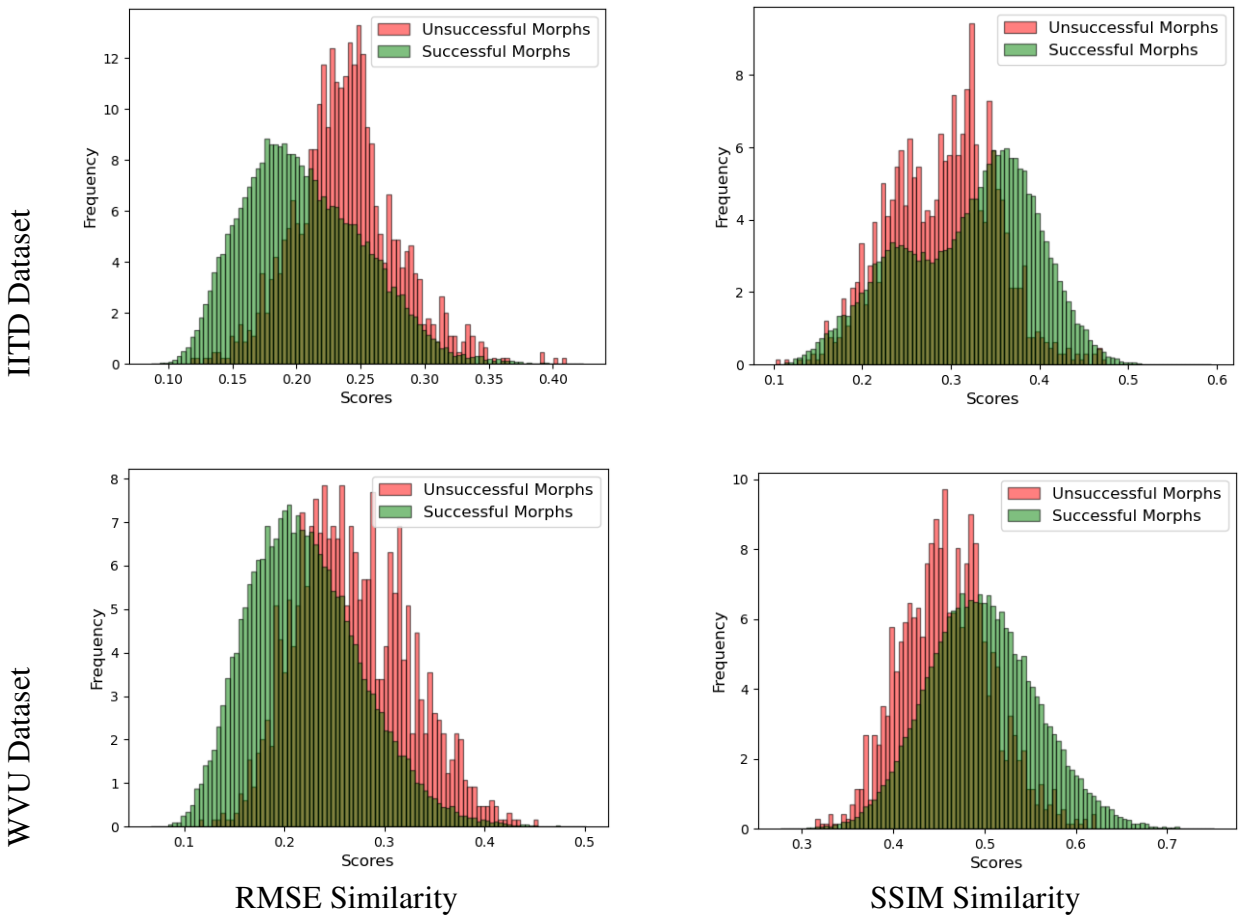


Figure 7.4: Distributions of similarity scores between the component images corresponding to successful (green) and unsuccessful (red) morphs using the RMSE (higher the value, lower the similarity) and SSIM (higher the value, higher the similarity) measures on the IITD and WVU datasets.

morphed images and 99.55% TDR on the WVU morphed images at 0.2% FDR. In the proposed morphing technique, we did not perform post-processing on the morphed images due to which some artifacts are present outside the iris region. We hypothesize that these artifacts aid in the detection of morphed iris images.

7.6 Summary

We successfully generate image-level morphed iris images, which can be used to denote two identities thereby posing a security concern. The morphed images show a high morph attack success rate (> 90%) on three high-performing iris recognition methods (USITv3.0, VeriEye, and

CNN-Pairwise) when assessed on two datasets (IITD and WVU multi-modal). We also explore the textural similarity required between the component samples to create a successful morphed image. Finally, we present preliminary results on the detection of morphed iris images.

CHAPTER 8

MATCHING IRIS IMAGES WITH FACE IMAGES

8.1 Introduction

Biometric recognition involves matching two biometric samples primarily from the same modality, such as the face, iris, fingerprint, or voice to identify an individual. However, cross-modal recognition implies matching of two biometric samples from different modalities. These two biometric samples are refered as an enrolled and a probe sample, where the enrolled sample exists in a corresponding legacy database. Cross-modal recognition helps in the case of unavailability of legacy datasets or where we need to analyze the relationship between two modalities. It also helps in boosting the recognition confidence even if the legacy dataset is available. Various efforts made in this direction are [129, 163, 168, 185, 234]. In this work, we focus on matching iris images captured in the near-infrared spectrum against face images captured in the visible (Figure 8.1). It also helps in recognizing humans when face recognition is not reliable, such as the presence of occlusions on the face (face mask).

Two main challenges arise from matching a face image to an iris image: (i) a large domain gap and (ii) imbalanced training data. The domain gap results from the following factors:

1. *Cross-Modality*: There occurs matching of iris modality images to face modality images.

2. *Cross-Sensor*: Different sensors are used to capture the face and iris images. Sensors add various noises to the images, for instance, fixed pattern noise, pixel response non-uniformity (PRNU), random noise, etc.

3. *Cross-Spectrum*: Generally, face image captures in the visible spectrum (VIS), whereas iris image captures in the near-infrared (NIR) spectrum. When considering the iris region only, NIR illumination (700-900nm) captures the stromal features (fibrovascular layer) of the

iris, whereas VIS illumination (400-700nm) captures melanin pigment and a meshwork of ligament features.

4. *Cross-Resolution*: Iris or ocular regions cropped from face images are of very low resolution as compared to iris images. For example, in the BioCop-2008 dataset, the ocular or iris regions cropped from face images are of 0.06 or 0.006 megapixels, whereas it is of 0.3 or 0.03 megapixels on iris images.

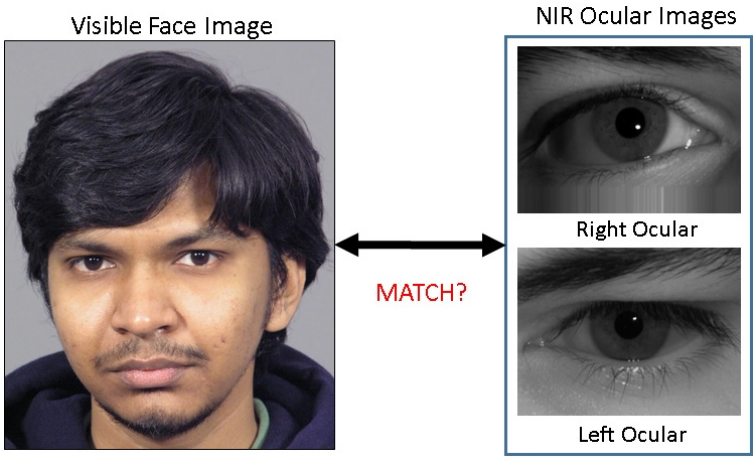


Figure 8.1: The objective is to match a visible spectrum face image with the NIR spectrum iris image, or vice versa.

Previous literature focuses on one or two of these factors. To the best of our knowledge, only one paper [129] dealt with all four challenges. The authors propose various handcrafted features (Local Binary Patterns (LBP), Normalized Gradient Correlation (NGC), and Joint Dictionary-based Sparse Representation (JDSR)) and the score-level fusion of those features. They achieve a 23% Equal Error Rate (EER), which shows the difficulty of the task. Generally, techniques used to reduce the domain gap categorize as feature-level or image-level. At the feature-level, the focus is on the extraction of discriminative features invariant to the factors (cross-spectrum, cross-sensor, or cross-resolution). For cross-spectral iris recognition, Abdullah *et al.* [10] propose three descriptors: Gabor-difference of Gaussian (G-DoG), Gabor-binarized statistical image feature (G-BSIF), and Gabor-multi-scale Weberface (G-MSW) as well as a fusion of these features at the decision-level. Oktiana *et al.* [192] propose phase-based features utilizing phase-only correlation

(POC) and band-limited phase-only correlation (BLPOC). Wang and Kumar [287] investigate a range of deep learning architectures: CNN with softmax cross-entropy loss, Siamese network, and triplet network. Regarding cross-spectral ocular recognition, Sharma *et al.* [248] propose combined neural network architecture, which first trains two neural networks separately on each spectrum and then jointly learns the cross-spectral variability using cross-spectral training data. Raja *et al.* [215] utilize Binarized Statistical Image Features (BSIF) and perform matching using Chi-Square distance. Later, Raja *et al.* [216] propose another method based on steerable pyramid features and a multi-class SVM classifier. At the image-level, the domain gap is reduced by transforming one domain image into another. Burge and Monaco [43] approximate a NIR iris image using features derived from the color and structure of the VIS iris image. Zuo *et al.* [319] generate a NIR iris image from the VIS iris image using a feed-forward neural network. Ramaiah and Kumar [222, 223] synthesize visible texture from the NIR image using Markov Random Fields (MRF) for both iris and ocular images. More recently, Hernandez-Diaz *et al.* [110] propose Conditional Generative Adversarial Networks (cGAN) for synthesizing a NIR ocular image from the VIS ocular image or vice versa.

Another major challenge in cross-modal recognition is imbalanced train data, where the number of pairs from different individuals (impostor pairs) is very high compared to pairs from the same individual (genuine pairs). As far as we know, no work on cross-modal or spectrum iris recognition focuses on the challenge in this literature.

In this work, we focus on both these challenges and propose three deep learning approaches. The first is at the feature-level, where the aim is to extract common features from both the images and the method is called Multi-channel CNN. It is a convolution neural network (CNN) that inputs face and iris images as different channels and extract common features together. The second strategy is at the training-level, where we generate synthetic training samples to increase the training data for learning as the number of genuine pairs in the cross-modal setting is insufficient for the training of deep architecture. We use Dual Variational Generation (DVG) framework [88] to synthesize the genuine pairs for training. The third strategy is at the image-level, where we transform one modality image

into another modality using the Generative Adversarial Network (GAN) framework. We use various GAN architectures for image-to-image translation, such as BicycleGAN [318], ESRGAN [289], Pix2Pix GAN [125], and StarGANv2 [53]. Here, we present the results of Pix2Pix GAN [125], BicycleGAN [318] and StarGANv2 [53] as these GAN frameworks perform the best. The main contributions of the work are as follows:

1. We propose deep learning approaches at three different levels (feature-level, image-level, and training-level) to address the domain gap and imbalanced training data challenges for cross-modal recognition.

2. We evaluate the performance of the proposed approaches on four cross-modal datasets: BioCop-2008, BioCop-2009, cross-spectrum PolyU, and WVU datasets.

Section 8.2 explains the architectural details of the proposed approaches. Section 8.3 describes datasets used in this work. Section 8.4 describes the experimental setup and results on both the datasets. Section 8.5 reports the impact of eye color on the performance of cross-model matching. Section 8.6 summarize the chapter.

8.2 Proposed Approaches and Rationale

In this section, we explain the two main challenges of cross-modal matching (domain gap and imbalanced training data) and the proposed approaches to address them.

To understand the domain gap, we conduct an initial analysis using histograms of ocular images under the VIS intra-modal, NIR intra-modal, and cross-modal scenarios. The analysis is on the randomly selected subset (5,000 genuine pairs and 5,000 impostor pairs) of ocular images from the BioCop-2008 dataset. We crop ocular regions from face images to form VIS ocular images, which are of low resolution. We generate histograms using similarity scores among genuine and impostor pairs, where similarity scores are computed using the Structural Similarity (SSIM) index [290]. Figure 8.2 shows the histograms of all three scenarios. Below each histogram statistics are provided in terms of genuine distribution mean, impostor distribution mean, d-prime (distance between two

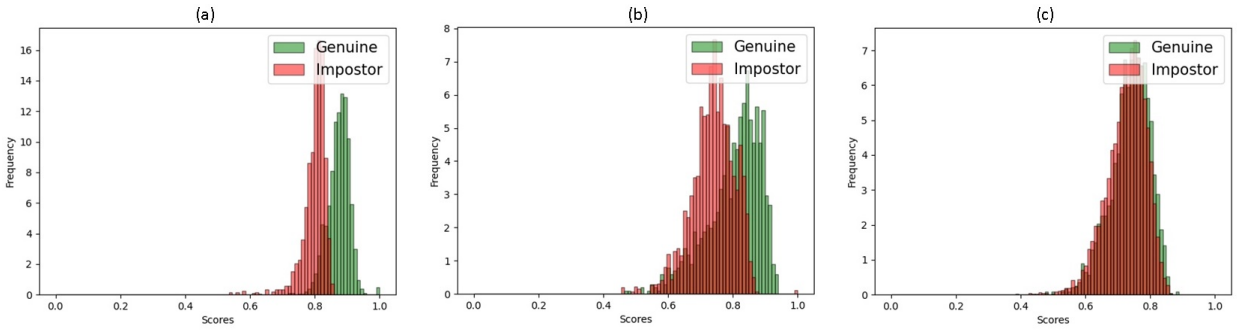


Figure 8.2: Histograms of similarity scores obtained from ocular images under (a) intra-modal VIS, (b) intra-modal NIR, and (c) cross-modal scenario. Similarity scores are estimated using the Structural Similarity (SSIM) index on ocular images of the BioCop-2008 dataset. The statistics of the histograms are given below each figure. There are two observations: first, the similarity between genuine pairs (Genuine Mean) reduces in the cross-modal scenario as compared to the intra-modal scenario; second, the overlapping area between two distributions increases dramatically for the cross-modal. For accurate matching, the overlapping area should be as minimum as possible.

distributions) [170], and overlap area of distributions. There are two noteworthy observations: first, the similarity between genuine pairs (Genuine Mean) reduces under cross-modal scenario; second, the overlapping area increases dramatically for cross-modal (genuine and impostor distributions are almost overlapping). Due to the large domain gap, the similarity of genuine pairs overlaps the imposter pairs to a significant extent. To reduce the domain gap, we propose two approaches one at the feature-level and the other at the image-level. The feature-level solution (Multi-channel CNN) jointly learns discriminative features from a pair of cross-modal images and outputs similarity score. For image-level solution, we translate one domain image into another using various GAN architectures (Pix2Pix GAN [125], BicycleGAN [318] and StarGANv2 [53]) before matching.

The second major challenge in cross-modal matching is imbalanced training data or insufficient genuine pairs training data. For example, if we consider genuine and impostor pairs of BioCop-2008 and PolyU datasets, BioCop-2008 contains 3,534 genuine pairs and 2,287,398 impostor pairs, whereas the PolyU dataset contains 6,287 genuine pairs and 10,630,762 impostor pairs. The genuine-impostor pairs ratio for BioCop-2008 and PolyU datasets are 1:645 and 1:1690, respectively. This is the case for any biometric dataset in verification mode. A large number of datasets are available for intra-modal matching, whereas only a few datasets exist for cross-modal

matching. To address the imbalanced data, we synthesize a large number of genuine pairs using the DVG framework [88] to augment the training data. A description of all three approaches, feature-level, image-level, and training-level are as follows:

8.2.1 Feature-level Approach: Multi-channel CNN (MT-CNN)

In the first method, we attempt to reduce the domain gap with a Multi-channel CNN (MT-CNN). Generally, deep networks input three input channels, namely the Red (R), Green (G), and Blue (B) channels. In MT-CNN (Figure 8.3), we use six input channels, where three correspond to the VIS image (R, G, and B channels) and the other three to the NIR image (one NIR channel repeated thrice). The motivation comes from Aguilera *et al.* [11] work, which utilizes a two-channel architecture for similarity measurement from a pair of natural images and outperforms Siamese and Pseudo-Siamese networks. We also applied two channels (one for VIS and the other for NIR image), four channels (three for VIS and one for NIR image), and the Siamese network. We report the best network (six channels) in the result section. The six-channel network does not compress the information as in the case of a two-channel CNN, where R, G, and B channels compress into one gray channel. The six-channel network also has an advantage over the four-channel network as it equally weights both VIS and NIR images. In contrast to the Siamese network, where weights are shared in the later layers, the MT-CNN jointly processes information from both images at the first layer of the network [311].

The backbone architectures used in the MT-CNN is DenseNet201 [122]. The base networks is pre-trained on the ImageNet dataset [72], and then fine-tuned on iris training data described in Section 3. The output is a similarity score between 0 and 1, where '0' implies an impostor pair and '1' implies a genuine pair. Training performed using Stochastic Gradient Descent with a learning rate of 0.005, weight decay of 10^{-6}, and a momentum of 0.9. The batch size is 20 and the number of epochs is 50.

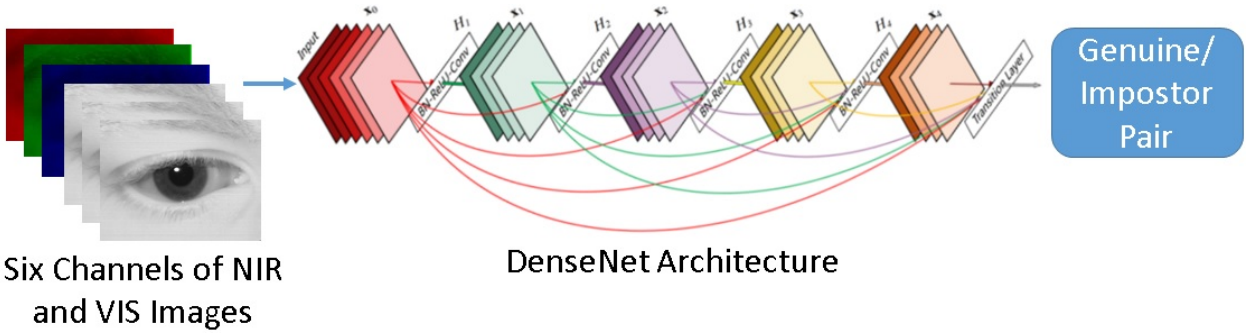


Figure 8.3: The architecture of Multi-channel CNN (MT-CNN). The base architecture used in the MT-CNN is DenseNet201 [122]. It estimates a similarity score between the images of the two domains.

8.2.2 Image-level Approach

This method aims to transform one modality image (cropped VIS iris image) into another modality (NIR iris image) image. For image-to-image translation, we utilize three GAN architectures: Pix2Pix GAN [125], BicycleGAN [318] and StarGANv2 [53]. We translate a low-resolution (301 x 201) visible spectrum (VIS) iris image to a high-resolution (640 x 480) near-infrared spectrum (NIR) iris image. We are performing VIS to NIR image translation instead of NIR to VIS as: (i) NIR discerns iris patterns more effectively compared to VIS image, and translation from NIR to VIS loses relevant information, and (ii) compression of three channels of VIS image into one channel of NIR image is easier than an expansion of one channel of NIR image into three channels of VIS image. After image translation, we calculate similarity score of the GAN-generated NIR image with the real NIR image using Multi-channel CNN. We utilize the same losses for BicycleGAN [318] and StarGANv2 [53] as specified in the original work, whereas for Pix2Pix GAN we include additional identification loss. Therefore, we only describe the Pix2Pix GAN below.

8.2.2.1 Pix2Pix GAN with Identification Loss (Pix2Pix GAN ID)

We attempt to reduce the domain gap using a deep generative model, Pix2Pix GAN [125]. It is a conditional Generative Adversarial Network (cGAN) designed for image-to-image translation. We introduce identification loss into its objective function and term it as Pix2Pix GAN ID. Our

objective is to synthesize NIR image of the same identity as VIS image and match it against the real NIR image. Figure 8.4 represents the overall testing setup of cross-modal matching utilizing Pix2Pix GAN ID and MT-CNN.

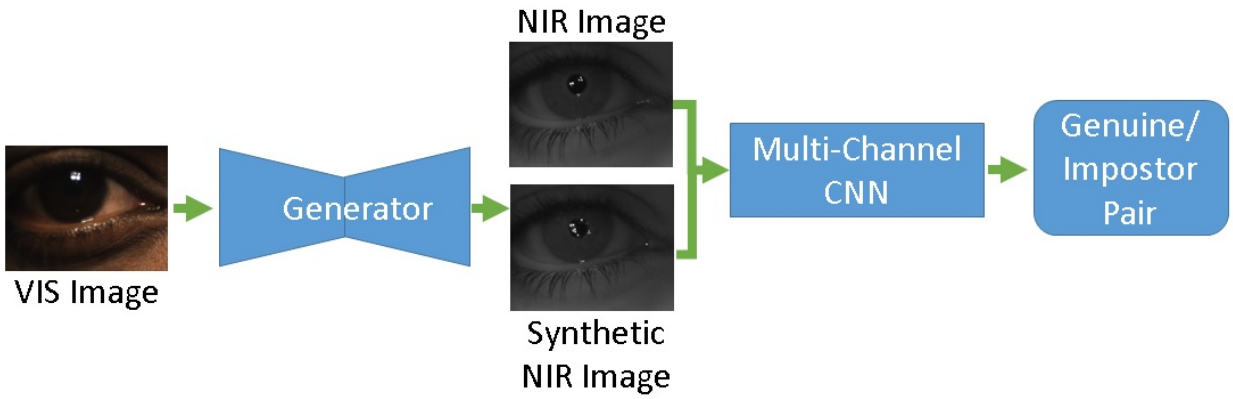


Figure 8.4: The overall testing scenario of Pix2Pix GAN ID and MT-CNN for cross-modal matching. The Pix2Pix GAN ID's generator synthesizes a NIR image from the VIS image. The MT-CNN then generates a similarity score from a pair of synthesized NIR and real NIR images.

There are two components of Pix2Pix GAN ID: a generator and a discriminator. The generator aims to generate a realistic image with a constraint that the image should retain iris biometric information, whereas the discriminator distinguishes between real and synthesized NIR images. The base architecture used for the generator is U-Net256 [232] with skip connections. The discriminator used is PatchGAN [164] classifier. The training of the generator and discriminator is performed using the following loss terms:

1. **Adversarial Loss:** It is a classical adversarial loss, where the discriminator and generator compete with each other until reaching an equilibrium. The adversarial loss is defined as follows:

$$L_{GAN}(G, D) = E_{x,y}[logD(x, y)] + E_x[log(1 - D(x, G(x)))] \tag{8.2.1}$$

where, G is the generator function, D is the discriminator function, x is the input VIS image, and y is the target NIR image. The generator aims to minimize the objective, whereas the discriminator tries to maximize it. The generator is not directly affecting the $log(D(x, y))$

term in the function, so for the generator, minimizing the loss is equivalent to minimizing $E_x[log(1 - D(x, G(x))])$.

2. **Per-pixel Loss:** Per-pixel loss computes l_1 distance between two images at the pixel level and reduces the mapping space from the VIS spectrum to the NIR spectrum. The loss formulation is as follows:

$$L_{per-pixel}(G) = E_{x,y} \parallel G(x) - y \parallel_1 \tag{8.2.2}$$

where, $\parallel * \parallel_1$ is the l_1 norm between synthetic $(G(x))$ and real (x) images.

3. **Perceptual Loss:** It is the l_1 distance between deep features extracted from synthetic and real images. The features are extracted at multiple layers of the VGG19 network and concatenated to form a single feature descriptor. The VGG19 network is pre-trained on the ImageNet dataset [73]. The formulation is as follows:

$$L_{perceptual}(G) = E_{x,y} \parallel \phi_P(G(x)) - \phi_P(y) \parallel_1 \tag{8.2.3}$$

where, $\phi_P(G(x))$ and $\phi_P(y)$ are the VGG features extracted from the synthetic NIR image and the real NIR image, respectively. Perceptual loss [132] helps the generator to minimize the high-level semantic difference between the images. It ensures the smoothness and visual similarity of the generated image with the real NIR image.

4. **Identity Loss:** It is a cross-entropy loss estimated using MT-CNN, where input is a pair of real and synthetic NIR images, and output is a similarity score (1 for genuine pairs and 0 for impostor pairs). Its formulation is as follows:

$$L_{identity}(G) = -(t \log(M(G(x), y)) + (1 - t)log(1 - M(G(x), y)))) \tag{8.2.4}$$

where, $M(G(x), y)$ is the similarity score output by MT-CNN when synthetic NIR image $G(x)$ and real NIR image y are given as input, t is the ground-truth label specifying that input pair is genuine or an impostor. As we are using only genuine pairs for training Pix2Pix GAN ID, the identity loss reduced to $L_{identity}(G) = -(\log(M(G(x), y))$.

The overall loss function is as follows:

$$G^* = argmin_G(L_{GAN} + L_{per-pixel} + L_{perceptual} + L_{identity})$$
$$D^* = argmax_D L_{GAN} \tag{8.2.5}$$

8.2.3 Training-level: Dual Variational Generation

Another challenge of cross-modal matching is insufficient genuine training data, which also raises the imbalance issue between genuine and impostor pairs. To address the challenge, we utilize another deep generative framework: Dual Variational Generation (DVG) [88]. The DVG network is an unconditional variational autoencoder (VAE) that generates NIR and VIS paired images of the same identity from noise sampled from a standard normal distribution. The MT-CNN training is supplemented with the synthetic samples (i.e., the network is trained using both real VIS-NIR genuine pairs and synthetic VIS-NIR genuine pairs from the DVG). Figure 8.5 shows the overall architecture of cross-modal recognition training utilizing the DVG-based method and MT-CNN. Regarding identity constraint, the image-to-image translation focuses on identity preservation, where the identity of a synthesized image is the same as the input image. On the other hand, the DVG-based method focuses on the identity consistency of generated pairs, which is anonymous. Generally, for the image-to-image translation, only a few samples are available for learning identity preservation, whereas the DVG-based model utilizes entire training genuine pairs for identity consistency.

The architecture of DVG consists of two encoders that correspond to NIR and VIS input images and a decoder. Figure 8.6 represents the training architecture of the DVG model. Encoder E_N is responsible for mapping input NIR image x_N to latent space z_N, whereas encoder E_V is responsible for mapping input VIS image x_V to latent space z_V. The latent representation of NIR and VIS images is then concatenated and fed into the decoder, which reconstructs the NIR and VIS images. Training of DVG-based model is performed using the following loss terms:

1. **KL Divergence Loss:** The constraint is applied over the encoders E_N and E_V which outputs

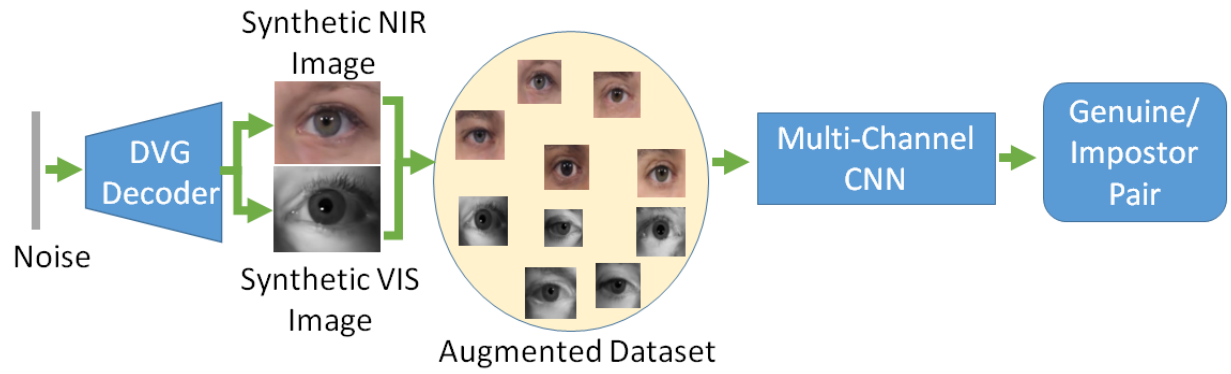


Figure 8.5: Training architecture of the DVG-based model. The figure is adapted from [88]. It consists of two encoders that correspond to NIR and VIS input images and a decoder. The encoder transforms input image space into latent space. The decoder utilizes the latent space of NIR and VIS images and reconstructs them back into the image space.

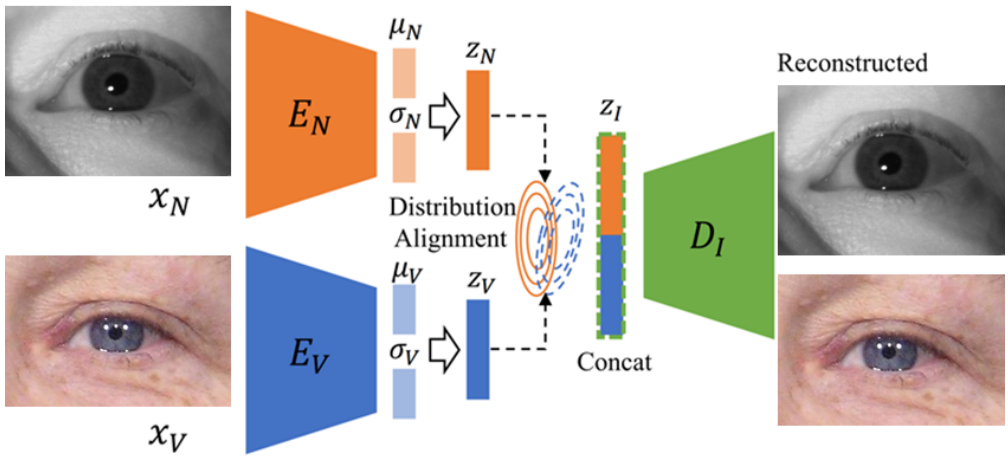


Figure 8.6: Training procedure of the DVG-based method.

posterior distributions $q_{\phi_N}(z_N \mid x_N)$ and $q_{\phi_V}(z_V \mid x_V)$ using Kullback-Leibler divergence:

$$Loss_{KL} = D_{KL}(q_{\phi_N}(z_N \mid x_N) \parallel p(z_N)) + D_{KL}(q_{\phi_V}(z_V \mid x_V)) \parallel p(z_V)) \tag{8.2.6}$$

where x_* is the NIR or VIS input, z_* is the NIR or VIS latent output, $p(z_*)$ is the NIR or VIS prior distribution, and $q_{\phi_*}(z_* \mid x_*)$ is the posterior distribution output by the NIR or VIS encoder. We assume a multivariate standard normal distribution for the prior distributions.

2. **Reconstruction Loss:** This constraint is applied over the decoder, which reconstructs the

input images x_N and x_V. The formulation is as follows:

$$Loss_{rec} = -E_{q_{\theta_N}(z_N|x_N) \cup q_{\phi_V}(z_V|x_V)}[\log p_\theta(x_N, x_V \mid z_I)] \tag{8.2.7}$$

where z_I is the concatenation of z_N and z_V latent vectors, and $p_\theta(x_N, x_V \mid z_I)$ is the joint distribution. The reconstruction loss is calculated by the root mean square of input and reconstructed images.

3. **Distribution Alignment Loss:** We aim to project both NIR and VIS images to a common latent space. To achieve this, we minimize the Wasserstein distance [107] between the posterior distributions of NIR and VIS images. The loss formulation is as follows:

$$Loss_{dist} = \frac{1}{2} \parallel \mu_N - \mu_V \parallel_2^2 + \parallel \sigma_N - \sigma_V \parallel_2^2 \tag{8.2.8}$$

where μ_N and σ_N are the mean and standard deviation output by the NIR encoder E_N, and μ_V and σ_V are the mean and standard deviation output by VIS encoder E_V.

4. **Identity Loss:** Another constraint is to preserve the biometric content within the domain and ensure the identity is consistent across the domains. We use MT-CNN for both identity preservation and consistency.

   a) *NIR-VIS pair identity loss:* A cross-entropy loss is estimated using MT-CNN to keep the same identity across the reconstructed NIR and VIS images. Its formulation is as follows:

   $$Loss_{id-pair} = -log(M_{NV}(\widehat{x}_N, \widehat{x}_V)) \tag{8.2.9}$$

   where $\widehat{x}_N$ is the DVG reconstructed NIR image, $\widehat{x}_V$ is the DVG reconstructed VIS image and $M(\widehat{x}_N, \widehat{x}_V)$ is the MT-CNN having NIR and VIS image as input.

   b) *NIR-NIR and VIS-VIS identity loss:* It is a cross-entropy loss to preserve the identity within the domain. Reconstructed NIR or VIS images should be of the same identity as the original NIR or VIS images. Its formulation is as follows:

   $$Loss_{id-rec} = -\log(M_N(\widehat{x}_N, x_N)) - \log(M_V(\widehat{x}_V, x_V)) \tag{8.2.10}$$

where x_N is the NIR input image, x_V is the VIS input image, $M_N(*,*)$ is the MT-CNN for NIR images, and $M_V(*,*)$ is the MT-CNN for VIS images. The original framework [88] utilizes a l_2 norm over the features extracted from Light-CNN to reserve the identity.

The overall objective for the DVG-based model is as follows:

$$\begin{aligned} Loss_{Overall} &= Loss_{rec} + \gamma_{KL} Loss_{KL} + \gamma_{dist} Loss_{dist} \\ &+ \gamma_{id-pair} Loss_{id-pair} + \gamma_{id-rec} Loss_{id-rec} \end{aligned} \tag{8.2.11}$$

where γ_* are empirically set to 0.1.

During the testing of the DVG-based method (Figure 8.7), we discard both the encoders and only utilize the decoder. Noise is sampled from the standard normal distribution and the same is concatenated with itself and fed into the decoder which generates NIR and visible images of the same identity. The decoder generates genuine pairs that can be included in the training process of the Multi-channel CNN.

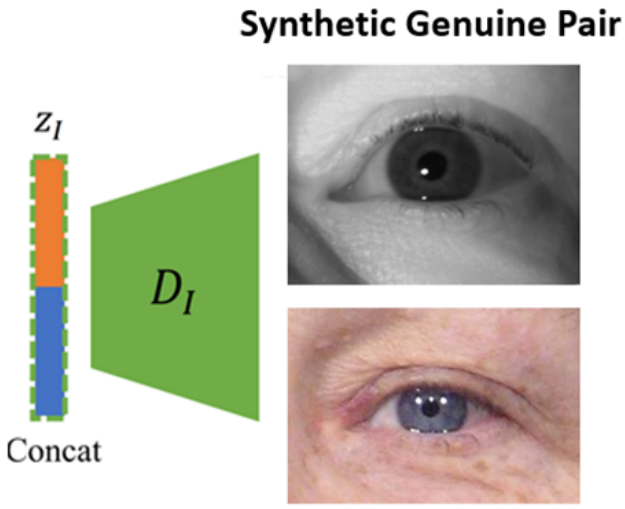


Figure 8.7: The testing procedure of the DVG-based method. Noise is an input to the Decoder D_I which generates a synthesized genuine pair.

8.3 Dataset Description

8.3.1 BioCop-2008 Dataset

The first dataset we use for our experiments is the FBI Biometric Collection of People (BioCoP-2008) dataset. The BioCoP-2008 is an extension of the dataset mentioned in [129]. It is a multi-modal biometric dataset consisting of the face and ocular images collected from two sessions (SET1 and SET2). Subjects are the same in both sessions. The face images are acquired in visible illumination by Olympus C8080 camera from 1,135 subjects, whereas the ocular images are acquired in NIR illumination by Oki IrisPass M iris sensor from 1,097 subjects. The images are not simultaneously captured, so images are not aligned. There are a total of 3,608 iris images and 2,270 frontal face images. The dataset also consists of face images with 45 and 90-degree pose angles, but we utilize only the frontal images.

The original size of VIS face images is 3264×2448. We cropped left and right ocular regions from the face images, which results in the ocular images of size 301×201. We further cropped the left and right iris regions from the left and right ocular images, respectively. The size of the cropped iris images is 81×81. The original size of NIR ocular images is 640×480. We cropped the left and right iris regions from the left and right NIR ocular images, respectively. The size of the cropped NIR iris images is 180×190. Figure 8.8 shows the original VIS face image, cropped VIS ocular image, cropped VIS iris image, and their corresponding NIR images.

8.3.2 BioCop-2009 Dataset

The second dataset we use for our experiments is the FBI Biometric Collection of People (BioCoP-2009) dataset. It is dataset is also a multi-modal biometric dataset consisting of images of a face and iris modalities. Subjects are the same in both the modalities collection. The face images are collected in two sessions from 1,100 subjects using Canon EOS 5D Mark II camera in the visible spectrum. Images are provided with different angles (-90, -45, 0, 45, and 90) and scales (raw, SAP50, and SAP51). We utilize only the frontal face (angle of 0) with SAP51 scale face images,

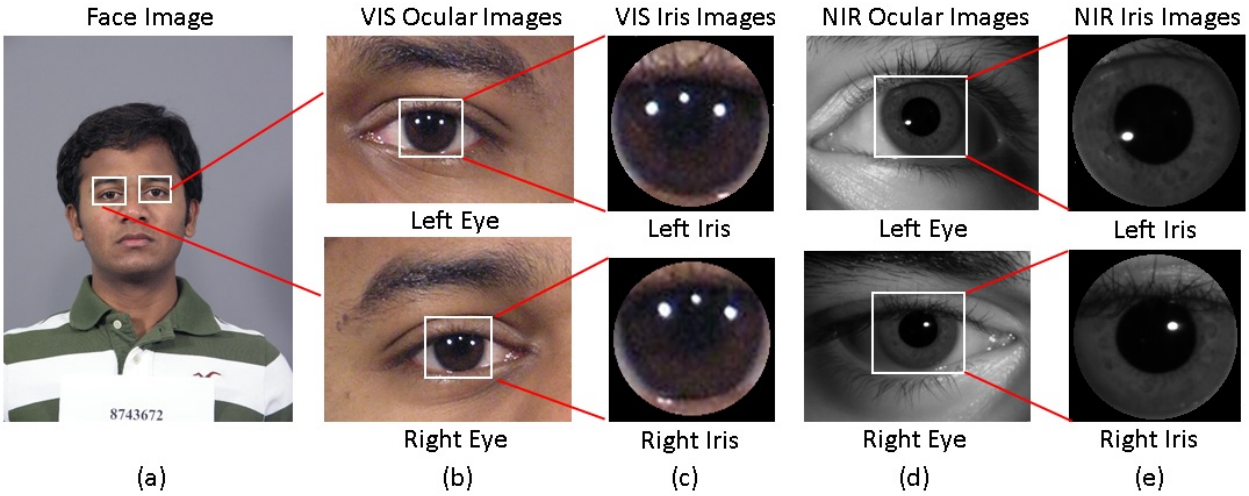


Figure 8.8: (a) A sample face image from the BioCop-2008 dataset. The face image is in the VIS spectrum. (b) Cropped left and right VIS ocular images from the face image. (c) Cropped left and right iris images from the left and right ocular images, respectively. The size of ocular and iris VIS images are 301 × 201 and 81 × 81, respectively. (d) Left and right NIR ocular images from the BioCop-2008 dataset. (e) Cropped left and right iris images from the left and right NIR ocular images, respectively. The size of ocular and iris NIR images are 640 × 480 and 180 × 190, respectively.

so it results in a total of 2,199 face images. The resolution of face images is 2400 x 3200. We manually crop left and right ocular images from the face images. The size of the ocular images is 402 x 301. We further crop left and right iris regions from the left and right ocular images, respectively. The size of iris images varies according to the iris region.

The data collected from the iris modality is acquired from 1,098 subjects in the NIR spectrum using three sensors: Aoptix Insight, CrossMatch I SCAN 2, and LG ICAM 4000. There are five sessions for each sensor. In each session, there are 2 images (one left and one right) from Aoptix Insight and CrossMatch I SCAN 2 sensors and 4 images from LG ICAM 4000 sensor. The total number of images from the Aoptix Insight sensor is 11,000, from CrossMatch I SCAN 2 is 10,910, and from LG ICAM 4000 is 21,980. The image size of NIR ocular images is 640 x 480. We further crop left and right iris regions from left and right NIR ocular images, respectively. The size of the cropped iris images varies according to the iris region.

8.3.3 PolyU Dataset

The third dataset we use for our experiments is the publicly available PolyU dataset [223]. It is a bi-spectral dataset used to analyze the cross-spectral iris recognition algorithms. The sensor used for the data collection is an in-house imaging setup that acquires NIR and VIS iris images simultaneously in a single shot. The two spectral images collected have pixel-to-pixel correspondences. The dataset consists of images from two sessions. In the first session, there are images of 209 subjects. Approximate 15 images are there for each eye (left and right). In the second session, there are images of 11 subjects. The subjects are the same in both sessions. The total number of images from both sessions are 12,574. The resolution of both spectral images is 640×480. The first two rows of Figure 8.9 show a few samples of the PolyU dataset.

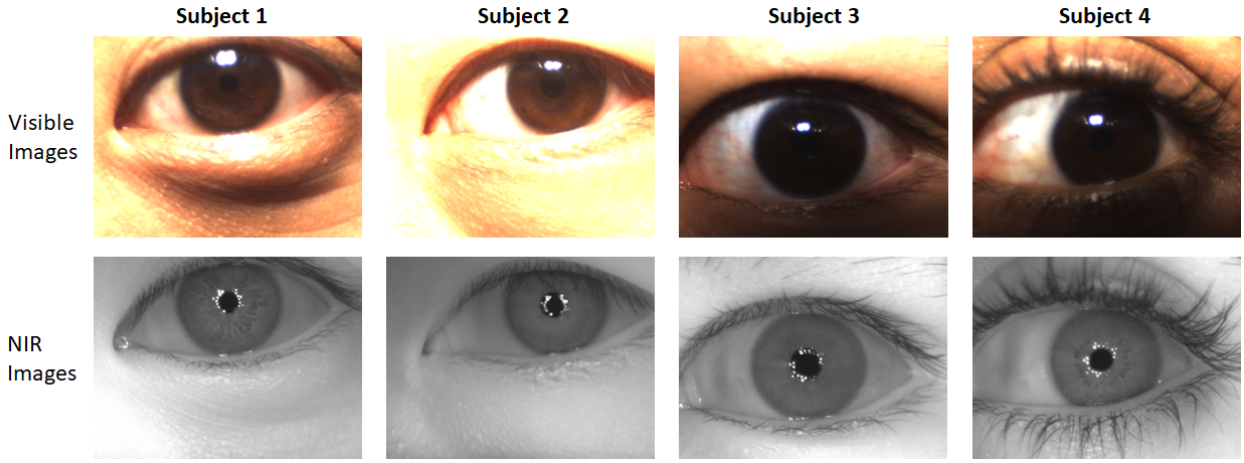


Figure 8.9: Samples of VIS and NIR ocular images from the PolyU dataset. The first and second row represents the corresponding VIS and NIR ocular images of four different subjects, respectively.

8.3.4 WVU Dataset

The WVU multimodal dataset [55] is a biometric dataset consists of images from face, iris, fingerprint, hand geometry, palmprint, and voice modalities. We utilize only face and iris modality images. The face images are acquired in visible illumination using Sony EVI-D30 and Sony EVI-D31 cameras from 269 subjects. The iris images are acquired in NIR illumination using the Irispass

iris sensor from 244 subjects. There are 234 subjects common in both modalities (face and iris). The total number of iris images is 3,099, and frontal face images is 1,746.

The original size of the VIS face images captured from two sensors are 768×576 and 640×480. We crop left and right ocular regions from the face image, which results in the ocular image of size approx. 51×61 (varies as per the size of the ocular region). We further crop left and right iris regions from the left and right ocular images, respectively. The size of the cropped iris images is approximately 24×24 (varies as per the size of the iris region). The original size of NIR ocular images is 640×480. We crop left and right iris regions from the left and right NIR ocular images, respectively. The size of the cropped NIR iris images is approximately 300×300 (varies as per the size of the iris region). Figure 8.10 shows the original VIS face image, cropped VIS ocular image, cropped VIS iris image, and their corresponding NIR images.

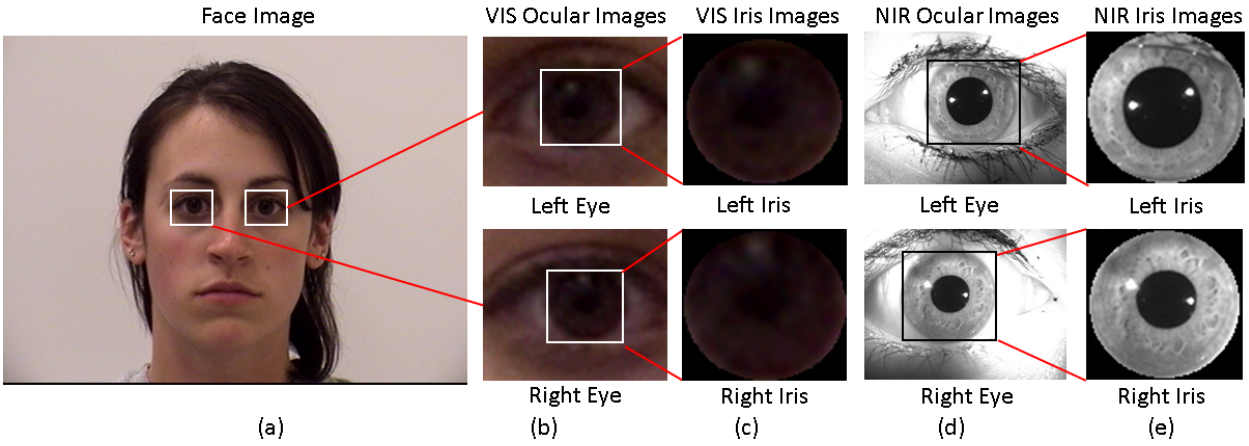


Figure 8.10: (a) A sample face image from the WVU dataset. The face image is in the VIS spectrum. (b) Cropped left and right VIS ocular images from the face image. (c) Cropped left and right iris images from the left and right ocular images, respectively. The size of ocular and iris VIS images are 51×61 and 24×24, respectively. (d) Left and right NIR ocular images from the WVU dataset. (e) Cropped left and right iris images from the left and right NIR ocular images, respectively. The size of ocular and iris NIR images are 640×480 and 300×300, respectively.

8.4 Experimental Setup and Results

8.4.1 BioCop-2008 and BioCop-2009 Dataset

For the evaluation of cross-modal matching on BioCop-2008 and BioCop-2009 datasets, we consider three matching scenarios and two types of input (iris and ocular). In the first scenario (Face-Face), the iris or ocular region from the face visible image is matched against the face visible image. In the second (Iris-Iris), the iris or ocular region from the iris NIR image (original dataset image) is matched against the iris NIR image. In the last scenario (Iris-Face), the iris or ocular region from the iris NIR image is matched against the face visible image. In the BioCop-2009 dataset, there are three experiments in Iris-Iris and Iris-Face scenarios corresponding to iris images captured from three different iris sensors (Aoptix, CrossMatch, and LG4000). In all experiments, we perform training on 70% of subjects and testing on the rest (30%) using the subject disjoint protocol. Table 8.1 provides the details on the genuine and impostor pairs used for training and testing in all three scenarios for the BioCop-2008 dataset. Table 8.2 provides the same details for the BioCop-2009 dataset. We utilize the entire training genuine pairs, but partial impostor pairs (50,000) to reduce time complexity. However, testing is performed on the entire genuine and impostor pairs. We repeat the random selection of impostor pairs for training five times and report the cross-validation results. We perform iris recognition using VeriEye, USITv3.0, MT-CNN techniques, and ocular recognition using MT-CNN. The VeriEye is a commercially available off-the-shelf technique that performs iris recognition. It is used as a baseline. The USITv3.0 is an open-source iris recognition software toolkit from the University of Salzburg Iris Toolkit. We utilize the best-performing technique from the toolkit. The technique extracts iris-code using quadratic spline wavelet (QSW) and uses hamming distance to measure the dissimilarity between the iris codes. The technique also performs iris recognition and is considered a baseline for our cross-modal evaluation. We manually segment the iris images for USITv3.0, whereas VeriEye utilizes its iris segmentation module. Evaluation measures used in the experiments are True Match Rate (TMR) at 0.1% False Match Rate (FMR) and Equal Error Rate (EER). Tables 8.3 and 8.4 present the performance of all methods on the

Table 8.1: Description of genuine and impostor pairs used in experiments from the BioCop-2008 dataset.

	Train Set			**Test Set**	
	Genuine Pairs	Impostor Pairs	Impostor Pairs Used	Genuine Pairs	Impostor Pairs
Face-Face	1,588	629,642	50,000	682	115,940
Iris-Iris	1,044	389,713	50,000	425	67,906
Face-iris	2,448	1,875,168	50,000	1,030	338,870

Table 8.2: Description of genuine and impostor pairs used in experiments from the BioCop-2009 dataset.

	Train Set			**Test Set**	
	Genuine Pairs	Impostor Pairs	Impostor Pairs Used	Genuine Pairs	Impostor Pairs
Face-Face	1,538	590,592	50,000	660	109,230
Iris-Iris (Aoptix)	15,550	587,522	50,000	6,600	107,912
Iris-Iris (CrossMatch)	15,320	575,322	50,000	6,600	107,912
Iris-Iris (LG4000)	69,610	587,522	50,000	29,700	107,912
Face-iris (Aoptix)	15,420	1,178,112	50,000	6,580	215,824
Face-Iris (CrossMatch)	15,300	1,170,442	50,000	6,520	213,850
Face-Iris (LG4000)	30,800	1,178,112	50,000	13,160	215,824

BioCop-2008 and BioCop-2009 datasets, respectively. Figure 8.11 shows the ROC curves of four methods in the Iris-Face matching scenario and the histogram corresponds to the MT-CNN on the BioCop-2008 dataset. Figures 8.12a, 8.12b, and 8.12c show the ROC curves of four methods in the Iris-Face matching scenario corresponding to three iris sensors images and 8.12d shows histogram corresponds to the MT-CNN method on the BioCop-2009 dataset.

The MT-CNN performs the best in Face-Face and Iris-Face matching scenarios on both datasets. VeriEye technique performs the best in the case of Iris-Iris matching on the BioCop-2008 dataset and the MT-CNN on the BioCop-2009 dataset. There occur a few segmentation failures in the

Table 8.3: Performance of different methods on the BioCop-2008 dataset. MT-CNN with ocular input outperforms on this dataset.

Experiments	**VeriEye**		**USITv3.0**		**MT-CNN(Iris)**		**MT-CNN(Ocular)**	
	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)
Face-Face	47.38	21.84	45.60	20.74	95.74	0.86	**98.53**	**0.84**
Iris-Iris	**98.80**	**0.67**	95.76	1.64	96.70	1.20	82.58	4.67
Iris-Face	34.84	29.07	14.46	37.01	47.45±1.58	9.05±0.45	**50.46±2.48**	**7.32±0.47**

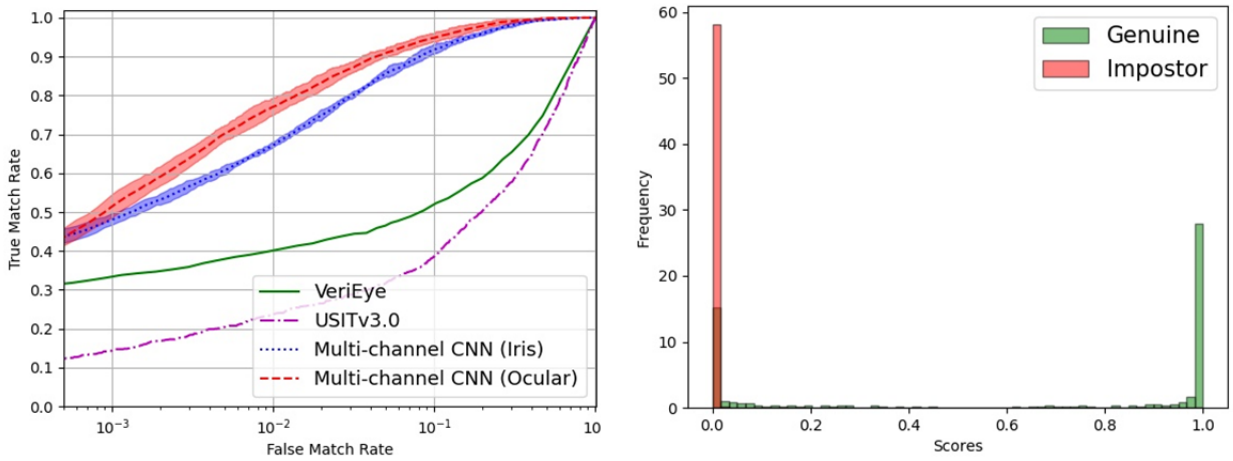


Figure 8.11: ROC curves of different methods and histogram (MT-CNN) in the Iris-Face matching scenario on the BioCop-2008 dataset. MT-CNN with ocular input outperforms on this dataset.

case of VeriEye, which are provided in Table 8.6. Figures 12 and 13 show a few failure cases from the BioCop-2008 and BioCop-2009 datasets, respectively. There is no clear winner between iris and ocular recognition when considering the MT-CNN. The results show the efficiency of the learning-based method (MT-CNN) over the hand-crafted features-based techniques in the cross-modal matching scenario.

In another experimental setup, we used a small set of 5,000 impostor pairs for the training and evaluate the DVG-based method on the BioCop-2008 dataset. Using the DVG-based method, we generate 50,000 genuine pairs and are included in the training process of the MT-CNN. The testing set remains the same as in Table 8.1. Only the cross-modal (Iris-Face) scenario is tested. Table 8.6 shows the results of MT-CNN when trained on only real genuine pairs and when trained on both real and synthetically generated genuine pairs (DVG-based method). As the DVG-based method

Table 8.4: Performance of different methods on the BioCop-2009 dataset. MT-CNN with iris input outperforms on Aoptix Insight and CrossMatch sensor images, whereas MT-CNN with ocular input outperforms on LG ICAM 4000 sensor images.

Experiments	**VeriEye**		**USITv3.0**		**MT-CNN(Iris)**		**MT-CNN(Ocular)**	
	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)
Face-Face	88.86	5.66	82.42	7.13	**98.18**	**0.44**	97.87	1.27
Iris-Iris (Aoptix)	96.40	3.24	94.93	3.97	**99.89**	**0.11**	99.83	0.12
Iris-Iris (CrossMatch)	99.96	0.03	99.89	0.10	**99.98**	**0.02**	99.56	0.31
Iris-Iris (LG4000)	99.85	0.14	99.60	0.34	**99.88**	**0.11**	97.04	0.44
Iris-Face (Aoptix)	51.30	25.38	29.76	32.88	**80.48**	**2.18**	46.79	2.36
Iris-Face (CrossMatch)	58.48	22.34	40.73	27.94	**76.13**	**2.88**	70.03	2.99
Iris-Face (LG4000)	28.05	21.09	39.60	29.39	89.38±0.95	2.16±0.15	**91.82±3.26**	**1.55±0.15**

Table 8.5: Number of genuine and impostor pairs excluded from the test set due to the segmentation errors by the VeriEye technique on both the datasets. The numbers shown in the parenthesis are the total number of genuine and impostor pairs used in the test set.

	BioCop-2008		**BioCop-2009**	
	Genuine Pairs	Impostor Pairs	Genuine Pairs	Impostor Pairs
Face-Face	12 (682)	1,357 (115,940)	17 (660)	3,926 (109,230)
Iris-Iris (Aoptix)	6 (425)	1,617 (67,906)	4 (6,600)	328 (107,912)
Iris-Iris (CrossMatch)	-	-	52 (6,600)	1,302 (107,912)
Iris-Iris (LG4000)	-	-	100 (29,700)	655 (107,912)
Face-iris (Aoptix)	24 (1,030)	7,611 (338,870)	100 (6,580)	2,943 (215,824)
Face-Iris (CrossMatch)	-	-	110 (6,520)	4,214 (213,850)
Face-Iris (LG4000)	-	-	40 (13,160)	692 (215,824)

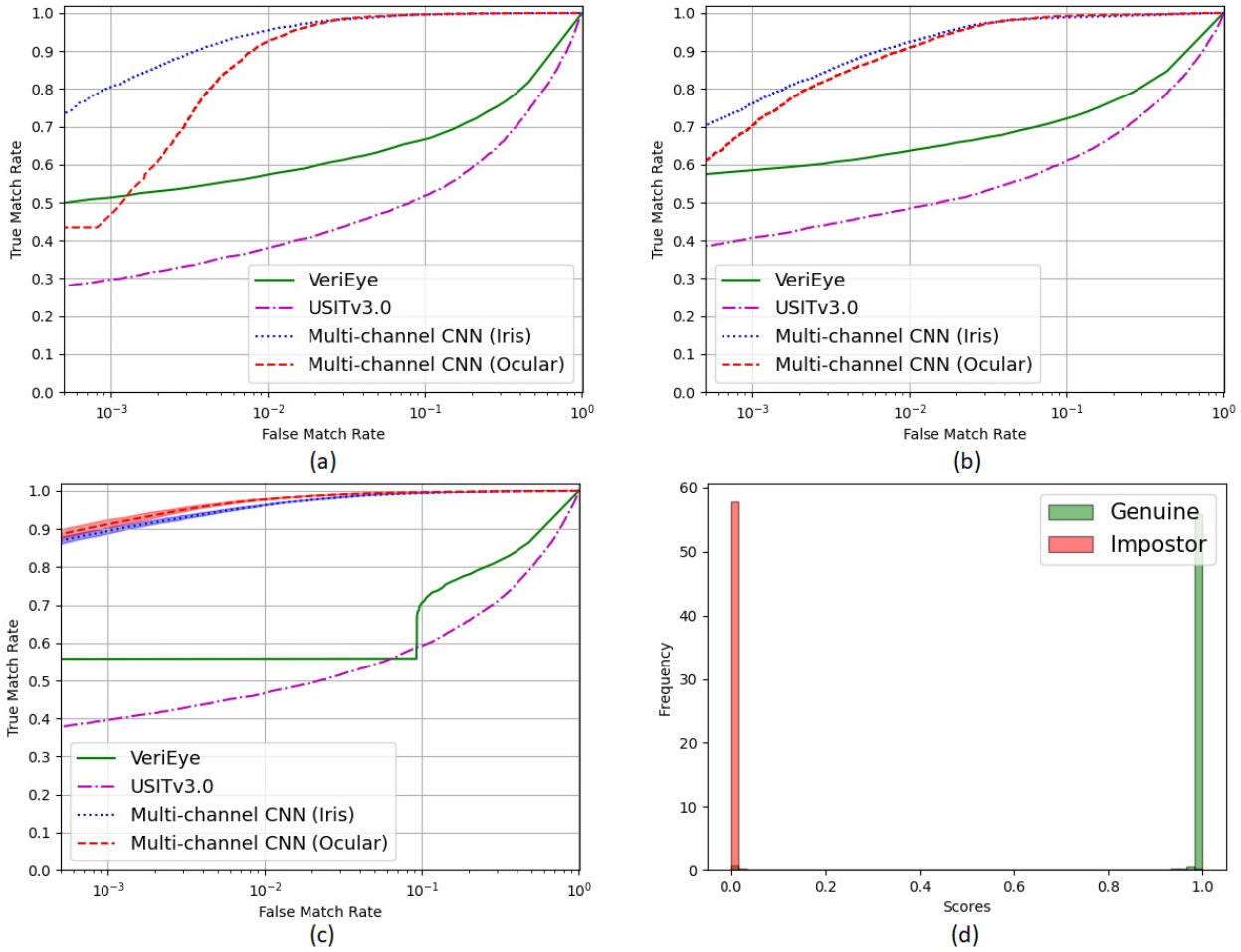


Figure 8.12: ROC curves of different methods in the Iris-Face matching scenario on the BioCop-2009 dataset corresponding to (a) Aoptix Insight, (b) CrossMatch I SCAN 2, and (c) LG ICAM 4000 iris sensors. MT-CNN with iris input outperforms on Aoptix Insight and CrossMatch sensor images, whereas MT-CNN with ocular input outperforms on LG ICAM 4000 sensor images. (d) Histogram corresponds to the MT-CNN method with ocular input on LG ICAM 4000 sensor images.

generates the genuine pairs from the distribution of available real genuine pairs, therefore there is no significant improvement occurred in the performance.

8.4.2 PolyU Dataset

For the evaluation of the cross-spectrum setting on the PolyU dataset, we use the subject-disjoint strategy in the experiments, where 60% of subjects were present in training and 40% in testing. It generates 4,067 genuine pairs and 8,199,862 impostor pairs for the training, whereas 2,220 genuine

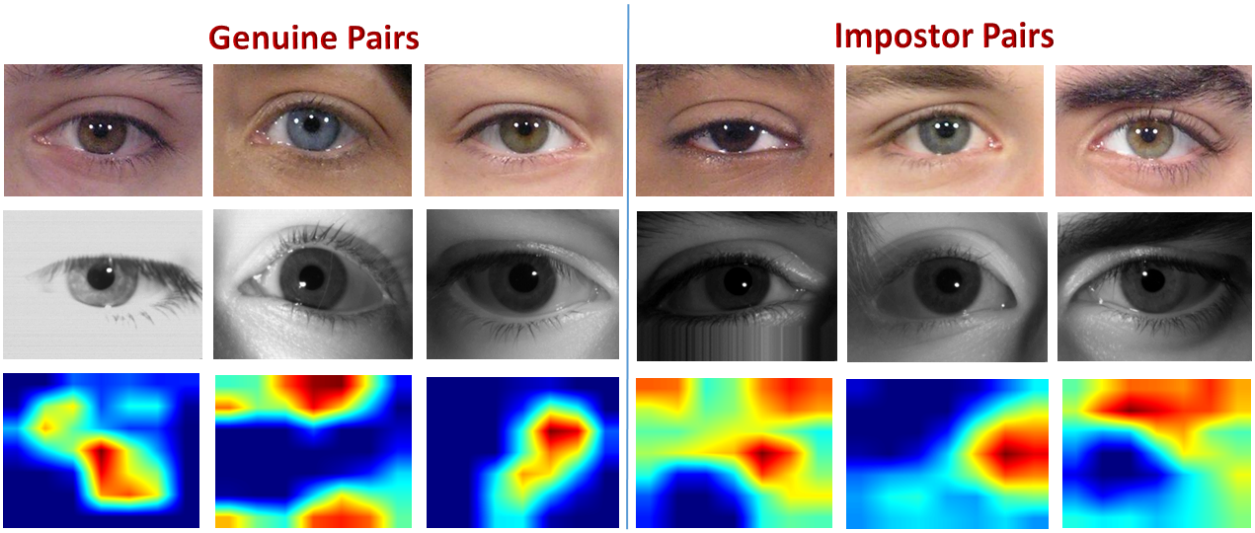


Figure 8.13: Failure cases of the MT-CNN in genuine and impostor pairs from the BioCop-2008 dataset. The last row represents the GradCam maps [8] which show regions focused by the network to make the decision.

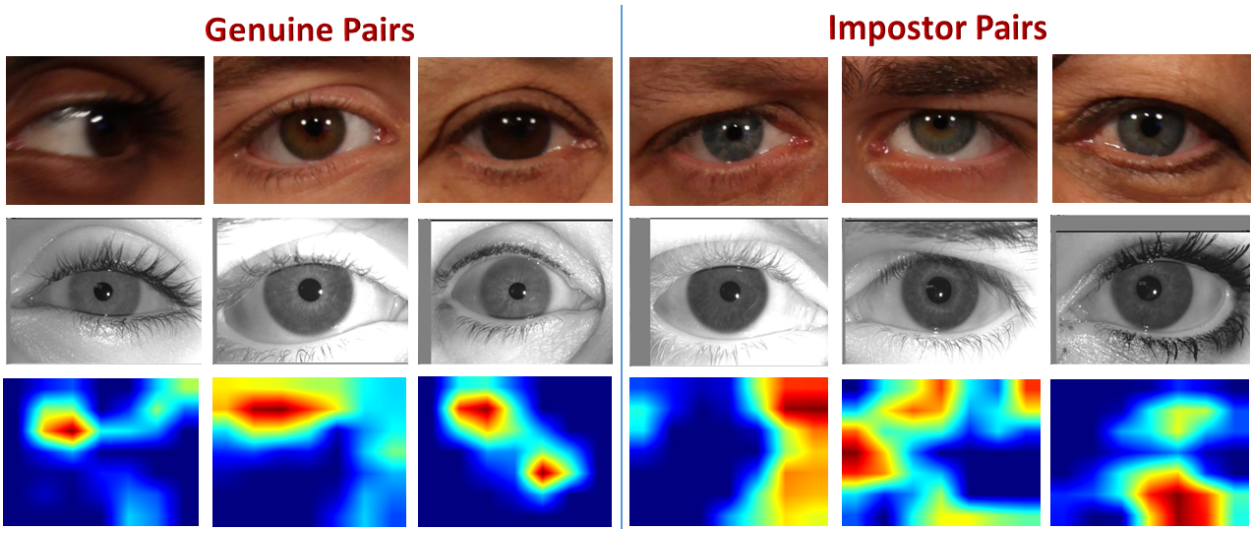


Figure 8.14: Failure cases of the MT-CNN in genuine and impostor pairs from the BioCop-2009 dataset. The last row represents the GradCam maps [8] which show regions focused by the network to make the decision.

pairs and 2,430,900 impostor pairs for the testing. From the train and test sets, we utilize all genuine pairs, but 10,000 randomly selected impostor pairs to reduce the computational time. Table 8.7 provides the number of genuine and impostor pairs used for training and testing. The evaluation measures used for the comparison are TMR (%) at 0.1% FMR and EER. The methods used for

Table 8.6: TMR and EER of ocular and iris recognition methods on the entire test set of BioCop-2008 dataset when a small set (5,000 impostor pairs) is used for the training. Including additional training samples generated from the DVG-based method does not improve the performance.

Experiments	**Ocular**		**Iris**	
	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)
Multi-channel CNN	28.84 ± 3.27	**9.69 ± 0.50**	**29.88 ± 4.07**	**10.87 ± 0.67**
DVG-based Method	**29.15 ± 1.45**	9.61 ± 0.41	28.63 ± 3.12	11.02 ± 0.66

Table 8.7: Data distribution among train and test sets from the PolyU dataset.

	Train Set		**Test Set**	
	Genuine Pairs	Impostor Pairs	Genuine Pairs	Impostor Pairs
PolyU Dataset	4,067	10,000	2,220	2,430,900

comparison on the dataset are VeriEye, MT-CNN, Pix2Pix GAN ID, StarGANv2, and BicycleGAN. Table 8.8 presents the results of all ocular and iris recognition algorithms. Figure 8.15 shows NIR ocular samples generated from StarGANv2 given VIS ocular images and Figure 8.16 shows NIR iris region images generated from StarGANv2 given VIS iris images.

The MT-CNN outperforms the other methods. The StarGANv2 generated images perform better than the Pix2Pix GAN ID when StarGANv2 is trained on unpaired images, whereas Pix2Pix GAN ID is trained on paired images. However, there is still scope for improvement as the MT-CNN is still performing better than GAN-generated images. The ocular recognition is performing better than iris recognition on this dataset.

8.4.3 WVU Dataset

For the evaluation of the cross-modal setting on the WVU dataset, we again follow the subject-disjoint strategy, where 60% (140) of subjects utilize in training and 40% (94) in testing. We perform experiments in three settings as before: the first setting matches VIS face images with VIS face images (Face-Face), and the second setting matches NIR iris images with NIR iris images

Table 8.8: TMR (%) at 0.1% FMR and EER of all ocular and iris recognition methods on the entire test set of the PolyU dataset. MT-CNN outperforms in both ocular and iris recognition.

Experiments	**Ocular**		**Iris**	
	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)
VeriEye	-	-	56.77	18.19
BicycleGAN	67.19	6.55	3.30	20.56
Pix2Pix GAN ID + MT-CNN	94.46 ± 2.82	1.33 ± 0.66	26.30 ± 9.29	14.24 ± 1.88
StarGANv2 [5] + MT-CNN	98.58 ± 0.41	0.42 ± 0.06	28.24 ± 5.12	8.47 ± 0.44
MT-CNN	**99.25 ± 0.29**	**0.35 ± 0.08**	**83.54 ± 0.93**	**2.77 ± 0.32**

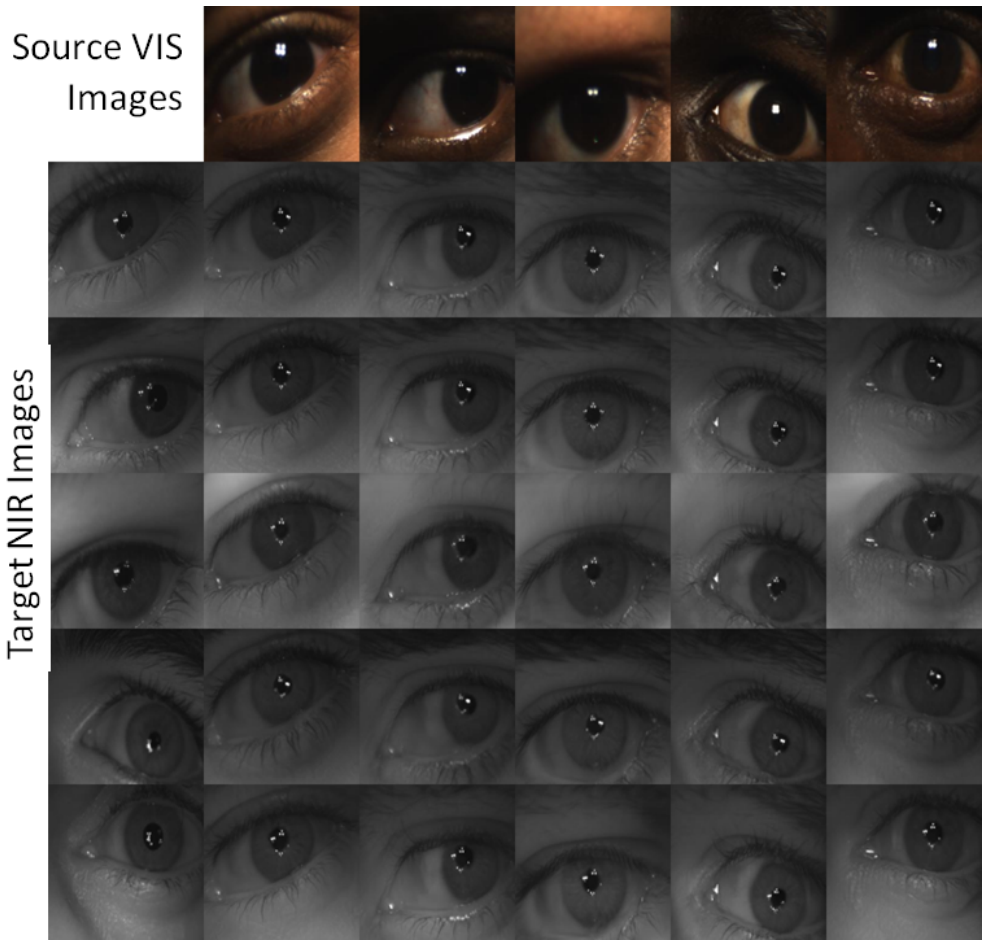


Figure 8.15: StarGANv2 generated ocular images from VIS domain to NIR domain.

(Iris-Iris), and the third setting matches VIS face images with NIR iris images (Iris-Face). Table 8.10 provides the number of genuine and impostor pairs utilized for training and testing in all three

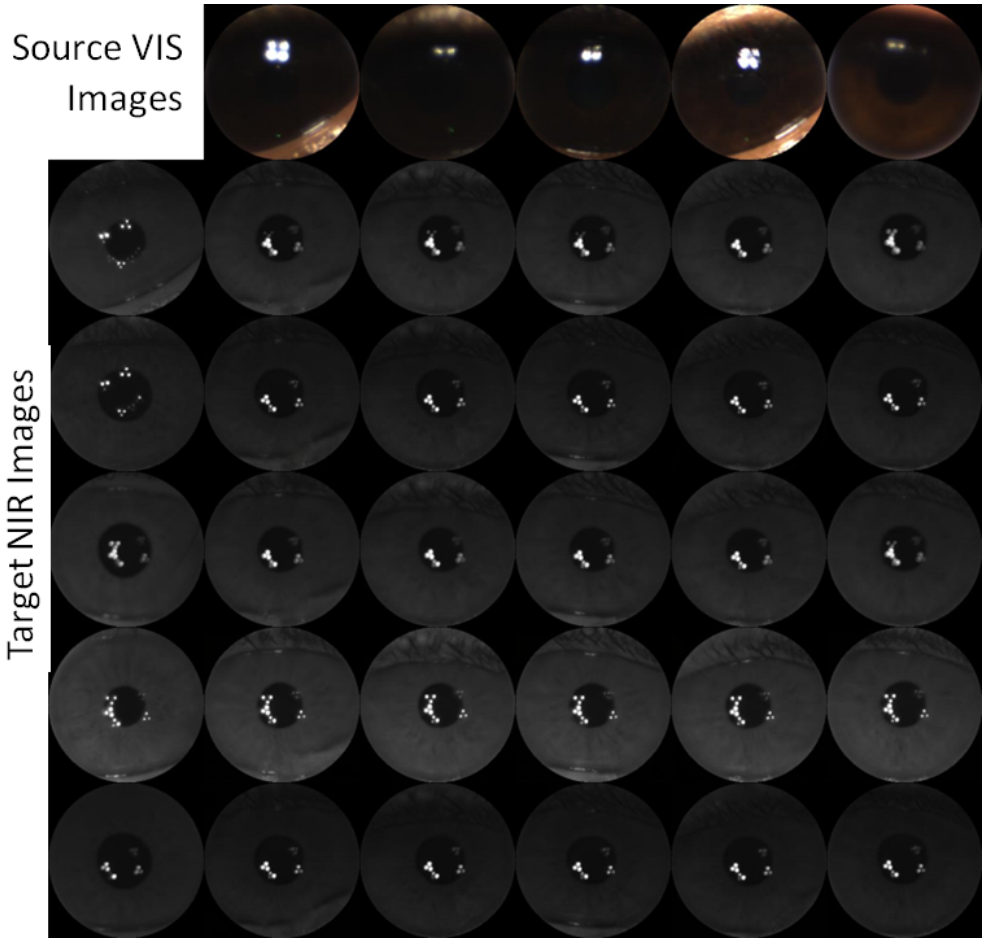


Figure 8.16: StarGANv2 generated iris region images from VIS domain to NIR domain.

settings. For training, we randomly select 50,000 impostor pairs from the entire train set. We perform experiments for both ocular as well as iris recognition, where the input is ocular and iris image, respectively. The evaluation measures used are TMR at 0.1% and EER. To set the maximum recognition performance, we perform face recognition using COTS Rank One Computing (ROC), which produces 99.98% TMR at 0.1% FMR. Table 8.10 provides the ocular and iris recognition results in all three settings. Figure 8.17 shows the Receiver Operating Characteristic (ROC) curves correspond to all methods in the Iris-Face scenario and the histogram corresponds to the MT-CNN.

Ocular recognition achieves the best (68.35% TMR @ 0.1% FMR) on VIS images (intra-modal scenario), and iris recognition achieves the best (92.90% TMR @ 0.1% FMR) on NIR images (intra-modal scenario). Both ocular and iris recognition drop significantly (1.15% TMR @ 0.1% FMR) under a cross-modal scenario, where VIS images (very low-resolution images) match with

Table 8.9: Data distribution among train and test sets for all three settings from the WVU dataset.

	Train Set		Test Set	
	Genuine Pairs	Impostor Pairs	Genuine Pairs	Impostor Pairs
Face-Face	10,164	50,000	9,104	920,890
Iris-Iris	5,668	50,000	4,792	870,882
Face-iris	13,016	50,000	11,296	866,651

Table 8.10: TMRs and EER of ocular and iris recognition techniques on the entire test set of the WVU dataset. All techniques fail on this dataset.

Experiments	**VeriEye**		**USITv3.0**		**MT-CNN(Iris)**		**MT-CNN(Ocular)**	
	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)	TMR (%) @ 0.1% FMR	EER (%)
Face-Face	-	-	0.59	42.47	32.96	11.40	**68.33**	**6.91**
Iris-Iris	**98.95**	**0.79**	96.38	2.15	92.90	3.58	42.94	15.77
Iris-Face	-	-	0.03	48.91	0.88	36.83	**1.07**	**34.12**

NIR images. The failure of the MT-CNN method under the cross-modal scenario is due to the very low resolution of VIS ocular and iris images. Figure 8.18 shows some failure cases when the MT-CNN method is used, and Figure 8.19 shows the t-SNE [280] plot of features extracted from the MT-CNN for the genuine and impostor pairs. There is a large overlap between the features of genuine pairs and impostor pairs, which results in poor performance on the WVU dataset.

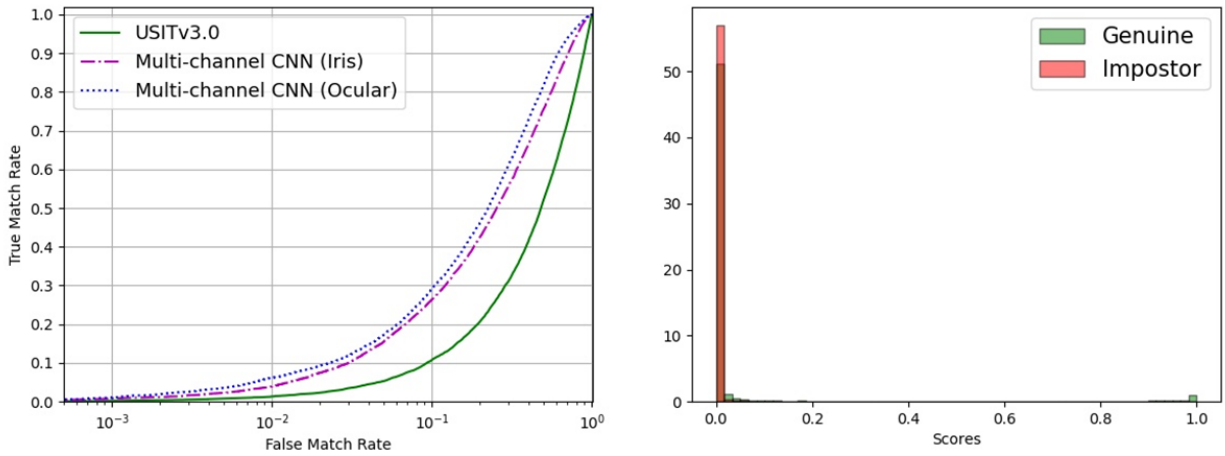


Figure 8.17: ROC curves of iris and ocular recognition techniques and histogram (MT-CNN) on the entire set of the WVU dataset. All techniques fail on this dataset.

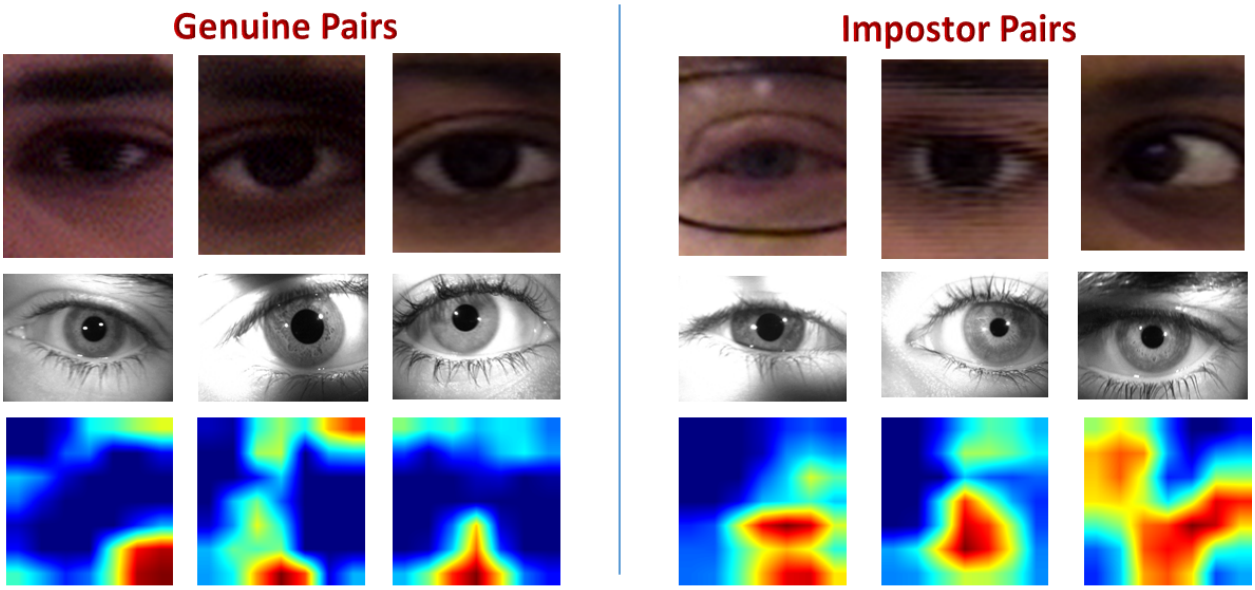


Figure 8.18: Failure cases of the MT-CNN in genuine and impostor pairs. The last row represents the GradCam maps [245] which show regions focused by the network to make the decision. The degraded and very low resolution of ocular images in the VIS spectrum causes the poor performance of cross-modal matching on the WVU dataset.

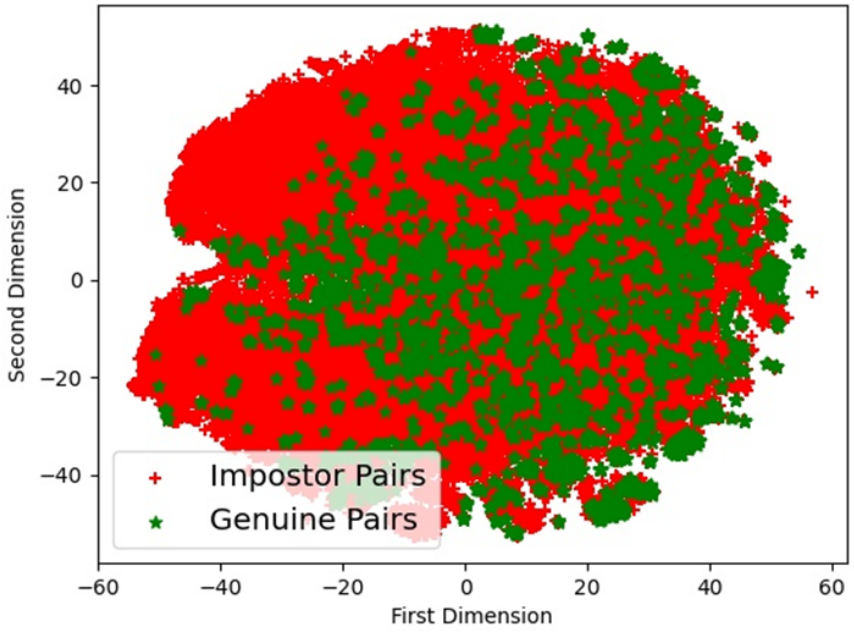


Figure 8.19: t-SNE [280] plot of genuine and impostor pairs features obtained from the MT-CNN network. There is a large overlap between the features of the two distributions. The overlapping criteria could be used to identify on which dataset cross-modal matching would be feasible.

Table 8.11: Iris color distribution of genuine scores obtained from Multi-channel CNN in three settings: face-face, iris-iris and face-iris matching. The region used for the matching is ocular region.

Eye Colors	Blue	Gray	Green	Hazel	Brown	Black	Total
Face-Face	170	8	80	64	296	60	678
Iris-Iris	94	7	42	48	178	39	408
Face-Iris	284	8	163	121	429	85	1090

8.5 Impact of Eye Color on Cross-model Matching

We analyze the influence of eye color on the ocular matching scores. The method used is MT-CNN, and the dataset is the BioCop2008 dataset. Eye color is the result of the amount of melanin present in the iris. Dark-colored irides contain more melanin than light-colored irides. The presence of a high concentration of melanin in the iris absorbs most of the light, causing dark-colored irides. The lack of melanin causes scattering of the light, resulting in the light-colored irides. According to the melanin pigment concentration, eye colors from light to dark can be sequenced as blue, gray, green, hazel, brown, and black. These irides colors can be categorized into broader categories – light-colored irides (blue, gray, green) and dark-colored irides (hazel, brown, and black). The BioCop2008 dataset provides the eye color information for each subject. We use the MT-CNN ocular matching techniques to analyze the effect of eye colors on matching scores.

We used three multi-channel CNNs models trained matching Face (VIS)-Face (VIS), Iris (NIR)-Iris (NIR), and for Face (VIS)-Iris (NIR). Genuine scores from all three models are used to analyze the eye color effect on the matching scores. Table 8.11 provides number of color distribution of genuine scores obtained in all three-matching scenario. Total number of genuine scores in each scenario is also provided. Figure 8.20a, 8.20b, and 8.21 is showing histogram of genuine scores corresponding to different eye colors under three scenarios (face-face, Iris-Iris, Iris-Face) respectively. Figure 8.20a (face-face matching) shows most of the genuine scores going below the threshold (0.71) belong to light colored irides (blue, gray and green). Though no such pattern is noticeable in the Iris-Iris and Face-Iris scenario.

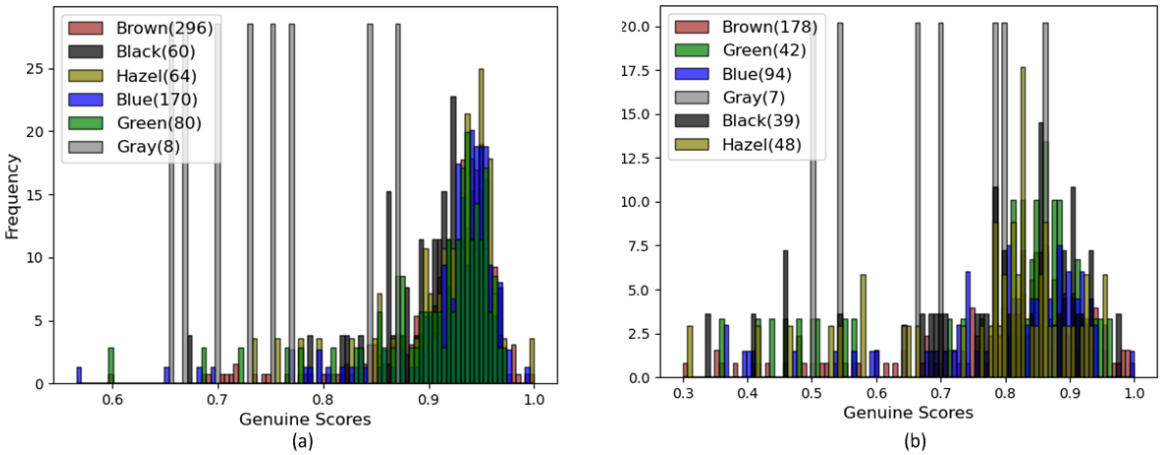


Figure 8.20: (a) Histogram of genuine scores when ocular region from two face images (VIS) are matched. The threshold is 0.71 at 0.2% FMR. (b) Histogram of genuine scores when ocular region from two iris images (NIR) are matched. The threshold is 0.61 at 0.2% FMR.

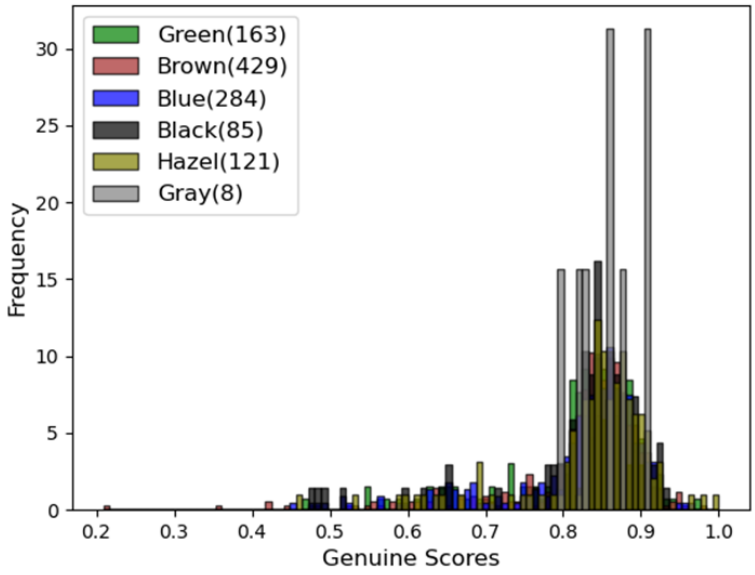


Figure 8.21: Histogram of genuine scores when ocular region from the face (VIS) and iris images (NIR) are matched. The threshold is 0.79 at 0.2% FMR. All techniques fail on this dataset.

8.6 Summary

There are two main challenges when face images match against the iris images (cross-modal recognition): large domain gap and imbalanced training data. We address these challenges with

three deep learning approaches at feature-level, image-level, and training-level. For the first approach, we use Multi-channel CNN, for the second we use three GAN-based architectures (BicycleGAN, Pix2Pix GAN ID, and StarGANv2), and for the third approach, we utilize Dual variation generation (DVG). The first two approaches (feature-level and image-level) aim to reduce the domain gap, whereas the third approach addresses the imbalanced training data issue. Superior results on a cross-modal (BioCop-2008, BioCop-2009, and WVU) and a cross-spectrum (PolyU dataset) datasets show their effectiveness. We further analyze the impact of eye color on the performance of cross-modal performance and it is found that eye color does not impact the performance of cross-model matching.

CHAPTER 9

CONCLUSION

9.1 Research Contributions

Iris recognition has been widely used in a number of large-scale or high-security real-world applications. In this thesis, we focus on some aspects of iris biometrics. Our primary focus is to provide countermeasures against two adversary attacks: presentation attacks and morph attacks. The second focus is on cross-modal matching of NIR iris images with RGB face images.

The first adversary attack we attempt to counteract is presentation attacks (PA) which occur when an adversary presents fake or altered biometric samples to the iris sensor in order to circumvent the biometric system. We propose three iris PA detection methodologies based on the input signal available to facilitate PA detection. The first method called D-NetPAD utilizes a near-infrared iris image for iris PA detection. The method is based on a densely connected convolutional network, which effectively characterizes the bonafide iris pattern to deflect iris PAs. It generalizes well across unseen attacks, sensors, and datasets. It emerges as the best performer in the Intelligence Advanced Research Projects Activity (IARPA) Odin program, LivDet-Iris-2017, and LivDet-Iris-2020 competitions. The second method we proposed utilizes additional hardware (webcam) to capture short videos (~4 secs) depicting the human behavior during their interaction with the iris sensor. The last proposed method employs additional hardware, namely, Optical Coherence Tomography (OCT) imaging. The OCT imaging provides a 2D cross-sectional view (internal structure) of an eye. The iris PA detection using OCT works on the principle that low-coherence light passes through the cornea of bonafide eyes, whereas it is partially (cosmetic contact lens) or completely (plastic eye) blocked by iris PAs. The blocking of the light causes voids in the imaged iris region and aids in the detection of iris PAs. Along with these PA detection methods, we also explain their performance using t-SNE scatter plots and GradCAM heatmaps.

We not only strive for the high performance of the iris PA detectors but also assess the robustness

of these models under input image and architectural parameters perturbations. In the case of input image perturbations, we apply various low and high frequencies manipulations, Gaussian noise, and salt & pepper noise to the input images. We observe that D-NetPAD is comparatively robust to these manipulations to the input images. In the case of architectural parameter perturbations, we apply Gaussian noise, weight zeroing, and weight scaling. We observed that the proposed iris PAD is only robust to the weight zeroing manipulations. The robustness analysis is not confined to iris PA detection methods but also can be applied to any deep neural network.

The maintenance of the performance of iris PA detectors in a non-stationary environment is another required factor in the deployment of the iris PAD in real-world applications. We propose a retraining methodology, where we build a new PA detector using new oncoming training data, and make a final decision for a probe sample by a weighted sum of old and new PA detector scores. We assign the weights dynamically for each probe sample using in-domain models (separate from iris PA detectors). Each in-domain model provides information about the membership of a probe sample to the training data.

The second adversary attack we focus on is morph attacks. A morph attack entails the generation of an image (morphed image) that embodies multiple different identities. The morph attack is not been widely analyzed in iris recognition. To the best of our knowledge, we are the first to report the vulnerability of the iris biometric system to morphed attacks at the image level. We develop a landmark-based iris morphing scheme and demonstrate the potential of morphed iris images to attack the systems. We also develop a deep learning-based network to detect the morphed iris images.

The last contribution of the thesis is to improve the performance of human recognition when matching iris images against face images. Such matching of different modalities is called cross-modal recognition. There are two main challenges: (i) a large domain gap due to different sensors, spectra, and resolutions, and (ii) an imbalance in the training data. We attempt to resolve these challenges with three deep learning approaches. The first approach is at the feature-level using a Multi-channel CNN. It jointly extracts discriminative features from the images of both modalities

to reduce the domain gap. The second approach is at the image-level, where the image from one domain is transformed into another utilizing various GAN architectures. The third approach is training-based which resolves the imbalance of train data by generating samples of the genuine class using the Dual Variational Generation (DVG) framework.

9.2 Future Work

We identify the following directions that require more attention in the future:

1. We proposed various effective software and hardware-based methods (Chapters 2, 3 and 4) for iris presentation attack (PA) detection. However, there is still scope for performance improvement across datasets (Table 2.4). This would entail focusing on the generalizability of the PA detection solutions by either applying domain transfer techniques or updating the existing PA detector with new training data. But this must be done in a manner so as to minimize the data needed from previously unseen domains. Work is also required to provide generalizability to morph attack detection.

2. We attempted to explain the overall results of iris PA detectors using Grad-CAM, t-sne plots, and frequency analysis. However, explainability must be imparted at the individual image level.

3. During sensitivity analysis of deep neural models against weight perturbations, we found that the weights learned using the stochastic gradient descent algorithm are not optimum. Even setting randomly selected weights to zero improves the performance of the models. The observation indicates that there requires additional strategies in finding optimum weights for the deep neural networks. In our work, we empirically selected high-performing models. In future work, we could attempt to analytically find the direction of optimum weights. Another noteworthy observation is that setting low-magnitude weights to zero improves the performance and reduces the model size. So, leveraging the sensitivity analysis, we could work in the direction of model compression or quantization.

4. In the retraining work, we are introducing two new models with every incoming data (or new task), which results in a linear increase in the number of models with an increase in tasks. This raises concern about the scalability of the proposed method. In future work, we could improve the scalability of the method by applying pre-conditions (performance difference or data distribution difference with already available models) before building additional models.

5. In cross-modal matching of iris images with face images, we observed the high-performance of the feature-level method, which involves extraction of common features from both modalities. We also utilized existing generative adversarial networks (GANs) to translate one domain image to another for matching, though the performance is not on par with the feature-level method. GAN-based techniques have shown great potential in various computer vision tasks. Therefore, we could focus on designing a GAN architecture specific to cross-modal matching with effective loss functions.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Biometric e-passport. https://en.wikipedia.org/wiki/Biometric_passport.

[2] IARPA, ODNI:IARPA-BAA-16-04 (Thor). https://www.iarpa.gov/index.php/research-programs/odin/odin-baa.

[3] ISO/IEC 30107-1:2016: Information technology – Biometric Presentation Attack Detection – Part 1: Framework. https://www.iso.org/standard/53227.html.

[4] NICE.I-Noisy Iris Challenge Evaluation Part I. http://nice1.di.ubi.pt/index.html.

[5] THORLabs Telesto series (TEL1325LV2) Spectral domain OCT scanner. https://www.thorlabs.com/catalogpages/Obsolete/2017/TEL1325LV2-BU.pdf.

[6] Unique Identification Authority of India: Govt. of India. Aadhaar Dashboard. https://uidai.gov.in/aadhaar_dashboard/.

[7] Warsaw University of Technology, Poland. http://zbum.ia.pw.edu.pl/EN/node/46.

[8] Andrea F. Abate, Maria Frucci, Chiara Galdi, and Daniel Riccio. BIRD: Watershed based iris detection for mobile devices. *Pattern Recognition Letters (PRL)*, 57:43–51, 2015. Mobile Iris CHallenge Evaluation part I (MICHE I).

[9] R. Abdal, Y. Qin, and P. Wonka. Image2StyleGAN: How to embed images into the StyleGAN latent space? *International Conference on Computer Vision (ICCV)*, pages 4431–4440, 2019.

[10] Mohammed A. M. Abdullah, Satnam S. Dlay, Wai L. Woo, and Jonathon A. Chambers. A novel framework for cross-spectral iris matching. *IPSJ Transactions on Computer Vision and Applications*, 8, 2016.

[11] Cristhian A. Aguilera, Francisco J. Aguilera, Angel D. Sappa, Cristhian Aguilera, and Ricardo Toledo. Learning cross-spectral similarity measures with deep convolutional neural networks. *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, page 9, 2016.

[12] Fares S. Al-Qunaieer and Lahouari Ghouti. Color iris recognition using hypercomplex gabor wavelets. *Symposium on Bio-inspired Learning and Intelligent Systems for Security*, pages 18–19, 2009.

[13] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. *European Conference on Computer Vision (ECCV)*, pages 139–154, 2018.

[14] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. *Advances in Neural Information Processing Systems (NeurIPS))*, 32, 2019.

[15] Fernando Alonso-Fernandez, Pedro Tome-Gonzalez, Virginia Ruiz-Albacete, and Javier Ortega-Garcia. Iris recognition based on sift features. *IEEE International Conference on Biometrics, Identity and Security (BIdS)*, pages 1–8, 2009.

[16] A. Anjos, M. M. Chakka, and S. Marcel. Motion-based counter-measures to photo attacks in face recognition. *IET Biometrics*, 3(3):147–158, 2014.

[17] André Anjos and Sébastien Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. *International Joint Conference on Biometrics (IJCB)*, 2011.

[18] S. S. Arora, M. Vatsa, R. Singh, and A. Jain. Iris recognition under alcohol influence: A preliminary study. *IAPR International Conference on Biometrics (ICB)*, pages 336–341, 2012.

[19] M. Arsalan, R. A. Naqvi, D. S. Kim, P. H. Nguyen, M. Owais, and K. R. Park. IrisDenseNet: Robust iris segmentation using densely connected fully convolutional networks in the images by visible light and near-infrared light camera sensors. *Sensors*, 2018.

[20] Muhammad Arsalan, Hyung Gil Hong, Rizwan Ali Naqvi, Min Beom Lee, Min Cheol Kim, Dong Seop Kim, Chan Sik Kim, and Kang Ryoung Park. Deep learning-based iris segmentation for iris recognition in visible light environment. *Symmetry*, 9(11), 2017.

[21] Sarah E. Baker, Amanda Hentz, Kevin W. Bowyer, and Patrick J. Flynn. Degradation of iris recognition performance due to non-cosmetic prescription contact lenses. *Computer Vision and Image Understanding (CVIU)*, 114(9):1030–1044, 2010.

[22] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8218–8227, June 2021.

[23] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, 1996.

[24] Carlos A. C. M. Bastos, Ing Ren Tsang, and George D. C. Calvalcanti. A combined pulling amp; pushing and active contour method for pupil segmentation. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 850–853, 2010.

[25] A. Bastys, J. Kranauskas, and R. Masiulis. Iris recognition by local extremum points of multiscale taylor expansion. *Pattern Recognition (PR)*, 42(9):1869–1877, 2009.

[26] T. Beier and S. Neely. Feature-based image metamorphosis. *SIGGRAPH Computer Graphics*, 26(2):35–42, 1992.

[27] Dalila Benboudjema, Nadia Othman, Bernadette Dorizzi, and Wojciech Pieczynski. Challenging eye segmentation using triplet markov spatial models. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1927–1931, 2013.

[28] Oliver Bergamin, M. Bridget Zimmerman, and Randy Kardon. Pupil light reflex in normal and diseased eyes: diagnosis of visual dysfunction using waveform partitioning. *Ophthalmology*, 110:106–14, 02 2003.

[29] T. Bergmüller, L. Debiasi, A. Uhl, and Z. Sun. Impact of sensor ageing on iris recognition. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–8, 2014.

[30] Rajesh M. Bodade and Sanjay N. Talbar. Shift invariant iris feature extraction using rotated complex wavelet and complex wavelet for iris recognition system. *International Conference on Advances in Pattern Recognition (ICAPR)*, pages 449–452, 2009.

[31] Vishnu Naresh Boddeti, B.V.K. Vijaya Kumar, and Krishnan Ramkumar. Improved iris segmentation based on local texture statistics. *Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 2147–2151, 2011.

[32] W.W. Boles and B. Boashash. A human identification technique using images of the iris and wavelet transform. *IEEE Transactions on Signal Processing*, 46(4):1185–1188, 1998.

[33] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid. OULU-NPU: a mobile face presentation attack database with real-world variations. *IEEE International Conference on Automatic Face and Gesture recognition (FG)*, 2017.

[34] Zinelabidine Boulkenafet, Jukka Komulainen, Zahid Akhtar, Azeddine Benlamoudi, Djamel Samai, Salah Eddine Bekhouche, Abdelkrim Ouafi, Fadi Dornaika, Abdelmalik taleb ahmed, L Qin, F Peng, Le-Bing Zhang, M Long, Shruti Bhilare, V Kanhangad, Artur Costa-Pazo, Esteban Vazquez-Fernandez, D Perez-Cabo, J J. Moreira-Perez, and A Hadid. A competition on generalized software-based face presentation attack detection in mobile scenarios. *International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2017.

[35] K. W. Bowyer, K. P. Hollingsworth, and P. J. Flynn. *A survey of Iris biometrics research: 2008-2010*. Springer, 2013.

[36] Kevin Bowyer, Sarah Baker, Amanda Hentz, Karen Hollingsworth, Tanya Peters, and Patrick Flynn. Factors that degrade the match distribution in iris biometrics. *Identity in the Information Society*, 2:327–343, 12 2009.

[37] Kevin W. Bowyer, Karen Hollingsworth, and Patrick J. Flynn. Image understanding for iris biometrics: A survey. *Computer Vision and Image Understanding (CVIU)*, 110(2):281–307, 2008.

[38] Aidan Boyd, Adam Czajka, and Kevin Bowyer. Deep learning-based feature extraction in iris recognition: Use existing models, fine-tune or train from scratch? *International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–9, 2019.

[39] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: Identifying density-based local outliers. *ACM SIGMOD International Conference on Management of Data*, page 93–104, 2000.

[40] A. Bron, R. Tripathi, and B. Tripathi. *Wolff 's anatomy of the eye and orbit*. 1998.

[41] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. *European Conference on Computer Vision (ECCV)*, pages 25–36, 2004.

[42] Adrian Bulat, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. Incremental multi-domain learning with network latent tensor factorization. *AAAI Conference on Artificial Intelligence*, 34(07):10470–10477, 2020.

[43] Mark J. Burge and Matthew K. Monaco. Multispectral iris fusion for enhancement, interoperability, and cross wavelength matching. *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XV*, 7334:494–501, 2009.

[44] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet Kumar Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. *arXiv*, abs/1902.10486, 2019.

[45] C. Chen and A. Ross. Exploring the use of iriscodes for presentation attack detection. *International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2018.

[46] C. Chen and A. Ross. A multi-task convolutional neural network for joint iris detection and presentation attack detection. *IEEE Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2018.

[47] C. Chen and A. Ross. An explainable attention-guided iris presentation attack detector. *IEEE Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2021.

[48] Jianxu Chen, Feng Shen, Danny Ziyi Chen, and Patrick J. Flynn. Iris recognition based on human-interpretable features. *IEEE Transactions on Information Forensics and Security (TIFS)*, 11(7):1476–1485, 2016.

[49] Rui Chen, Xirong Lin, and Tianhuai Ding. Liveness detection for iris recognition using multispectral images. *Pattern Recognition Letters (PRL)*, 33(12):1513–1519, 2012.

[50] Zhiyuan Chen, Bing Liu, Ronald Brachman, Peter Stone, and Francesca Rossi. *Lifelong Machine Learning*. Morgan Claypool Publishers, 2nd edition, 2018.

[51] Nicholas Cheney, Martin Schrimpf, and Gabriel Kreiman. On the robustness of convolutional neural networks to internal architecture and weight perturbations. *arXiv*, abs/1703.08245, 2017.

[52] Ivana Chingovska, J. Yang, Zhen Lei, Dong Yi, Stan Z. Li, Olga Kähm, Christian Glaser, Naser Damer, Arjan Kuijper, Alexander Nouak, Jukka Komulainen, Tiago Freitas Pereira, Shubham Gupta, Shubham Khandelwal, Shubham Bansal, Ayush Rai, Tarun Krishna, Dushyant Goyal, Muhammad-Adeel Waris, Honglei Zhang, Iftikhar Ahmad, Serkan Kiranyaz, Moncef Gabbouj, Roberto Tronci, Maurizio Pili, Nicola Sirena, Fabio Roli, Javier Galbally, Julian Fiérrez, Allan da Silva Pinto, Hélio Pedrini, W. S. Schwartz, Anderson Rocha, André Anjos, and Sébastien Marcel. The 2nd competition on counter measures to 2D face spoofing attacks. *International Conference on Biometrics (ICB)*, pages 1–6, 2013.

[53] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. StarGAN v2: Diverse image synthesis for multiple domains. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[54] Jonathan Connell, Nalini Ratha, James Gentile, and Ruud Bolle. Fake iris detection using structured light. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8692–8696, 2013.

[55] S. Crihalmeanu, A. Ross, S. Schuckers, and L. Hornak. A protocol for multibiometric data acquisition, storage and dissemination. *Technical Report, WVU, Lane Department of Computer Science and Electrical Engineering*, 2007.

[56] A. Czajka. Pupil dynamics for iris liveness detection. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(4):726–735, 2015.

[57] Adam Czajka. Iris liveness detection by modeling dynamic pupil features. In Mark J. Burge and Kevin W. Bowyer, editors, *Handbook of Iris Recognition*, volume 1542, pages 439–467. Springer-Verlag, 2013.

[58] Adam Czajka and Kevin W. Bowyer. Presentation attack detection for iris recognition: An assessment of the state-of-the-art. *ACM Computing Surveys (CSUR)*, 51(4):86:1–86:35, 2018.

[59] Adam Czajka, Daniel Moreira, Kevin Bowyer, and Patrick Flynn. Domain-specific human-inspired binarized statistical image features for iris recognition. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 959–967, 2019.

[60] N. Damer, A. M. Saladié, A. Braun, and A. Kuijper. MorGAN: Recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network. *International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–10, 2018.

[61] P. Das, J. McGrath, A. Boyd Z. Fang, G. Jang, A. Mohammadi, S. Purnapatra, D. Yambay, S. Marcel, M. Trokielewicz, P. Maciejewicz, K. Bowyer, A. Czajka, S. Schuckers, J. Tapia,

S. Gonzalez, M. Fang, N. Damer, F. Boutros, A. Kuijper, R. Sharma, C. Chen, and A. Ross. Iris liveness detection competition (LivDet-Iris) – the 2020 edition. *International Joint Conference on Biometrics (IJCB)*, 2020.

[62] J. Daugman. How iris recognition works. *Transactions on Circuits and Systems for Video Technology (TCSVT)*, 14(1), 2004.

[63] J Daugman and C Downing. Epigenetic randomness, complexity and singularity of human iris patterns. *Proceedings of the Royal Society B: Biological Sciences (Proc Biol Sci)*, 268:1737–40, 2001.

[64] John Daugman. Countermeasures against subterfuge. *Biometrics: Personal Identification in Networked Society*, pages 103–121, 1999.

[65] John Daugman. Demodulation by complex-valued wavelets for stochastic pattern recognition. *International Journal of Wavelets, Multi-resolution and Information Processing*, 1:1–17, 2003.

[66] John Daugman. Recognizing persons by their iris patterns. *Advances in Biometric Person Authentication*, 3338:5–25, 2004.

[67] John Daugman. New methods in iris recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(5):1167–1175, 2007.

[68] John Daugman. Collision avoidance on national and global scales: Understanding and using big biometric entropy. *TechRxiv*, Feb 2021.

[69] John G. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 15(11), 1993.

[70] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. Continual learning: A comparative study on how to defy forgetting in classification tasks. *arXiv*, abs/1909.08383(6), 2019.

[71] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pages 1–1, 2021.

[72] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: a large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[73] Jia Deng, Wei Dong, Richard Socher, Li jia Li, Kai Li, and Li Fei-fei. ImageNet: a large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[74] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, et al. Recent advances in deep learning for speech research at microsoft. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8604–8608, 2013.

[75] Li Deng and Yang Liu. *Deep learning in natural language processing*. Springer, Singapore, 2018.

[76] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, 39(4):677–691, 2017.

[77] Ruggero Donida Labati and Fabio Scotti. Noisy iris segmentation with boundary regularization and reflections removal. *Image and Vision Computing (IVC)*, 28(2):270–277, 2010.

[78] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*, 2021.

[79] James S. Doyle and Kevin W. Bowyer. Robust detection of textured contact lenses in iris recognition using BSIF. *IEEE Access*, 3:1672–1683, 2015.

[80] Y. Du, E. Arslanturk, Z. Zhou, and C. Belcher. Video-based noncooperative iris image segmentation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 41(1):64–74, 2011.

[81] Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. *European Conference on Computer Vision (ECCV)*, pages 386–402, 2020.

[82] Gizem Erdogan. *Contact Lenses in Iris Recognition*. M.S. dissertation, graduate theses, dissertations, and problem reports, number 4965, West Virginia University, 2013.

[83] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Robustness of classifiers: From adversarial to random noise. *International Conference on Neural Information Processing Systems (NeurIPS)*, page 1632–1640, 2016.

[84] S. P. Fenker, E. Ortiz, and K. W. Bowyer. Template aging phenomenon in iris recognition. *IEEE Access*, 1:266–274, 2013.

[85] M. Ferrara, R. Cappelli, and D. Maltoni. On the feasibility of creating double-identity fingerprints. *IEEE Transactions on Information Forensics and Security (TIFS)*, 12(4):892–900, 2017.

[86] M. Ferrara, A. Franco, and D. Maltoni. The magic passport. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–7, 2014.

[87] Maria Frucci, Michele Nappi, Daniel Riccio, and Gabriella Sanniti di Baja. WIRE: watershed based iris recognition. *Pattern Recognition (PR)*, 52:148–159, 2016.

[88] C. Fu, X. Wu, Y. Hu, H. Huang, and R. He. Dual variational generation for low-shot heterogeneous face recognition. *Neural Information Processing Systems (NIPS)*, 2019.

[89] J. Galbally, S. Marcel, and J. Fierrez. Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition. *IEEE Transactions on Image Processing (TIP)*, 23(2):710–724, 2014.

[90] Javier Galbally, Arun Ross, Marta Gomez-Barrero, Julian Fierrez, and Javier Ortega-Garcia. Iris image reconstruction from binary templates: An efficient probabilistic approach based on genetic algorithms. *Computer Vision and Image Understanding (CVIU)*, 117(10):1512–1525, 2013.

[91] A. Gangwar and A. Joshi. DeepIrisNet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition. *IEEE International Conference on Image Processing (ICIP)*, pages 2301–2305, 2016.

[92] A. Gangwar, Akanksha Joshi, Padmaja Joshi, and Ramachandra Raghavendra. DeepIrisNet2: learning deep-iriscodes from scratch for segmentation-robust visible wavelength and near infrared iris recognition. *arXiv*, abs/1902.05390, 2019.

[93] Prachi Garg, Rohit Saluja, Vineeth N Balasubramanian, Chetan Arora, Anbumani Subramanian, and C. V. Jawahar. Multi-domain incremental learning for semantic segmentation. *arXiv*, abs/2110.12205, 2021.

[94] Siddhant Garg, Adarsh Kumar, Vibhor Goel, and Yingyu Liang. Can adversarial weight perturbations inject neural backdoors. *ACM International Conference on Information and Knowledge Management (CIKM)*, page 2029–2032, 2020.

[95] M. Gomez-Barrero, C. Rathgeb, U. Scherhag, and C. Busch. Predicting the vulnerability of biometric systems to attacks based on morphed biometric information. *IET Biometrics*, 7(4):333–341, 2018.

[96] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations (ICLR)*, 2015.

[97] Diego Gragnaniello, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. An investigation of local descriptors for biometric spoofing detection. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(4):849–863, 2015.

[98] Diego Gragnaniello, Carlo Sansone, and Luisa Verdoliva. Iris liveness detection for mobile devices based on local descriptors. *Pattern Recognition Letters (PRL)*, 57:81–87, 2015.

[99] P. Grother, J. R. Matey, E. Tabassi, G. W. Quinn, and M. Chumakov. IREX VI: temporal stability of iris recognition accuracy. NIST Interagency Report 7948, 2013.

[100] P. Grother, G. W. Quinn, J. R. Matey, M. L. Ngan, W. J. Salamon, G. P. Fiumara, and C. I. Watson. IREX III - Performance of iris identification algorithms. NIST Interagency/Internal Report (NISTIR) 7836, 2012.

[101] Murthy Gudlavalleti, Sanjeev Gupta, Neena John, and Praveen Vashist. Current status of cataract blindness and vision 2020: The right to sight initiative in india. *Indian Journal of Ophthalmology (IJO)*, 56:489–94, 05 2008.

[102] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks. *International Conference on Neural Information Processing Systems (NeurIPS)*, page 1135–1143, 2015.

[103] M. Happold. Structured forest edge detectors for improved eyelid and iris segmentation. *International Conference of the Biometrics Special Interest Group (CCPR)*, page 28–33, 2015.

[104] Bilal Hassan, Ramsha Ahmed, Taimur Hassan, and Naoufel Werghi. SIP-SegNet: a deep convolutional encoder-decoder network for joint semantic segmentation and extraction of sclera, iris and pupil based on periocular region suppression. *arXiv*, abs/2003.00825, 2020.

[105] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[106] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[107] R. He, X. Wu, Z. Sun, and T. Tan. Wasserstein CNN: Learning invariant features for NIR-VIS face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 41(07), 2019.

[108] X. He, Y. Lu, and P. Shi. A fake iris detection method based on FFT and quality assessment. *Chinese Conference on Pattern Recognition (CCPR)*, pages 1–4, 2008.

[109] Zhaofeng He, Zhenan Sun, Tieniu Tan, and Zhuoshi Wei. Efficient iris spoof detection via boosted local binary patterns. *International Conference on Biometrics (ICB)*, 5558:1080–1090, 2009.

[110] Kevin Hernandez-Diaz, Fernando Alonso-Fernandez, and Josef Bigun. Cross-spectral periocular recognition with conditional adversarial networks. *International Joint Conference on Biometrics (IJCB)*, 2020.

[111] H. Hofbauer, I. Tomeo-Reyes, and A. Uhl. Isolating iris template ageing in a semi-controlled environment. *International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5, 2016.

[112] Heinz Hofbauer, Ehsaneddin Jalilian, and Andreas Uhl. Exploiting superior CNN-based iris segmentation for better recognition accuracy. *Pattern Recognition Letters (PRL)*, 120:17–23, 2019.

[113] Steven Hoffman, Renu Sharma, and Arun Ross. Convolutional neural networks for iris presentation attack detection: Toward cross-dataset and cross-sensor generalization. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1701–17018, 2018.

[114] Steven Hoffman, Renu Sharma, and Arun Ross. Iris + ocular: Generalized iris presentation attack detection using multiple convolutional neural networks. *International Conference on Biometrics (ICB)*, 2019.

[115] Karen Hollingsworth, Kevin Bowyer, and Patrick Flynn. Pupil dilation degrades iris biometric performance. *Computer Vision and Image Understanding (CVIU)*, 113:150–157, 01 2009.

[116] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *Association for Computational Linguistics (ACL)*, 2018.

[117] Sheng-Hsun Hsieh, Yunghui Li, Wei Wang, and Chung-Hao Tien. A novel anti-spoofing solution for iris recognition toward cosmetic contact lens attack using spectral ICA analysis. *Sensors*, 18:795–810, 2018.

[118] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. *arXiv*, abs/1810.12488, 2019.

[119] Yang Hu, Konstantinos Sirlantzis, and Gareth Howells. Iris liveness detection using regional features. *Pattern Recognition Letters (PRL)*, 82:242–250, 2016.

[120] D. Huang, E. A. Swanson, C. P. Lin, J. S. Schuman, W. G. Stinson, W. Chang, M. R. Hee, T. Flotte, K. Gregory, C. A. Puliafito, and et al. Optical coherence tomography. *Science*, 254:1178–1181, 1991.

[121] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger. Densely connected convolutional networks. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[122] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[123] K. Hughes and K. W. Bowyer. Detection of contact-lens-based iris biometric spoofs using stereo imaging. *Hawaii International Conference on System Sciences (HICSS)*, 2013.

[124] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning (ICML)*, 37:448–456, 2015.

[125] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017.

[126] A. K. Jain, A. A. Ross, and K. Nandakumar. *Introduction to Biometrics*. Springer Publishing Company, 2011.

[127] Ann A. Jarjes, Kuanquan Wang, and Ghassan J. Mohammed. Iris localization: Detecting accurate pupil contour and localizing limbus boundary. *2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010)*, 1:349–352, 2010.

[128] Dae Sik Jeong, Jae Won Hwang, Byung Jun Kang, Kang Ryoung Park, Chee Sun Won, Dong-Kwon Park, and Jaihie Kim. A new iris segmentation method for non-ideal iris images. *Image and Vision Computing (IVC)*, 28(2):254–260, 2010.

[129] R. Jillela and A. Ross. Matching face against iris images using periocular information. *IEEE International Conference on Image Processing (ICIP)*, pages 4997–5001, 2014.

[130] Raghavender R. Jillela and A. Ross. *Methods for Iris Segmentation*. Springer London, 2013.

[131] Liu Jin, Fu Xiao, and Wang Haopeng. Iris image segmentation based on k-means cluster. *IEEE International Conference on Intelligent Computing and Intelligent Systems*, 3:194–198, 2010.

[132] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *European Conference on Computer Vision (ECCV)*, 2016.

[133] Amin Jourabloo, Yaojie Liu, and Xiaoming Liu. Face de-spoofing: Anti-spoofing via noise modeling. *European Conference on Computer Vision (ECCV)*, 2018.

[134] Roy K. and Bhattacharya P. Iris recognition in nonideal situations. *International Conference on Information Security (ISC)*, 5735, 2009.

[135] Miwa Kanematsu, Hironobu Takano, and Kiyomi Nakamura. Highly reliable liveness detection method for iris recognition. *SICE Annual Conference*, pages 361–364, 2007.

[136] Ta-Chu Kao, Kristopher Jensen, Gido van de Ven, Alberto Bernacchia, and Guillaume Hennequin. Natural continual learning: success is a journey, not (just) a destination. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:28067–28079, 2021.

[137] Nikolaos Karianakis, Jingming Dong, and Stefano Soatto. An empirical evaluation of current convolutional architectures' ability to manage nuisance location and scale variability. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4442–4451, 2016.

[138] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.

[139] R. Kerekes, B. Narayanaswamy, J. Thornton, M. Savvides, and B. V. K. Vijaya Kumar. Graphical model approach to iris matching under deformation and occlusion. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–6, 2007.

[140] Younghwan Kim, Jang-Hee Yoo, and Kyoungho Choi. A motion and similarity-based fake detection method for biometric face recognition systems. *IEEE Transactions on Consumer Electronics (TCE)*, 57, 2011.

[141] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017.

[142] Maki Kojima, Toshiki Shioiri, Toshihiro Hosoki, Hideaki Kitamura, Takehiko Bando, and Toshiyuki Someya. Pupillary light reflex in panic disorder: A trial using audiovisual stimulation. *European Archives of Psychiatry and Clinical Neuroscience (EAPCN)*, 254:242–4, 09 2004.

[143] O. V. Komogortsev, A. Karpov, and C. D. Holland. Attack of mechanical replicas: Liveness detection with eye movements. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(4):716–725, 2015.

[144] Oleg Komogortsev and Alex Karpov. Liveness detection via oculomotor plant characteristics: Attack of mechanical replicas. *International Conference on Biometrics (ICB)*, pages 1–8, 2013.

[145] Jukka Komulainen, Abdenour Hadid, and Matti Pietikäinen. Context based face anti-spoofing. *International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–8, 2013.

[146] Jukka Komulainen, Abdenour Hadid, Matti Pietikainen, André Anjos, and Sébastien Marcel. Complementary countermeasures for detecting scenic face spoofing attacks. *International Conference on Biometrics (ICB)*, 2013.

[147] Emine Krichen. Lef3a: Pupil segmentation using viterbi search algorithm. *IAPR International Conference on Biometrics (ICB)*, pages 323–329, 2012.

[148] A. Kumar, T.-S. Chan, and C.-W. Tan. Human identification from at-a-distance face images using sparse representation of local iris features. *IAPR International Conference on Biometrics (ICB)*, pages 303–309, 2012.

[149] Ajay Kumar and Tak-Shing Chan. Iris recognition using quaternionic sparse orientation code (QSOC). *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 59–64, 2012.

[150] Ajay Kumar and Arun Passi. Comparison and combination of iris matchers for reliable personal authentication. *Pattern Recognition (PR)*, 43(3):1016 – 1026, 2010.

[151] L. Ma, T. Tan, Y. Wang, and D. Zhang. Efficient iris recognition by characterizing key local variations. *Transactions on Image Processing (TIP)*, 13(6):739–750, 2004.

[152] S. J. Lee, K. R. Park, and J. Kim. Robust fake iris detection based on variation of the reflectance ratio between the iris and the sclera. *Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pages 1–6, 2006.

[153] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. *International Conference on Learning Representations (ICLR)*, 2020.

[154] Sung Lee, Kang Park, Youn Lee, Kwanghyuk Bae, and Jai Kim. Multifeature-based fake iris detection method. *Optical Engineering*, 46(12), 2007.

[155] Timothée Lesort, Massimo Caccia, and Irina Rish. Understanding continual learning settings with data distribution drift analysis. *arXiv*, abs/2104.01678, 2021.

[156] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. *AAAI Conference on Artificial Intelligence*, 2018.

[157] Haiqing Li, Zhenan Sun, and Tieniu Tan. Robust iris segmentation based on learned boundary detectors. *IAPR International Conference on Biometrics (ICB)*, pages 317–322, 2012.

[158] X. Li. Modeling intra-class variation for nonideal iris recognition. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3832 LNCS:419–427, 2006.

[159] Y. Li and M. Savvides. An automatic iris occlusion estimation method based on high-dimensional density estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(4):784–796, 2013.

[160] Yung-Hui Li, Po-Jen Huang, and Yun Juan. An efficient and robust iris segmentation algorithm using deep learning. *Mobile Information Systems*, 2019.

[161] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 40(12):2935–2947, 2017.

[162] Jie Lin, Jian-Ping Li, Hui Lin, and Ji Ming. Robust person identification with face and iris by modified PUM method. *International Conference on Apperceiving Computing and Intelligence Analysis (ICACIA)*, pages 321–324, 2009.

[163] Christoph Lippert, Riccardo Sabatini, M. Cyrus Maher, Eun Yong Kang, Seunghak Lee, Okan Arikan, Alena Harley, Axel Bernal, Peter Garst, Victor Lavrenko, Ken Yocum, Theodore Wong, Mingfu Zhu, Wen-Yun Yang, Chris Chang, Tim Lu, Charlie W. H. Lee, Barry Hicks, Smriti Ramakrishnan, Haibao Tang, Chao Xie, Jason Piper, Suzanne Brewerton, Yaron Turpaz, Amalio Telenti, Rhonda K. Roby, Franz J. Och, and J. Craig Venter. Identification of individuals by trait prediction using whole-genome sequencing data. *Proceedings of the National Academy of Sciences (PNAS)*, 114(38):10166–10171, 2017.

[164] Nianfeng Liu, Man Zhang, Haiqing Li, Zhenan Sun, and Tieniu Tan. DeepIris: learning pairwise filter bank for heterogeneous iris verification. *Pattern Recognition Letters (PRL)*, 82:154–161, 2016.

[165] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[166] Yaojie Liu, Joel Stehouwer, Amin Jourabloo, and Xiaoming Liu. Deep tree learning for zero-shot face anti-spoofing. *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[167] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[168] Kamachi M., Hill H.and Lander K., and Vatikiotis-Bateson E. Putting the face to the voice: matching identity across modality. *Current Biology*, 13(19), 2003.

[169] Li Ma, Tieniu Tan, Yunhong Wang, and Dexin Zhang. Local intensity variation analysis for iris recognition. *Pattern Recognition (PR)*, 37(6):1287–1298, 2004.

[170] Neil A. Macmillan and C. Douglas Creelman. Detection theory: A user's guide. *Lawrence Erlbaum Associates*.

[171] A. Makrushin, T. Neubert, and J. Dittmann. Automatic generation and detection of visually faultless facial morphs. *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, pages 39–50, 2017.

[172] Sébastien Marcel, Mark S. Nixon, Julian Fiérrez, and Nicholas W. D. Evans, editors. *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection, Second Edition*. Advances in Computer Vision and Pattern Recognition. Springer, 2019.

[173] Libor Masek. Recognition of human iris patterns for biometric identification. Technical report, 2003.

[174] Carver Mead and Mohammed Ismail, editors. *Analog VLSI Implementation of Neural Systems*. The Kluwer International Series in Engineering and Computer Science. Kluwer / Springer US, 1989.

[175] Hunny Mehrotra, Banshidhar Majhi, and Phalguni Gupta. Annular iris recognition using SURF. *Pattern Recognition and Machine Intelligence (PAMI)*, pages 464–469, 2009.

[176] David Menotti, Giovani Chiachia, Allan Pinto, William Robson Schwartz, Helio Pedrini, Alexandre Xavier Falcao, and Anderson Rocha. Deep Representations for Iris, Face, and Fingerprint Spoofing Detection. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(4):864–879, 2015.

[177] 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 (PAMI)*, 30(10):1741–1756, 2008.

[178] Murali Mohan Chakka, André Anjos, Sébastien Marcel, Roberto Tronci, Daniele Muntoni, Gianluca Fadda, Maurizio Pili, Nicola Sirena, Gabriele Murgia, Marco Ristori, Fabio Roli, Junjie Yan, Dong Yi, Zhen Lei, Zhiwei Zhang, Stan Li, William Schwartz, Anderson Rocha, Helio Pedrini, and Matti Pietikainen. Competition on counter measures to 2-D facial spoofing attacks. *International Joint Conference on Biometrics (IJCB)*, pages 1 – 6, 2011.

[179] D. M. Monro, S. Rakshit, and D. Zhang. DCT-based iris recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(4):586–595, 2007.

[180] Y. Moolla, L. Darlow, A. Sharma, A. Singh, and J. Van Der Merwe. Optical coherence tomography for fingerprint presentation attack detection. *Handbook of Biometric Anti-Spoofing*, pages 49–70, 2019.

[181] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[182] Jose G. Moreno-Torres, Troy Raeder, RocÃo Alaiz-RodrÃguez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern Recognition (PR)*, 45(1):521–530, 2012.

[183] Satish Mulleti and Chandra Sekhar Seelamantula. Ellipse fitting using the finite rate of innovation sampling principle. *IEEE Transactions on Image Processing (TIP)*, 25(3):1451–1464, 2016.

[184] Tajbakhsh N., Misaghian K., and Bandari N.M. A region-based iris feature extraction method based on 2d-wavelet transform. *Biometric ID Management and Multimodal Communication (BioID)*, 5707, 2009.

[185] A. Nagrani, S. Albanie, and A. Zisserman. Seeing voices and hearing faces: Cross-modal biometric matching. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8427–8436, 2018.

[186] P. R. Nalla and A. Kumar. Toward more accurate iris recognition using cross-spectral matching. *IEEE Transactions on Image Processing (TIP)*, 26(1):208–221, 2017.

[187] K. Nguyen, C. Fookes, and S. Sridharan. Fusing shrinking and expanding active contour models for robust iris segmentation. *International Conference on Information Sciences, Signal Processing and their Applications (ISSPA)*, pages 185–188, 2010.

[188] Kien Nguyen, Clinton Fookes, Raghavender Jillela, Sridha Sridharan, and Arun Ross. Long range iris recognition: A survey. *Pattern Recognition (PR)*, 72:123–143, 2017.

[189] Kien Nguyen, Clinton Fookes, Arun Ross, and Sridha Sridharan. Iris recognition with off-the-shelf CNN features: A deep learning perspective. *IEEE Access*, 6:18848–18855, 2018.

[190] Ishan Nigam, Mayank Vatsa, and Richa Singh. *Ophthalmic Disorder Menagerie and Iris Recognition*, pages 359–396. Springer London, 2016.

[191] Roman Novak, Yasaman Bahri, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. *International Conference on Learning Representations (ICLR)*, 2018.

[192] Maulisa Oktiana, Takahiko Horiuchi, Keita Hirai, Khairun Saddami, Fitri Arnia, Yuwaldi Away, and Khairul Munadi. Cross-spectral iris recognition using phase-based matching and homomorphic filtering. *Heliyon*, 6(2):e03407, 2020.

[193] Andrzej Pacut and Adam Czajka. Aliveness detection for iris biometrics. *IEEE International Carnahan Conferences Security Technology (ICCST)*, pages 122 – 129, 2006.

[194] Federico Pala and Bir Bhanu. Iris liveness detection by relative distance comparisons. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 664–671, 2017.

[195] Gang Pan, Lin Sun, Zhaohui Wu, and Yueming Wang. Monocular camera-based face liveness detection by combining eyeblink and scene context. *Telecommunication Systems*, 47(3):215–225, 2011.

[196] Lili Pan, Mei Xie, Tao Zheng, and Jianli Ren. A robust iris localization model based on phase congruency and least trimmed squares estimation. *Image Analysis and Processing (ICIAP)*, 2009.

[197] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 22(10):1345–1359, 2010.

[198] Jong Hyun Park and Moon-Gi Kang. Multispectral iris authentication system against counterfeit attack using gradient-based image fusion. *Optical Engineering*, 46(11):1–14, 2007.

[199] Keyurkumar Patel, Hu Han, and Anil K. Jain. Cross-database face antispoofing with robust feature representation. In Zhisheng You, Jie Zhou, Yunhong Wang, Zhenan Sun, Shiguang Shan, Weishi Zheng, Jianjiang Feng, and Qijun Zhao, editors, *Biometric Recognition*, pages 611–619, Cham, 2016. Springer International Publishing.

[200] C. Patil and S. Patilkulkarni. An approach to enhance security environment based on sift feature extraction and matching to iris recognition. *Information Processing and Management*, page 527–530, 2010.

[201] J. K. Pillai, M. Puertas, and R. Chellappa. Cross-sensor iris recognition through kernel learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(1):73–85, 2014.

[202] Jaishanker K. Pillai, Vishal M. Patel, Rama Chellappa, and Nalini K. Ratha. Secure and robust iris recognition using random projections and sparse representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(9):1877–1893, 2011.

[203] R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.

[204] Ameya Prabhu, Philip H. S. Torr, and Puneet K. Dokania. GDumb: a simple approach that questions our progress in continual learning. *European Conference on Computer Vision (ECCV)*, pages 524–540, 2020.

[205] Hugo Proenca and Luis A. Alexandre. Toward noncooperative iris recognition: A classification approach using multiple signatures. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(4):607–612, 2007.

[206] H. Proença and J. C. Neves. A reminiscence of "mastermind": Iris/periocular biometrics by "in-set" CNN iterative analysis. *IEEE Transactions on Information Forensics and Security (TIFS)*, 14(7):1702–1712, 2019.

[207] Hugo Proença. Iris recognition: On the segmentation of degraded images acquired in the visible wavelength. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(8):1502–1516, 2010.

[208] Hugo Proença and João C. Neves. IRINA: iris recognition (even) in inaccurately segmented data. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6747–6756, 2017.

[209] S. J. Pundlik, D. L. Woodard, and S. T. Birchfield. Non-ideal iris segmentation using graph cuts. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR-W)*, pages 1–6, 2008.

[210] K. R. Radhika, S. V. Sheela, M. K. Venkatesha, and G. N. Sekhar. Multi-modal authentication using continuous dynamic programming. *Biometric ID Management and Multimodal Communication*, pages 228–235, 2009.

[211] R. Raghavendra and Christoph Busch. Robust Scheme for Iris Presentation Attack Detection using Multiscale Binarized Statistical Image Features. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(4):703–715, 2015.

[212] R. Raghavendra, K. B. Raja, and C. Busch. ContlensNet: robust iris contact lens detection using deep convolutional neural networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1160–1167, 2017.

[213] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. *International Conference on Machine Learning (ICML)*, 97:5301–5310, 2019.

[214] K. B. Raja, R. Raghavendra, and C. Busch. Video presentation attack detection in visible spectrum iris recognition using magnified phase information. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10(10):2048–2056, 2015.

[215] K. B. Raja, R. Raghavendra, and C. Busch. Cross-spectrum periocular authentication for NIR and visible images using bank of statistical filters. *International Conference on Imaging Systems and Techniques (IST)*, pages 227–231, 2016.

[216] K. B. Raja, R. Raghavendra, and C. Busch. Scale-level score fusion of steered pyramid features for cross-spectral periocular verification. *International Conference on Information Fusion (Fusion)*, pages 1–7, 2017.

[217] Kiran B. Raja, Raghavendra Ramachandra, and Christoph Busch. Presentation attack detection using laplacian decomposed frequency response for visible spectrum and near-infra-red iris systems. *International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8, 2015.

[218] M R Rajput and G S Sable. IRIS biometrics survey 2010–2015. *IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 2028–2033, 2016.

[219] Raghavendra Ramachandra and Christoph Busch. Presentation attack detection on visible spectrum iris recognition by exploring inherent characteristics of light field camera. *International Joint Conference on Biometrics (IJCB)*, pages 1–8, 2014.

[220] Raghavendra Ramachandra and Christoph Busch. Robust scheme for iris presentation attack detection using multiscale binarized statistical image features. *IEEE Transactions on Information Forensics and Security (TIFS)*, 10:703–715, 2015.

[221] Raghavendra Ramachandra and Christoph Busch. Presentation attack detection methods for face recognition systems: A comprehensive survey. *ACM Computing Surveys*, 50(1):8:1–8:37, 2017.

[222] N. P. Ramaiah and A. Kumar. On matching cross-spectral periocular images for accurate biometrics identification. *International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, 2016.

[223] N. Pattabhi Ramaiah and A. Kumar. Toward more accurate iris recognition using cross-spectral matching. *Transactions on Image Processing (TIP)*, 26(1):208–221, 2017.

[224] Nalini Ratha, Jonathan Connell, and Ruud Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40:614–634, 01 2001.

[225] C. Rathgeb and C. Busch. On the feasibility of creating morphed iris-codes. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 152–157, 2017.

[226] C. Rathgeb, F. Struck, and C. Busch. Efficient BSIF-based near-infrared iris recognition. *International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6, 2016.

[227] Christian Rathgeb, Andreas Uhl, Peter Wild, and Heinz Hofbauer. *Design Decisions for an Iris Recognition SDK*, pages 359–396. Springer London, 2016.

[228] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, G. Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5533–5542, 2017.

[229] Hippolyt Ritter, Aleksandar Botev, and David Barber. Online structured laplace approximations for overcoming catastrophic forgetting. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

[230] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[231] Tang Rongnian and Weng Shaojie. Improving iris segmentation performance via borders recognition. *International Conference on Intelligent Computation Technology and Automation*, 2:580–583, 2011.

[232] O. Ronneberger, P.Fischer, and T. Brox. U-Net: convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 9351:234–241, 2015.

[233] Amir Rosenfeld and John K. Tsotsos. Incremental learning through deep adaptation. *arXiv*, abs/1705.04228, 2018.

[234] A. Ross, S. Banerjee, C. Chen, A. Chowdhury, V. Mirjalili, R. Sharma, T. Swearingen, and S. Yadav. Some research problems in biometrics: The future beckons. *International Conference on Biometrics (ICB)*, 2019.

[235] A. Ross and S. Shah. Segmenting non-ideal irises using geodesic active contours. *Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pages 1–6, 2006.

[236] Wayne J. Ryan, Damon L. Woodard, Andrew T. Duchowski, and Stan T. Birchfield. Adapting starburst for elliptical iris segmentation. *IEEE Second International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–7, 2008.

[237] H. J. Santos-Villalobos, D. R. Barstow, M. Karakaya, C. B. Boehnen, and E. Chaum. ORNL biometric eye model for iris recognition. *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 176–182, 2012.

[238] Mousumi Sardar, Subhashis Banerjee, and Sushmita Mitra. Iris segmentation using interactive deep learning. *IEEE Access*, 8:219322–219330, 2020.

[239] Nadezhda Sazonova, Fang Hua, Xuan Liu, Jeremiah Remus, Arun Ross, Lawrence Hornak, and Stephanie Schuckers. A study on quality-adjusted impact of time lapse on iris recognition. *Proceedings of the SPIE*, 8371:320 – 328, 2012.

[240] U. Scherhag, A. Nautsch, C. Rathgeb, M. Gomez-Barrero, R. Veldhuis, L. Spreeuwers, M. Schils, D. Maltoni, P. Grother, S. Marcel, R. Breithaupt, R. Raghavendra, and C. Busch. Biometric systems under morphing attacks: Assessment of morphing techniques and vulnerability reporting. *International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–7, 09 2017.

[241] U. Scherhag, R. Raghavendra, K. B. Raja, M. Gomez-Barrero, C. Rathgeb, and C. Busch. On the vulnerability of face recognition systems towards morphed face attacks. *International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6, 2017.

[242] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch. Face recognition systems under morphing attacks: A survey. *IEEE Access*, 7:23012–23026, 2019.

[243] S. A. C. Schuckers, N. A. Schmid, A. Abhyankar, V. Dorairaj, C. K. Boyce, and L. A. Hornak. On techniques for angle compensation in nonideal iris recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(5):1176–1190, 2007.

[244] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. *International Conference on Machine Learning (ICLR)*, pages 4528–4537, 2018.

[245] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *The IEEE International Conference on Computer Vision (ICCV)*, 2017.

[246] A. F. Sequeira, S. Thavalengal, J. Ferryman, P. Corcoran, and J. S. Cardoso. A realistic evaluation of iris presentation attack detection. *International Conference on Telecommunications and Signal Processing (TSP)*, pages 660–664, 2016.

[247] S. Shah and A. Ross. Iris segmentation using geodesic active contours. *IEEE Transactions on Information Forensics and Security (TIFS)*, 4(4):824–836, 2009.

[248] A. Sharma, S. Verma, M. Vatsa, and R. Singh. On cross spectral periocular recognition. *International Conference on Image Processing (ICIP)*, pages 5007–5011, 2014.

[249] R. Sharma and A. Ross. D-NetPAD: an explainable and interpretable iris presentation attack detector. *International Joint Conference on Biometrics (IJCB)*, 2020.

[250] Renu Sharma and Arun Ross. Viability of optical coherence tomography for iris presentation attack detection. *International Conference on Pattern Recognition (ICPR)*, 2021.

[251] E. Shechtman, A. Rav-Acha, M. Irani, and S. Seitz. Regenerative morphing. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[252] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[253] Hai Shu and Hongtu Zhu. Sensitivity analysis of deep neural networks. *arXiv*, abs/1901.07152, 2019.

[254] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in Neural Information Processing Systems (NIPS)*, pages 568–576, 2014.

[255] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, 2015.

[256] G. Song, K. K. Chu, S. Kim, M. Crose, B. Cox, E. T. Jelly, N. Ulrich, and A. Wax. First clinical application of low-cost OCT. *Translational vision science and technology (TVST)*, 8(3):61, 2019.

[257] Zhenan Sun and Tieniu Tan. Ordinal measures for iris recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(12):2211–2226, 2009.

[258] Manisha Sam Sunder and Arun Ross. Iris image retrieval based on macro-features. *International Conference on Pattern Recognition (ICPR)*, pages 1318–1321, 2010.

[259] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations (ICLR)*, 2014.

[260] E. Tabassi, P. Grother, and W. Salamon. IREX II - IQCE: Iris Quality Calibration and Evaluation. NIST Interagency/Internal Report (NISTIR) 7820, 2011.

[261] C.-W. Tan and A. Kumar. Accurate iris recognition at a distance using stabilized iris encoding and zernike moments phase features. *IEEE Transactions on Image Processing (TIP)*, 23(9):3962–3974, 2014.

[262] C.-W. Tan and A. Kumar. Efficient and accurate at-a-distance iris recognition using geometric key-based iris encoding. *IEEE Transactions on Information Forensics and Security (TIFS)*, 9(9):1518–1526, 2014.

[263] Chun-Wei Tan and Ajay Kumar. Unified framework for automated iris segmentation using distantly acquired face images. *IEEE Transactions on Image Processing (TIP)*, 21(9):4068–4079, 2012.

[264] Tieniu Tan, Zhaofeng He, and Zhenan Sun. Efficient and robust segmentation of noisy iris images for non-cooperative iris recognition. *Image and Vision Computing (IVC)*, 28(2):223–230, 2010.

[265] S. Thavalengal, T. Nedelcu, P. Bigioi, and P. Corcoran. Iris liveness detection for next generation smartphones. *IEEE Transactions on Consumer Electronics (TCE)*, 62(2):95–102, 2016.

[266] Shejin Thavalengal, Tudor Nedelcu, Petronel Bigioi, and Peter Corcoran. Iris liveness detection for next generation smartphones. *Transactions on Consumer Electronics (TCE)*, 62:95–102, 2016.

[267] The Notre Dame Contact Lense Dataset 2015. https://cvrl.nd.edu/projects/data/#the-notre-dame-contact-lense-dataset-2015ndcld15.

[268] J. Thornton, M. Savvides, and B. V. K. V. Kumar. A bayesian approach to deformed pattern matching of iris images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(4):596–606, 2007.

[269] P. Tome-Gonzalez, F. Alonso-Fernandez, and J. Ortega-Garcia. On the effects of time variability in iris recognition. *International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–6, 2008.

[270] I. Tomeo-Reyes, A. Ross, and V. Chandran. Investigating the impact of drug induced pupil dilation on automated iris recognition. *IEEE International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8, 2016.

[271] Inmaculada Tomeo-Reyes. *Robust Iris Recognition using Decision Fusion and Degradation Modelling*. Ph.D. dissertation, Queensland University of Technology, 2015.

[272] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. *International Conference on Computer Vision (ICCV)*, pages 4489–4497, 2015.

[273] M. Trokielewicz. Linear regression analysis of template aging in iris biometrics. *International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6, 2015.

[274] Mateusz Trokielewicz, Adam Czajka, and Piotr Maciejewicz. Implications of ocular pathologies for iris recognition reliability. *arXiv*, abs/1809.00168, 2018.

[275] Mateusz Trokielewicz, Adam Czajka, and Piotr Maciejewicz. Post-mortem iris recognition with deep-learning-based image segmentation. *Image and Vision Computing (IVC)*, 94:103866, 2020.

[276] Yu-Lin Tsai, Chia-Yi Hsu, Chia-Mu Yu, and Pin-Yu Chen. Formalizing generalization and adversarial robustness of neural networks to weight perturbations. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:19692–19704, 2021.

[277] Gido M Van de Ven and Andreas S Tolias. Generative replay with feedback connections as a general strategy for continual learning. *arXiv*, abs/1809.10635, 2018.

[278] Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *arXiv*, abs/1904.07734, 2019.

[279] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research (JMLR)*, 9(11), 2008.

[280] L.J.P. van der Maaten and G.E. Hinton. Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research (JMLR)*, page 2579–2605, 2008.

[281] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[282] Mayank Vatsa, Richa Singh, and Afzel Noore. Improving iris recognition performance using segmentation, quality enhancement, match score fusion, and indexing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(4):1021–1035, 2008.

[283] Vladan Velisavljevic. Low-complexity iris coding and recognition based on directionlets. *IEEE Transactions on Information Forensics and Security (TIFS)*, 4(3):410–417, 2009.

[284] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch. Face morphing attack generation and detection: A comprehensive survey. *arXiv*, abs/2011.02045, 2020.

[285] F. M. Villalbos-Castaldi and E. Suaste-Gómez. In the use of the spontaneous pupillary oscillations as a new biometric trait. *International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6, 2014.

[286] Ritesh Vyas, Tirupathiraju Kanumuri, and Gyanendra Sheoran. Cross spectral iris recognition for surveillance based applications. *Multimedia Tools and Applications (MTA)*, 78(5):5681–5699, 2019.

[287] Kuo Wang and Ajay Kumar. Cross-spectral iris recognition using CNN and supervised discrete hashing. *Pattern Recognition (PR)*, 86:85–98, 2019.

[288] Limin Wang, Zhe Wang Yuanjun Xiong, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. *European Conference on Computer Vision (ECCV)*, pages 20–36, 2016.

[289] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. ESRGAN: enhanced super-resolution generative adversarial networks. *European Conference on Computer Vision (ECCV) Workshops*, pages 63–79, 2018.

[290] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *Transactions on Image Processing (TIP)*, 13(4):600–612, 2004.

[291] Zhuoshi Wei, Tieniu Tan, and Zhenan Sun. Nonlinear iris deformation correction based on gaussian model. In Seong-Whan Lee and Stan Z. Li, editors, *Advances in Biometrics*, pages 780–789, 2007.

[292] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 2016.

[293] Tsui-Wei Weng, Pu Zhao, Sijia Liu, Pin-Yu Chen, Xue Lin, and Luca Daniel. Towards certificated model robustness against weight perturbations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6356–6363, Apr. 2020.

[294] Peter Wild, James Ferryman, and Andreas Uhl. Impact of (segmentation) quality on long vs. short-timespan assessments in iris recognition performance. *IET Biometrics*, 4:227–235(8), 2015.

[295] R. P. Wildes. Iris recognition: an emerging biometric technology. *Proceedings of the IEEE*, 85(9):1348–1363, 1997.

[296] Yue Wu, Yan-Jia Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. *Conference on Computer Vision and Pattern (CVPR)*, 2019.

[297] Lin Xiang, Xiaoqin Zeng, Yuhu Niu, and Yanjun Liu. Study of sensitivity to weight perturbation for convolution neural network. *IEEE Access*, 7:93898–93908, 2019.

[298] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 1492–1500, 2017.

[299] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. *International Conference On Neural Information Processing (ICONIP)*, 11953:264–274, 2019.

[300] D. Yadav, N. Kohli, J. S. Doyle, R. Singh, M. Vatsa, and K. W. Bowyer. Unraveling the effect of textured contact lenses on iris recognition. *IEEE Transactions on Information Forensics and Security (TIFS)*, 9(5):851–862, 2014.

[301] D. Yadav, N. Kohli, M. Vatsa, R. Singh, and A. Noore. Detecting textured contact lens in uncontrolled environment using DensePAD. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2336–2344, 2019.

[302] Shivangi Yadav, Cunjian Chen, and Arun Ross. Relativistic discriminator: A one-class classifier for generalized iris presentation attack detection. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020.

[303] Shivangi Yadav and Arun Ross. CIT-GAN: Cyclic image translation generative adversarial network with application in iris presentation attack detection. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021.

[304] D. Yambay, B. Becker, N. Kohli, D. Yadav, A. Czajka, K. W. Bowyer, S. Schuckers, R. Singh, M. Vatsa, A. Noore, D. Gragnaniello, C. Sansone, L. Verdoliva, L. He, Y. Ru, H. Li, N. Liu, Z. Sun, and T. Tan. LivDet iris 2017 — Iris liveness detection competition 2017. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 733–741, 2017.

[305] D. Yambay, J. S. Doyle, K. W. Bowyer, A. Czajka, and S. Schuckers. LivDet-iris 2013– iris liveness detection competition 2013. *IEEE International Joint Conference on Biometrics (ICB)*, pages 1–8, 2014.

[306] David Yambay, Brian Walczak, Stephanie Schuckers, and Adam Czajka. LivDet-Iris 2015 — iris liveness detection competition 2015. In *IEEE International Conference on Identity, Security, and Behavior Analysis (ISBA)*, pages 1–6, 2017.

[307] Fei Yan, Yantao Tian, Haiwei Wu, Yanhua Zhou, Liuyang Cao, and Changjiu Zhou. Iris segmentation using watershed and region merging. *IEEE Conference on Industrial Electronics and Applications*, pages 835–840.

[308] Junjie Yan, Zhiwei Zhang, Zhen Lei, Dong Yi, and Stan Z. Li. Face liveness detection by exploring multiple scenic clues. *International Conference on Control Automation Robotics and Vision (ICARCV)*, pages 188–193, 2012.

[309] Daniel S. Yeung, Ian Cloete, Daming Shi, and Wing W.Y. Ng. *Sensitivity Analysis for Neural Networks*. Springer Publishing Company, Incorporated, 2009.

[310] Sowon Yoon, Kwanghyuk Bae, Kang Ryoung Park, and Jaihie Kim. Pan-tilt-zoom based iris image capturing system for unconstrained user environments at a distance. *Advances in Biometrics*, pages 653–662, 2007.

[311] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4353–4361, 2015.

[312] G. Zeng, Y. Chen, B. Cui, and S. Yu. Continual learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1:364–372, 2019.

[313] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. *International Conference on Machine Learning (ICML)*, pages 3987–3995, 2017.

[314] H. Zhang, S. Venkatesh, R. Ramachandra, K. Raja, N. Damer, and C. Busch. MIPGAN – generating robust and high quality morph attacks using identity prior driven GAN. *arXiv*, abs/2009.01729, 2020.

[315] Hui Bin Zhang, Zhenan Sun, Tieniu Tan, and Jianyu Wang. Learning hierarchical visual codebook for iris liveness detection. *International Joint Conference on Biometrics (IJCB)*, 2011.

[316] Zijing Zhao and Ajay Kumar. An accurate iris segmentation framework under relaxed imaging constraints using total variation model. *IEEE International Conference on Computer Vision (ICCV)*, pages 3828–3836, 2015.

[317] Bo-Ren Zheng, Dai-Yan Ji, and Yung-Hui Li. Heterogeneous iris recognition using heterogeneous eigeniris and sparse representation. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3764–3768, 2014.

[318] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[319] J. Zuo and N.A. Schmid. On a methodology for robust segmentation of nonideal iris images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 40(3):703–718, 2010.

[320] Jinyu Zuo and Natalia A. Schmid. An automatic algorithm for evaluating the precision of iris segmentation. *International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–6, 2008.